



# O Sistema Operacional Linux

## IoT para Sistemas Embarcados

Departamento de Engenharia de Controle e Automação  
Instituto de Ciência e Tecnologia – UNESP – Campus Sorocaba

**Prof. Dr. Dhiego Fernandes Carvalho**

dhiego.fernandes@unesp.br

# Objetivos

---

1. Compreender a História e Evolução do Linux
2. Entender a importância do Linux em IoT e Sistemas Embarcados.
3. Introduzir conceitos de Virtualização e Containerização.
4. Dominar os comandos básicos do Linux.
5. Aplicar comandos de Rede no Linux.

# Índice

---

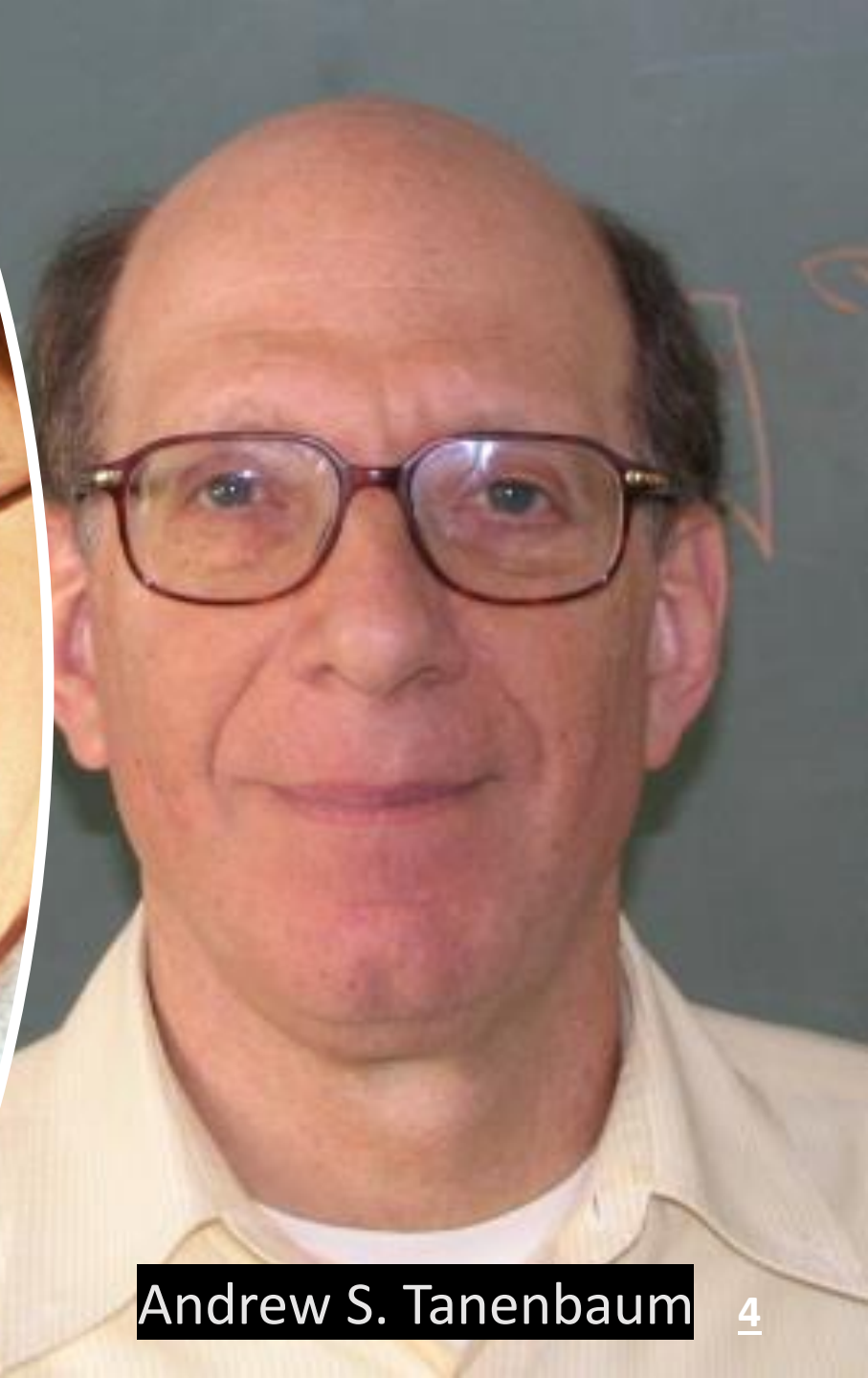
1. História do Linux
2. Linux em Sistemas Embarcados
3. Virtualização
4. Containerização
5. Principais comandos básicos no Linux
6. Comandos de Rede no Linux.
7. Conclusões

# História do Linux

- Em 1991 um estudante Finlandês, Linus Torvalds, iniciou um projeto de criação de um Sistema Operacional como hobby, baseado no sistema Minix (mini Unix), desenvolvido por Andrew S. Tanenbaum.



Linus Torvalds

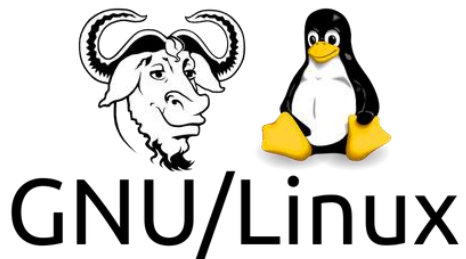


Andrew S. Tanenbaum



# História do Linux

- O Linus Torvalds foi responsável pela criação do kernel do sistema operacional.
- Em 1983 foi lançado por Richard Stallman o projeto GNU com o objetivo de criar um SO completamente livre.
- O projeto GNU desenvolveu muitas ferramentas essenciais que são usadas em SO modernos (compilador GCC, emacs, utilitários shells etc).
- Por tais motivos, muitos chamam o Linux de GNU/Linux.



Richard Stallman

# História do Linux

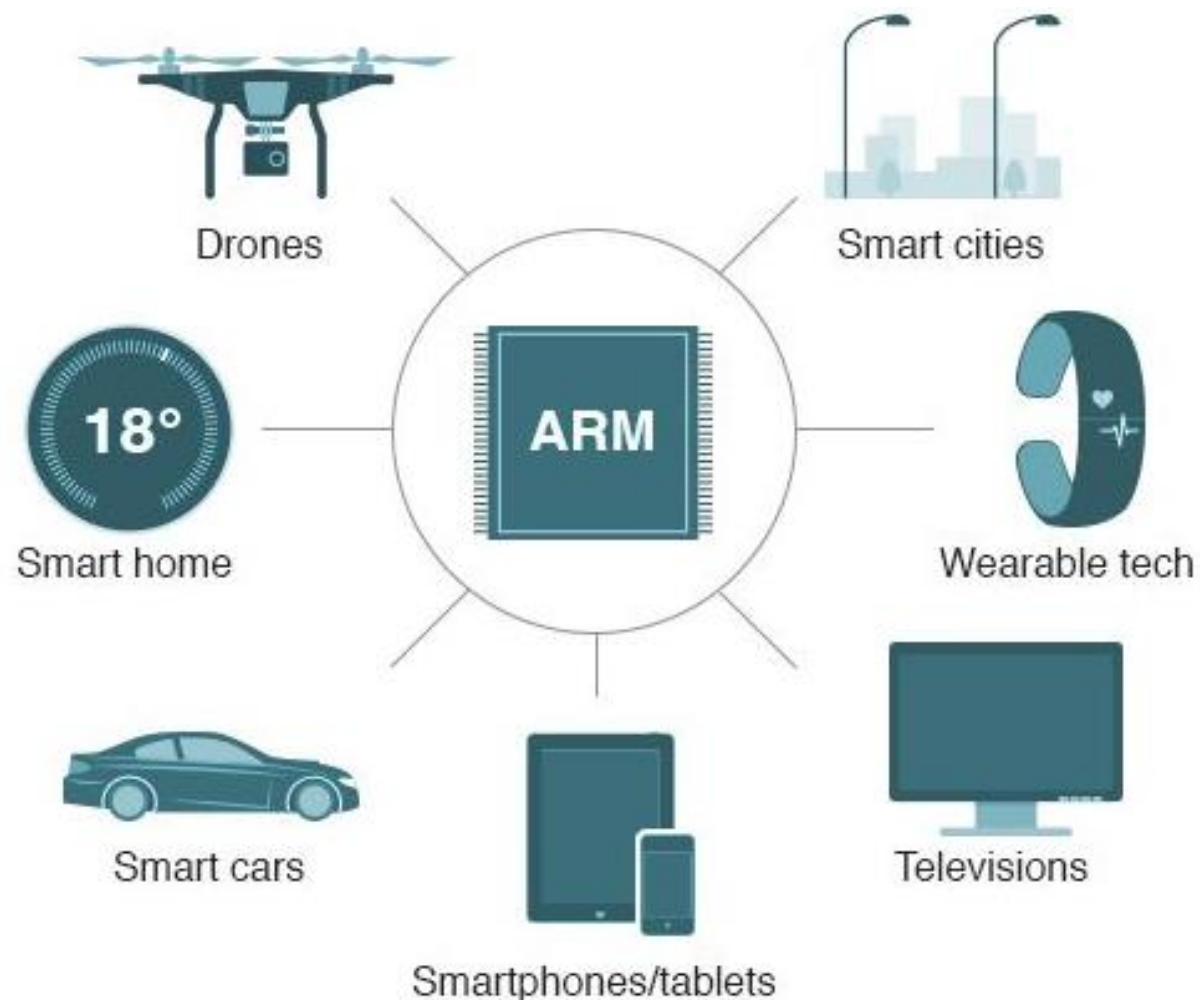
- O Projeto GNU foi fundado com um forte compromisso com os princípios de software livre.
- Para garantir tal liberdade, foi criada a licença GPL (*General Public License*).



- A partir da licença GPL, foi possível criar várias distribuições Linux.

# Linux em Sistemas Embarcados

- Os sistemas embarcados mais complexos usam uma arquitetura ARM (*Advanced RISC Machine*), diferente das x86 usadas nos computadores pessoais e servidores.
- A arquitetura ARM tem como características sua eficiência energética, baixo custo e alta personalização.
- Pode-se dizer que a arquitetura ARM é a arquitetura dos dispositivos IoT.

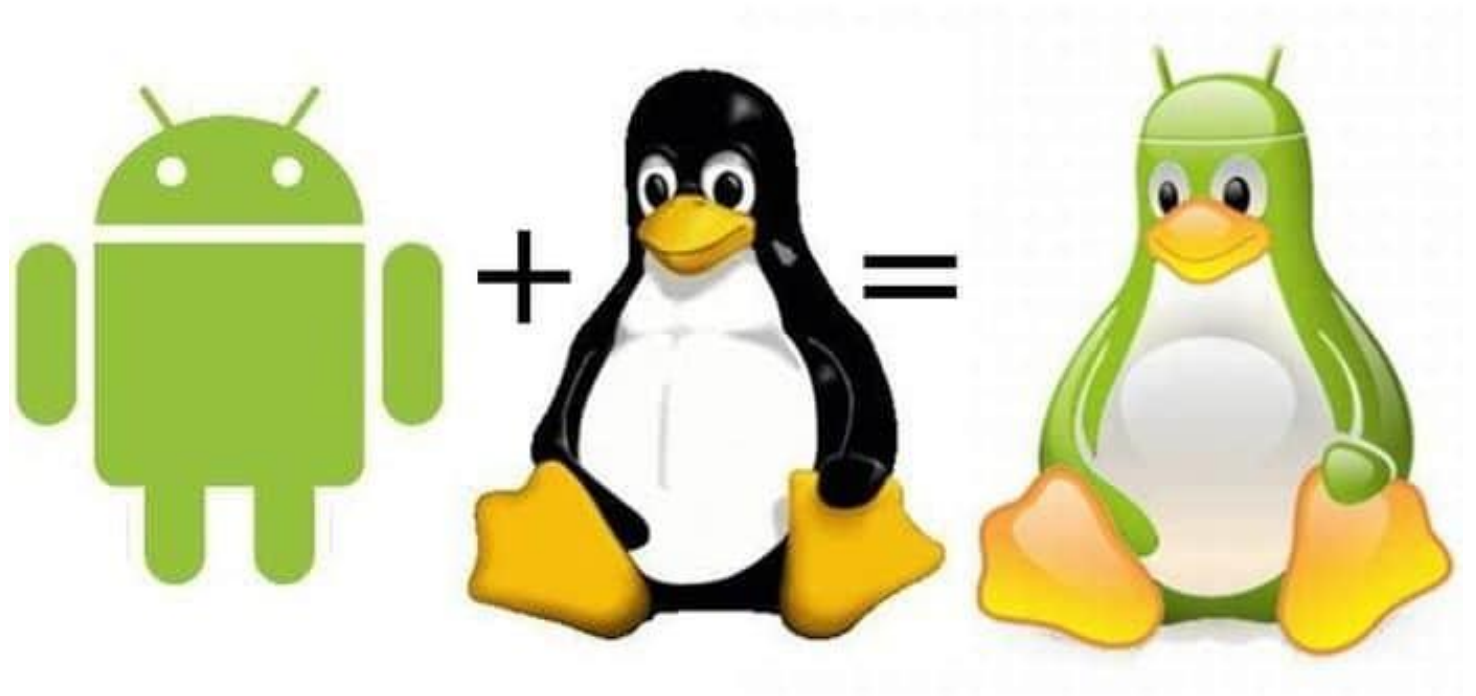


# Linux em Sistemas Embarcados

- O Android é um SO baseado no Kernel do Linux, sendo mantida pela *Android Open Source Project* (AOSP), encabeçada pelo Google.
- O Android possui um ecossistema de Aplicativos disponíveis através da Google Play Store.



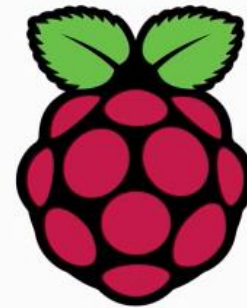
Google Play





# Linux em Sistemas Embarcados

Nos Raspberry Pi, a distribuição Linux mais comum a ser usada é Raspberry Pi OS (antigo Raspbian).



**Raspberry Pi OS**

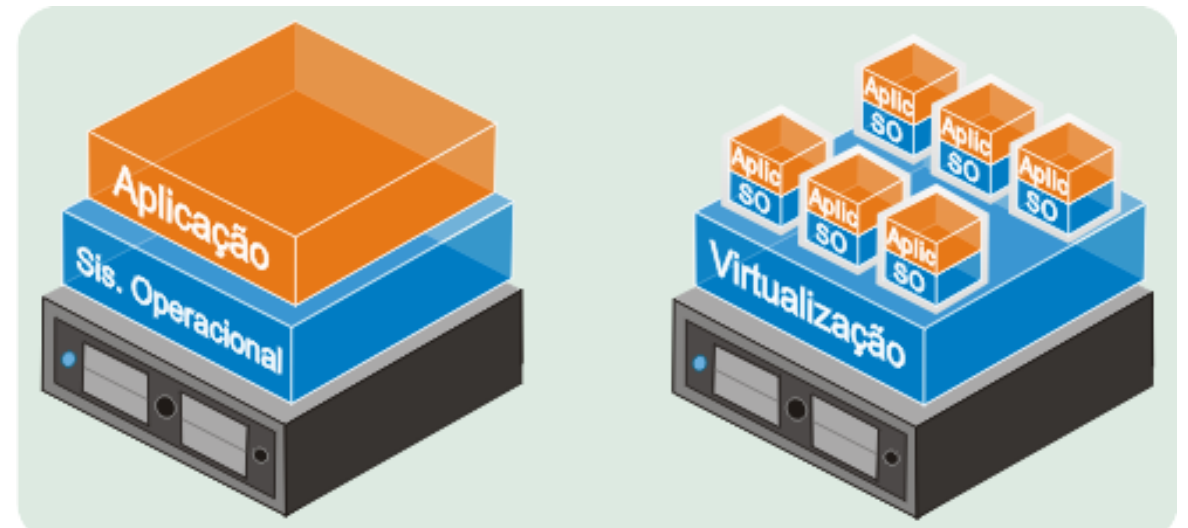
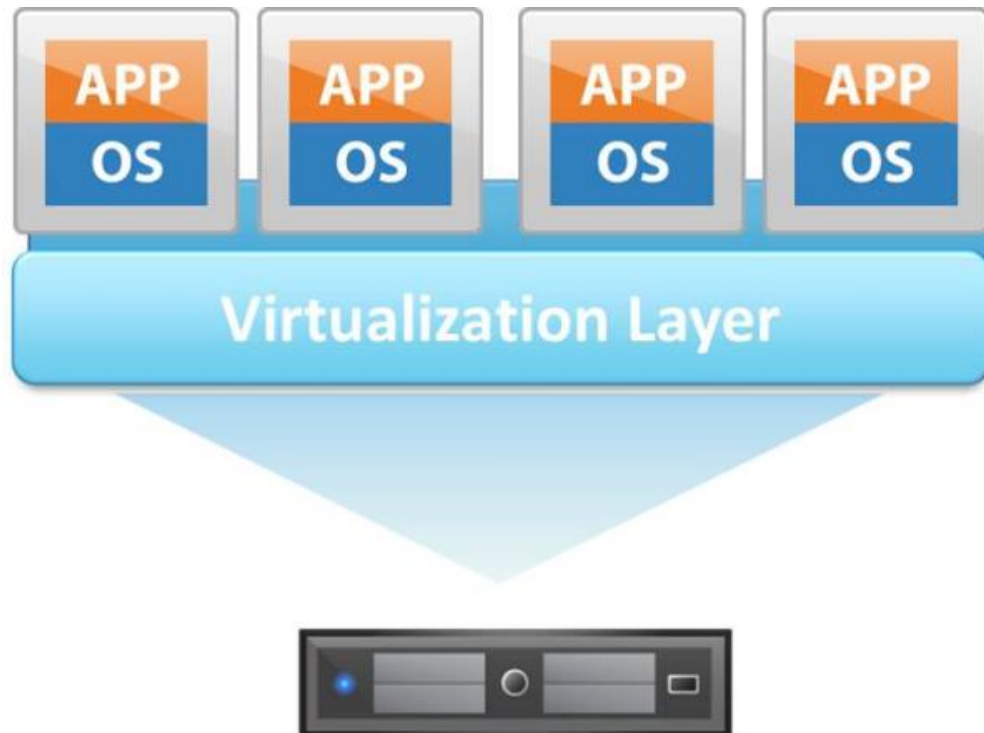
# Linux em Sistemas Operacionais

- O Armbian é baseado no Debian ou Ubuntu focado para a arquitetura ARM.
- As TVBoxs vêm com um Android TV instalado, mas pode ser substituído por uma versão do Armbian.



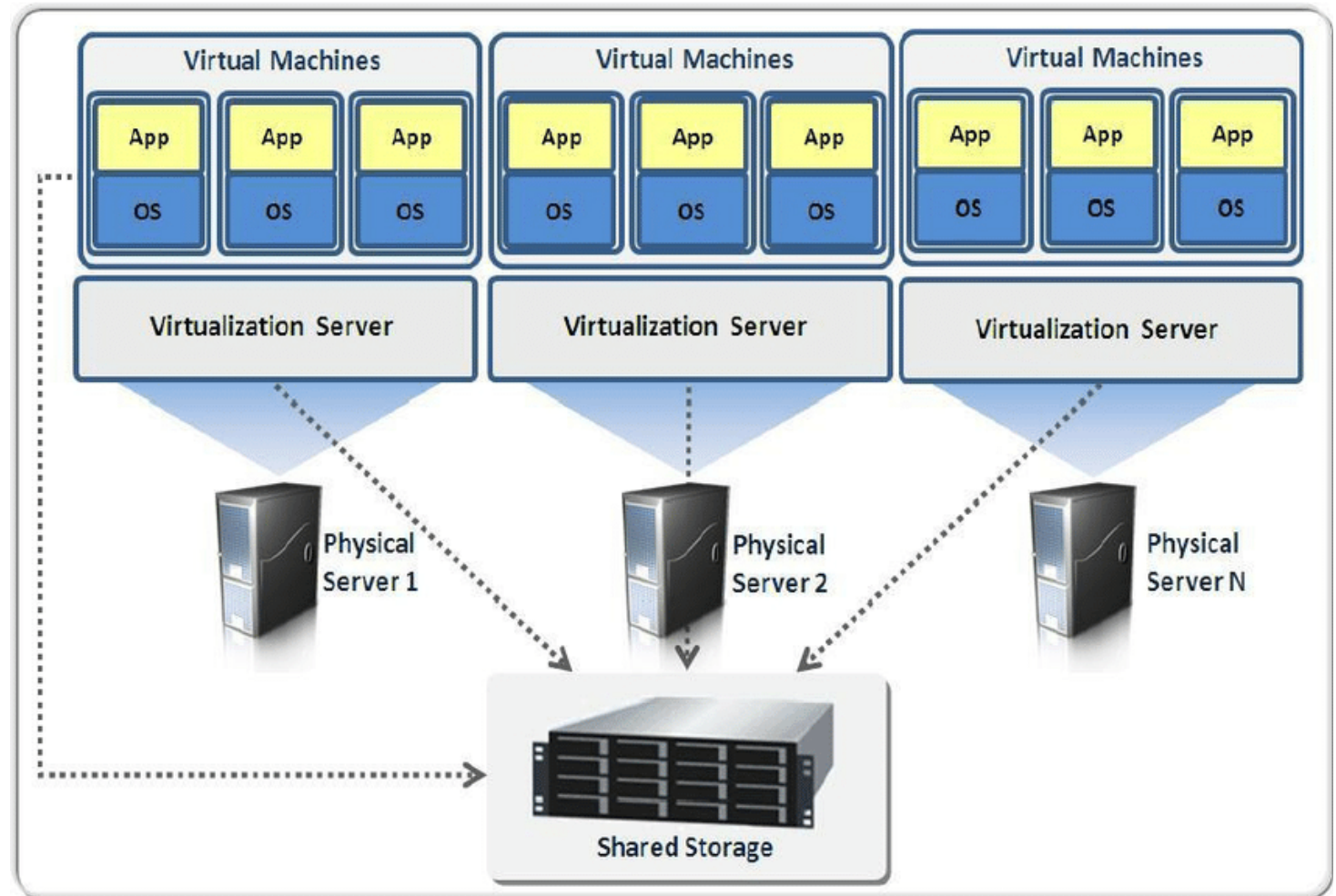
# Virtualização

- É uma tecnologia que permite executar múltiplos SOs simultaneamente em um único hardware físico, compartilhando seus recursos.



# Virtualização

- Benefícios:
  - Menos máquinas físicas.
  - Isolamento e segurança.
  - Flexibilidade e escalabilidade.
  - Desenvolvimento e Teste.
  - Recuperação de Desastres e Continuidade de Negócios.

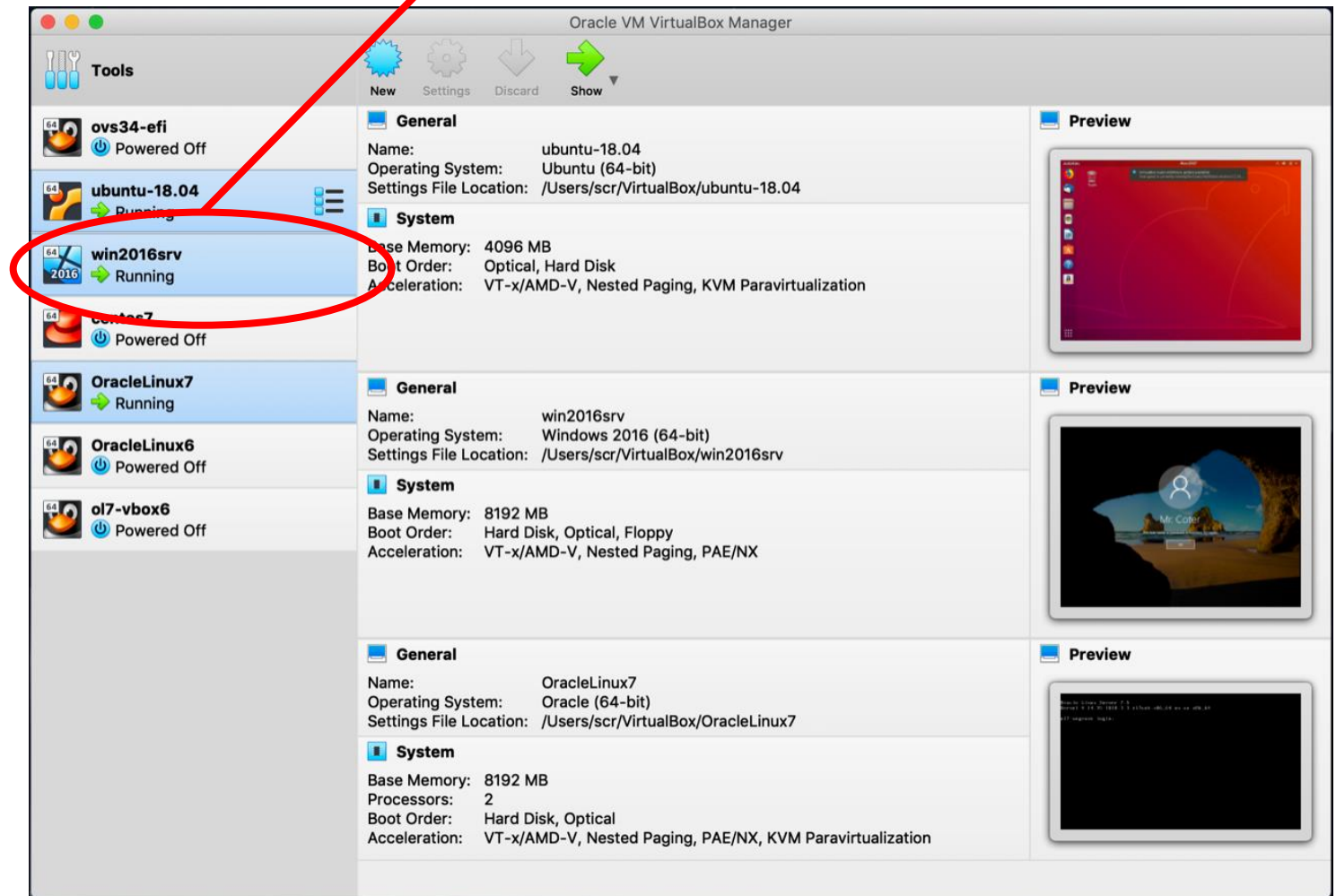




# Virtualização

- No nosso caso, seria a instalação de um Sistema Operacional Linux em uma máquina com Windows.

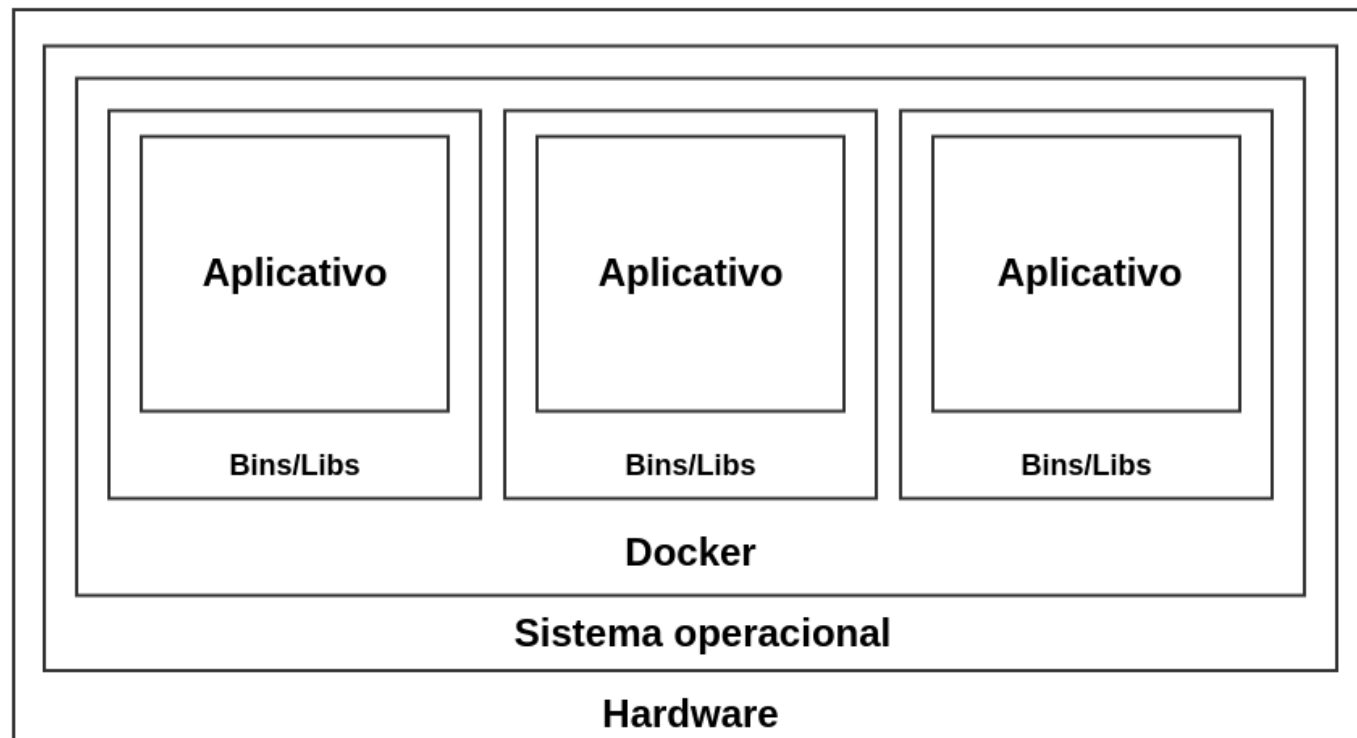
Cada Máquina Virtual tem seu próprio SO, com hardware virtual próprio (disco, memória, processador etc)





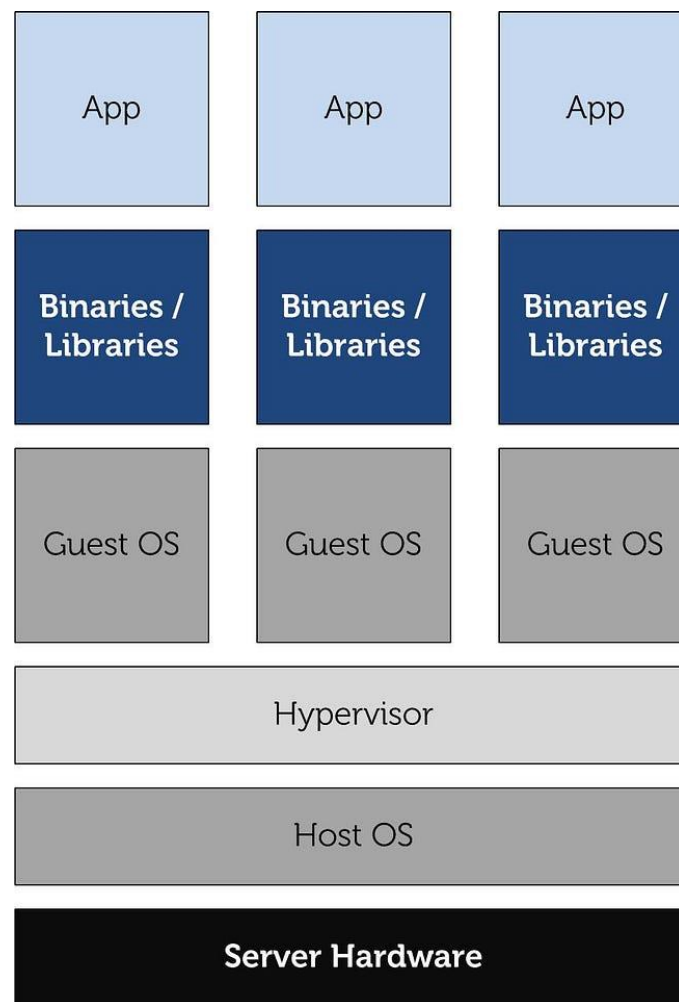
# Containerização

- A containerização é um processo de implantação de software que agrupa o código de uma aplicação com todos os arquivos e bibliotecas de que ela precisa para ser executado em qualquer infraestrutura.

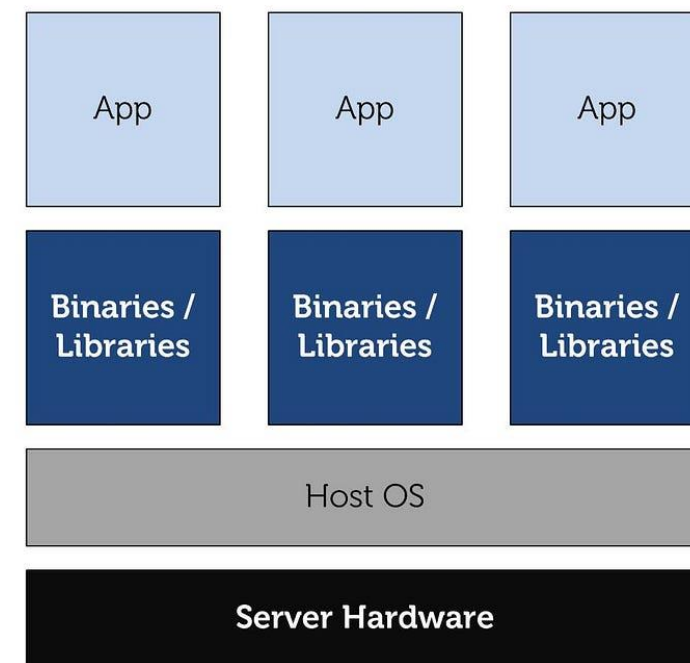


# Containerização

- Ao contrário da virtualização tradicional, a containerização permite que múltiplos contêineres compartilhem o mesmo kernel do SO do host, reduzindo significativamente o uso de recursos.



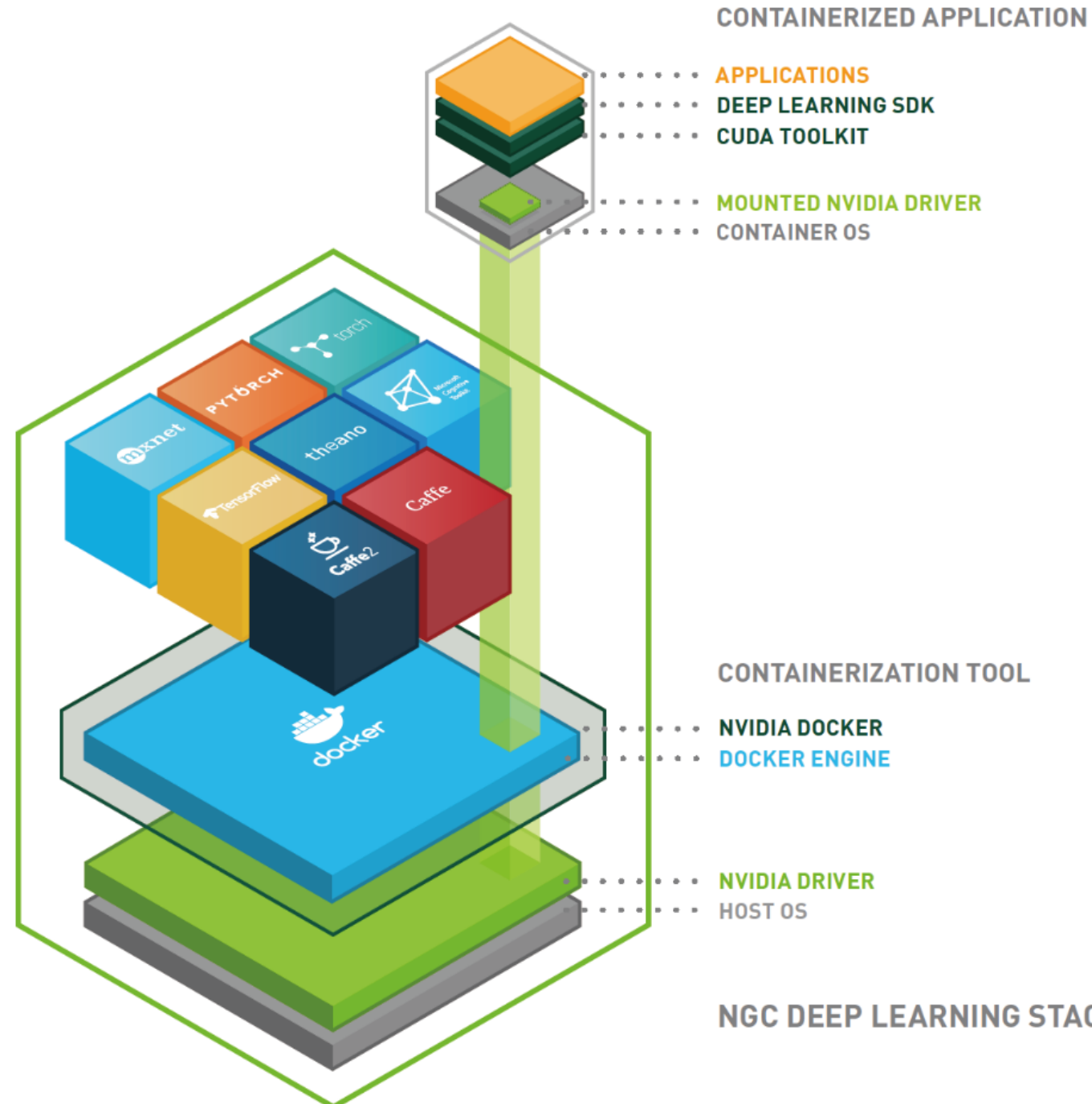
Virtualization



Containers

# Containerização

- Os contêineres são usados em:
  - Aplicações Web
  - Microserviços
  - Aplicações de Big Data e Análise
  - Aplicações de IA e Machine Learning



# Containerização

- Um ótimo programa para Containerização é o Docker.
- No nosso caso, pode ser usado para rodar programas com bibliotecas do Linux em ambiente Windows.

The screenshot shows the Docker Desktop interface. At the top, there's a blue header with 'Docker Desktop' and an 'Upgrade plan' button. Below the header, the 'Images' section is active, showing a progress bar for '52.11 MB / 983.69 MB in use' and '25 images'. The 'LOCAL' tab is selected, displaying a table of local images. The first image, 'docker/desktop-storage-provisioner', is circled in red. A red arrow points from the text 'Imagem rodando no Docker Desktop' to this image. The table has columns for NAME, TAG, STATUS, CREATED, and SIZE. Other images listed include 'docker/desktop-vpnkit-controller', 'docker/getting-started', 'mongo', 'rabbitmq', and 'weaveworksdemos/carts'. The status of 'docker/getting-started' is 'In use', while others are 'Unused'.

NAME	TAG	STATUS	CREATED	SIZE
docker/desktop-storage-provisioner 115d77efe6e2	v2.0	Unused	15 days ago	16.16
docker/desktop-vpnkit-controller b5bcb3a7172a	v2.0	Unused	15 days ago	5.87
docker/getting-started d79336f4812b	latest	In use	less than a minute ago	20.75
mongo d314f7eca9d3	3.4	Unused	13 days ago	156.49
rabbitmq a9f4923559bb	3.6.8	Unused	13 days ago	86.35
weaveworksdemos/carts 424d2f5a6a0a	0.4.8	Unused	13 days ago	98.29

AM 4.35 GB CPU 4.68% Disk 47.47 GB avail. of 58.37 GB Connected to Hub

# Principais comandos básicos do Linux

## Navegação Básica

### #Mostrar conteúdo do diretório

```
usuario@linux:~$ ls
```

Documentos Downloads Imagens Desktop

### #Navegar entre as pastas

```
usuario@linux:~$ cd Documentos
```

### #Mostrar diretório corrente

```
usuario@linux:~/Documentos$ pwd
```

```
/home/usuario/Documentos
```



# Principais comandos básicos do Linux

## Manipulação de Arquivos

### #Criação de Diretório

```
usuario@linux:~/Documentos$ mkdir ProjetoIoT
```

### #Criação de Arquivo

```
usuario@linux:~/Documentos$ touch ProjetoIoT/README.md
```

### #Copiar arquivos

```
usuario@linux:~/Documentos$ cp ProjetoIoT/README.md ProjetoIoT/LEIAME.md
```

### #Mover arquivos

```
usuario@linux:~/Documentos$ mv ProjetoIoT/LEIAME.md ProjetoIoT/LEIA-ME.txt
```

### #Remover Arquivo

```
usuario@linux:~/Documentos$ rm ProjetoIoT/LEIA-ME.txt
```

### #Remover Diretório

```
usuario@linux:~/Documentos$ rmdir ProjetoIoT
```

# Principais comandos básicos do Linux

## Gerenciamento de Processos

### #Mostrar processos

```
usuario@linux:~/Documentos$ ps
```

PID	TTY	TIME	CMD
-----	-----	------	-----

1234	pts/1	00:00:00	bash
------	-------	----------	------

5678	pts/1	00:00:00	ps
------	-------	----------	----

### #Matar Processo

```
usuario@linux:~/Documentos$ kill -9 5678
```

# Principais comandos básicos do Linux

## Informações do Sistema

```
usuario@linux:~/Documentos$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda1	50G	12G	35G	26%	/

```
usuario@linux:~/Documentos$ free -m
```

	total	used	free	shared	buff/cache	available
Mem:	3912	1342	1125	125	1444	2200
Swap:	1023	200	823			

# Principais comandos básicos do Linux

## Usar o nano para editar arquivos de texto

```
usuario@linux:~$ nano meu_arquivo.txt
```

**#Faz alterações no arquivo**

**##Sair sem salvar**

Pressiona: Ctrl + X

Pergunta: Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?

Pressiona: N

**##Salvar e sair**

Pressiona: Ctrl + X

Pergunta: Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?

Pressiona: Y

# Principais comandos básicos do Linux

## Inserir usuário no sudoers

```
usuario@linux:~$ su
```

Senha:

```
root@linux:/home/usuario$ usermod -aG sudo usuario
```

```
root@linux:/home/usuario$ exit
```

logout

```
usuario@linux:~$
```



# Principais comandos básicos do Linux

## Manipulação de Usuários

### #Adicionar Usuário

```
usuario@linux:~$ sudo adduser novo_usuario
```

### #Deletar Usuário

```
usuario@linux:~$ sudo deluser novo_usuario
```

### #Entrar e Sair do mesmo usuário

```
usuario@linux:~$ su novo_usuario
```

Senha:

```
novo_usuario@linux:~$ exit
```

logout

# Principais comandos básicos do Linux

## Instalação de Programas

### #Atualização

```
usuario@linux:~$ sudo apt update
```

### #Instalar programas

```
usuario@linux:~$ sudo apt install net-tools
```

# Comandos de Rede do Linux

## Mostrar interfaces de Redes

```
usuario@linux:~$ ip a
```

```
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
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 01:23:45:67:89:ab brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.2/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
        valid_lft 86398sec preferred_lft 86398sec
    inet6 fe80::a00:27ff:fe4e:66a1/64 scope link
        valid_lft forever preferred_lft forever
```

# Comandos de Rede do Linux

## Mostrar portas abertas e os serviços rodando nelas

```
usuario@linux:~$ netstat -tuln
```

Active Internet connections (only servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN
tcp6	0	0	:::80	:::*	LISTEN
udp	0	0	0.0.0.0:68	0.0.0.0:*	
udp	0	0	0.0.0.0:123	0.0.0.0:*	
udp6	0	0	:::123	:::*	

# Comandos de Rede do Linux

## Comando PING – Verificar se Máquina está ativa na rede

```
usuario@linux:~$ ping google.com
```

```
PING google.com (172.217.17.78) 56(84) bytes of data.
```

```
64 bytes from ams15s29-in-f14.1e100.net (172.217.17.78): icmp_seq=1 ttl=115 time=14.8 ms
```

```
64 bytes from ams15s29-in-f14.1e100.net (172.217.17.78): icmp_seq=2 ttl=115 time=13.7 ms
```

```
64 bytes from ams15s29-in-f14.1e100.net (172.217.17.78): icmp_seq=3 ttl=115 time=14.1 ms
```

```
64 bytes from ams15s29-in-f14.1e100.net (172.217.17.78): icmp_seq=4 ttl=115 time=13.8 ms
```

```
^C
```

```
--- google.com ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
```

```
rtt min/avg/max/mdev = 13.714/14.125/14.815/0.433 ms
```



# Comandos de Rede do Linux

## Requisição de Endereço IP

```
usuario@linux:~$ sudo dhclient -v eth0
Internet Systems Consortium DHCP Client 4.3.5
Copyright 2004-2016 Internet Systems Consortium.
All rights reserved.
Listening on LPF/eth0/00:1a:2b:3c:4d:5e
Sending on  LPF/eth0/00:1a:2b:3c:4d:5e
Sending on  Socket/fallback
DHCPRELEASE on eth0 to 192.168.1.1 port 67
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 3
DHCPOFFER from 192.168.1.1
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 192.168.1.1
bound to 192.168.1.2 -- renewal in 21600 seconds.
```

# Comandos de Rede do Linux

## Acesso Remoto via SSH

```
usuario@local:~$ ssh joao@192.168.1.100
```

```
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be  
established.
```

```
ECDSA key fingerprint is SHA256:xh3H...g8w2.
```

```
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

```
Warning: Permanently added '192.168.1.100' (ECDSA) to the list of known  
hosts.
```

```
joao@192.168.1.100's password:
```

```
joao@remote:~$
```

# Comandos de Rede do Linux

## Varredura de Portas Remotas

```
usuario@linux:~$ nmap 192.168.1.1
```

```
Starting Nmap 7.80 ( https://nmap.org ) at 2023-01-01 12:00 EST
```

```
Nmap scan report for 192.168.1.1
```

```
Host is up (0.0010s latency).
```

```
Not shown: 995 closed ports
```

```
PORT      STATE SERVICE
```

```
22/tcp    open  ssh
```

```
80/tcp    open  http
```

```
443/tcp   open  https
```

```
3306/tcp  open  mysql
```

```
8080/tcp  open  http-proxy
```

```
Nmap done: 1 IP address (1 host up) scanned in 6.76 seconds
```

# Conclusões

---

- Nessa aula foi visto a história do Sistema Operacional GNU/Linux, sua filosofia e como ele é usado em sistemas embarcados (arquitetura ARM).
- Foi ensinado o que são Virtualização e Containerização, assim como suas diferenças.
- Foi explicado os principais comandos básicos e de rede do Linux.
- A partir dessa aula, é possível o aluno instalar e manusear o Linux.

# DÚVIDAS?

---

# Exercícios

---

- Baixe o [VirtualBox](#) e instale no seu computador.
- Baixe a última versão do [Debian](#).
- Crie uma Máquina Virtual Linux no Virtualbox e em seguida instale o Debian nela.
- Depois da instalação do Debian, pratique os principais comandos do Linux ensinados nesta aula.