

# Material de apoio

Servidores são componentes fundamentais em redes de computadores e na Internet. Eles são sistemas de hardware ou software projetados para fornecer serviços, recursos ou funcionalidades a outros dispositivos, chamados de clientes, por meio de uma rede. Os servidores desempenham um papel essencial na infraestrutura de tecnologia da informação e são vitais para o funcionamento de aplicativos e serviços online.

## Servidores locais e servidores na nuvem

Um servidor local é um servidor físico que está instalado em uma infraestrutura física, como um data center local na empresa ou no local onde os recursos de computação são necessários. Qualquer tipo de computador pode se tornar um servidor local: seu computador principal; algum computador que estava sem utilização ou uma Raspberry Pi,

Um servidor na nuvem é hospedado em um data center remoto, geralmente em um provedor de serviços em nuvem, como Amazon Web Services (AWS), Microsoft Azure ou Google Cloud Platform. Os recursos estão localizados em servidores remotos e são acessados pela Internet. Em resumo, você faz uma assinatura e passa a poder utilizar computadores dessas grandes empresas para hospedar sua aplicação.

O problema de utilizar um servidor na nuvem é a dependência da internet. Se a internet cai, ou se o local não tem acesso a internet, pode comprometer todo o seu sistema IoT. Com uma rede local, ou **intranet**, não é necessário a conexão com o resto do mundo. A raspberry Pi se torna o centro desta rede local.

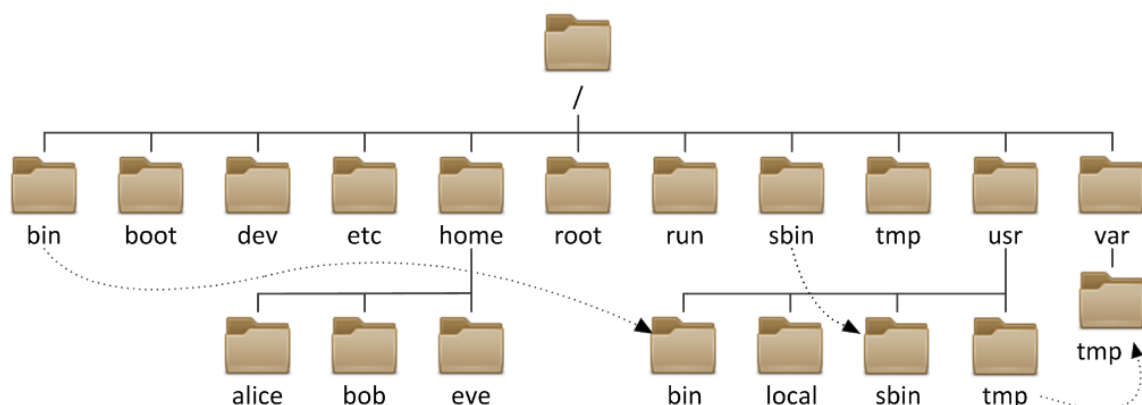
Além disso, caso necessário, é possível criar uma conexão entre seu servidor local e o servidor na nuvem. Desta forma, os dados do seu sistema ficam disponíveis mundialmente. Caso aja queda da conexão, o sistema IoT não é comprometido.

## Sistema Linux

Linux é um sistema operacional de código aberto amplamente utilizado que se baseia no kernel Linux, desenvolvido por Linus Torvalds em 1991. Conhecido por sua estabilidade, segurança e flexibilidade, o Linux é uma escolha popular em servidores, supercomputadores, dispositivos embarcados e até mesmo em desktops. Sua arquitetura de código aberto permite que os usuários e desenvolvedores personalizem e modifiquem o sistema de acordo com suas necessidades, tornando-o uma opção versátil e acessível para uma ampla variedade de aplicações e cenários de computação.

O kernel do Linux é o núcleo do sistema operacional. Uma distribuição completa do Linux consiste no kernel e em várias outras ferramentas de software para operações relacionadas a arquivos, gerenciamento de usuários e gerenciamento de pacotes de software. Cada uma dessas ferramentas fornece uma parte do sistema completo. Cada ferramenta geralmente é seu próprio projeto separado, com seus próprios desenvolvedores trabalhando para aperfeiçoar essa parte do sistema.

Qualquer servidor utiliza um sistema linux. Por isso, é importante estar familiarizado com o sistema de diretórios. Não existe dificuldade, apenas requer um tempo de adaptação.



- / → Diretório root, ou diretório raiz. É onde todos os outros diretórios estão contidos. É equivalente ao C:/ do windows;
- /bin → É o diretório de binários. É onde ficam armazenados os programas essenciais, bem como os comandos que

utilizamos no terminal;

- /boot → Contém os arquivos de boot loader e de kernel do sistema;
- /dev → Contém arquivos que representam hardwares conectados ao sistema;
- /etc → É onde estão os arquivos de configurações do sistema;
- /home → Onde os diretório dos usuários ficam armazenados os diretórios de usuários;
- /lib → Bibliotecas de vários programas do sistema;
- /media → Mostra hardwares removíveis, como CS, DVD e PenDrives;
- /mnt → Utilizado pelo sistema para montar arquivos temporários;
- /opt → Softwares adicionais que não são parte do sistema principal;
- /proc → Fornece informações sobre processos que estão em execução;
- /run → Arquivos temporários e serviços do sistema;
- /sbin → Arquivos binários de manutenção do sistema;
- /srv → Armazena dados dos serviços em execução;
- /sys → Sistema de arquivo virtual que contém informações do hardware;
- /tmp → Arquivos temporários em geral. São perdidos quando o sistema reinicia;
- /usr → Arquivos de usuários;
- /var → Arquivos variados, que mudam constantemente.

## Raspberry Pi

A Raspberry Pi é uma série de computadores de placa única (SBCs) desenvolvidos pela Raspberry Pi Foundation, uma

organização sem fins lucrativos com o objetivo de promover a educação em ciência da computação e tornar a tecnologia acessível a pessoas em todo o mundo. O primeiro modelo foi lançado em 2012, e desde então, várias gerações foram produzidas, cada uma com melhorias de desempenho e recursos. Os modelos que podem ser adquiridos que suportam rodar brokers mqtt e node-red são:

- Raspberry Pi 3, 4 ou 5. Esses modelos se diferenciam pelo seu tempo de mercado, sendo o raspberry Pi 5 o mais atualizado (e potente).
- Raspberry Pi Zero 2w. É a versão mais simples capaz de rodar o sistema operacional da raspberry. Certifique-se de adquirir a versão com WiFi.

A principal diferença entre os modelos é o desempenho. A mais simples já satisfaz os nossos requerimentos. O modelo Raspberry Pi 5 8Gb é o mais potente e pode ser utilizado como um computador pessoal.

Além da placa, você deverá adquirir:

- Cartão de memória class 10 com, no mínimo, 16 Gb.
- Fonte 5v 3A com conector USB C. Dê preferência para fontes feitas especialmente para a raspberry.
- Kit de refrigeração. É opcional, mas é recomendado.
- Case para raspberry Pi. Alguns kit de refrigeração já vêm embutidos em cases.
- Teclado, mouse e monitor hdmi.

## Instalando mosquitto e node-red na raspberry Pi

```
sudo apt update && sudo apt upgrade -y
```

```
sudo apt install vim
```

```
sudo apt install mosquitto -y
```

```
sudo apt install mosquitto-clients
```

## Atenção!!!!

O **vim** é um editor de texto para linha de comando exclusivo do linux. Ele tende a ser melhor que o **nano**, mas requer um certo aprendizado:

- Depois de abrir, antes escrever qualquer coisa, pressione a tecla **i** para entrar no modo inserção e digite o conteúdo;
- Depois de escrever o conteúdo desejado, tecle **esc** para sair do modo inserção
- Para salvar um arquivo e sair, digite: **wq!** e tecle **enter** (fora do modo inserção).
- Para sair de um arquivo sem salvar, digite **qa!**

Outras funcionalidades como copiar, colar, selecionar, etc. requerem mais comandos. Pesquise sobre a paleta de comandos completa do vim no google ou tutoriais para aprender mais.

Mas caso não queira, utilize o editor **nano**. Basta substituir **vim** por **nano** na linha de comando para utilizar este outro editor.

- Para salvar com nano pressione **ctrl+o** e depois *enter*. Para sair pressione **ctrl+x**.

## Definindo senha

```
sudo touch /etc/mosquitto/passwd
```

```
sudo mosquitto_passwd /etc/mosquitto/passwd admin
```

## Editando o mosquitto.conf

```
sudo vim /etc/mosquitto/mosquitto.conf
```

```
pid_file /run/mosquitto/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d

listener 1883
allow_anonymous false

password_file /etc/mosquitto/passwd
```

Mantenha o mosquitto rodando pelo `sudo systemctl start mosquitto`

## Instalação do Node

Link de instalação do node:

<https://github.com/nodesource/distributions>

```
node -v
```

```
npm -v
```

## Instalação do Node-Red

```
sudo npm install -g --unsafe-perm node-red
```

Executar → `node-red`

Para executar o node-red automaticamente via gerenciador de processos PM2 do linux

```
sudo npm install -g pm2
```

```
sudo pm2 start /usr/bin/node-red OU sudo pm2 start /usr/local/bin/node-red
```

```
sudo pm2 save
```

```
sudo pm2 startup
```

## AWS e computação em nuvem

Amazon Web Services (AWS) é uma plataforma de serviços em nuvem líder global, oferecida pela [Amazon.com](https://www.amazon.com). Lançada em 2006, a AWS fornece uma ampla gama de serviços de computação em nuvem, armazenamento, banco de dados, aprendizado de máquina, análise, Internet das Coisas (IoT), segurança e muito mais. Sua infraestrutura escalável e flexível permite que empresas, organizações governamentais e desenvolvedores implantem e gerenciem aplicativos e serviços de forma eficiente, pagando apenas pelos recursos que utilizam. A AWS desempenhou um papel fundamental na transformação digital, possibilitando inovações tecnológicas e permitindo que empresas de todos os tamanhos alcancem agilidade e escalabilidade em seus projetos de TI. Com uma presença global, a AWS continua a ser uma escolha popular para aqueles que buscam soluções de computação em nuvem confiáveis e de alto desempenho.

## Instalação dos pacotes na aws

Siga os mesmos passos da instalação dos pacotes na raspberry.

**Depois,** execute:

```
sudo apt install apache2 -y
```

```
sudo rm /var/www/html/index.html
```

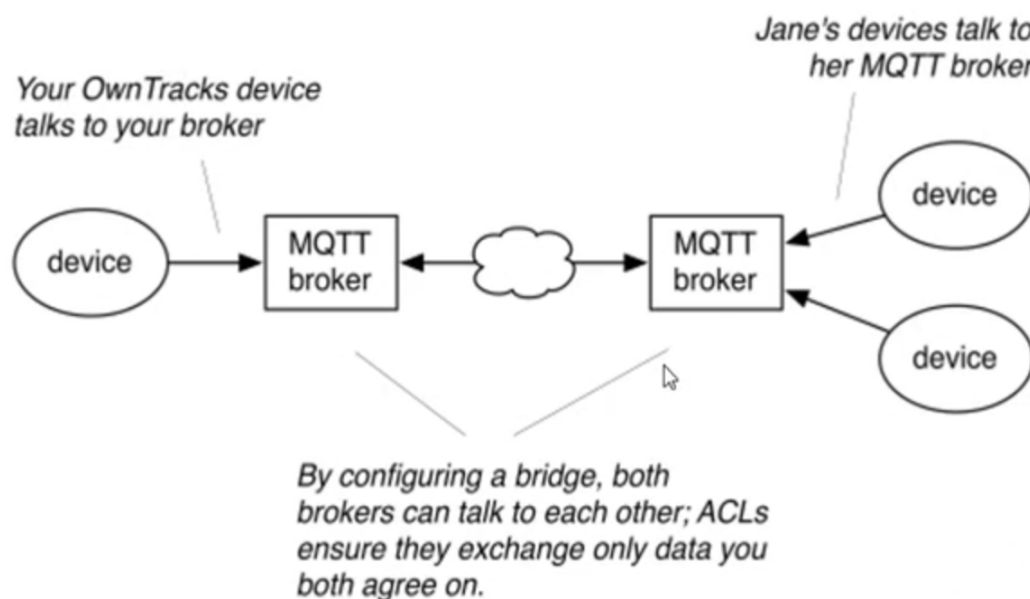
```
sudo vim /var/www/html/index.html
```

Insira o seguinte conteúdo:

```
<html>
  <script>
    window.location.replace("http://<host_da_maquina>:1880/ui")
  </script>
</html>
```

## Criando uma ponte (bridge) entre o broker da raspberry com o broker da AWS

O modo bridge do mosquitto conecta dois brokers diferentes e espelha os dados entre eles:



Isto significa que, o que acontecer em um broker, irá acontecer no outro. Qualquer informação que o broker MQTT da raspberry receber, broker MQTT da AWS também receberá. Desta

forma, fica muito fácil implementar uma conexão entre seu sistema IOT local com a nuvem.

Para criar o bridge, basta configurar apenas um dos brokers. Para facilitar, vamos configurar a raspberry

```
sudo vim /etc/mosquitto/mosquitto.conf
```

**Adicione** as seguintes configurações dentro do mosquitto.conf

```
connection nuvem
address 3.95.0.117:1883

# mesma regra dos wildcards
topic # out 0
topic # in 0

# defina um id para o cliente
remote_clientid aws_broker

# A sessão é mantida em caso de queda
cleansession false

notifications false
start_type automatic
try_private true
remote_username admin
remote_password 1234
```