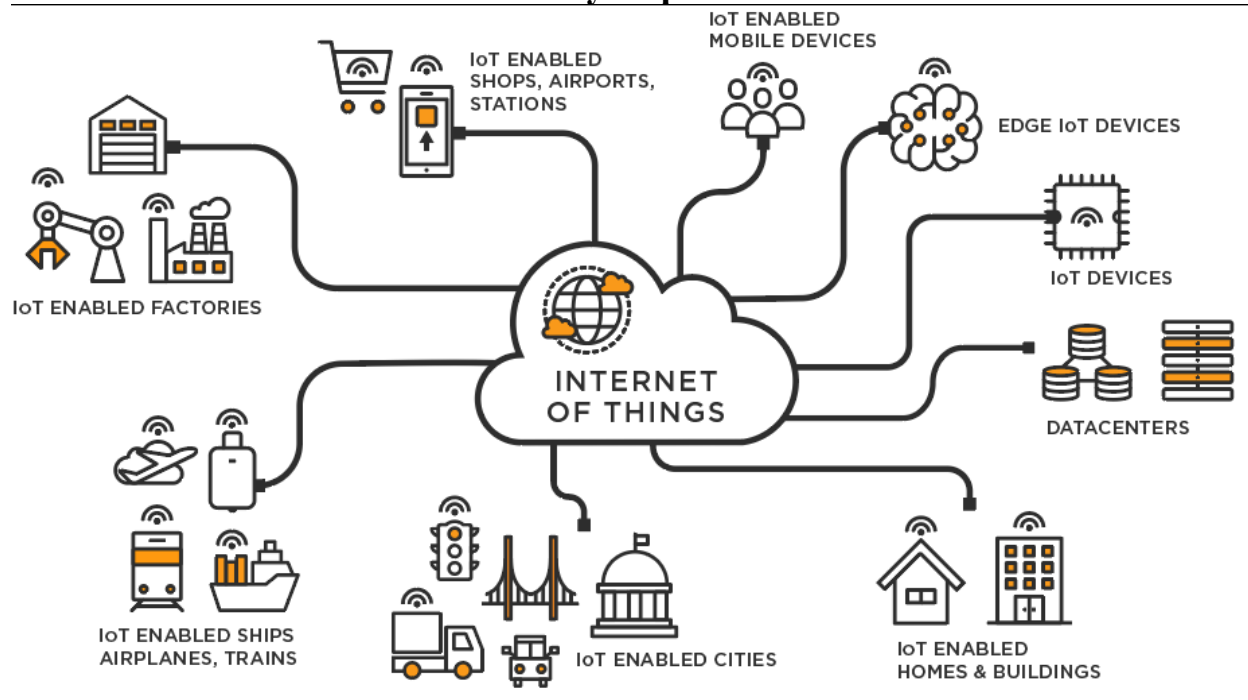


# WEB IOT STEP BY STEP Đỗ Duy Cốp - 2025



## MỤC LỤC

BÀI TOÁN: XÂY DỰNG WEB IOT MONITOR+REPORT .....	5
Sơ đồ mạng tại công ty: .....	5
Có các giải pháp để cài đặt/chỉnh sửa từ xa:.....	5
1. Cài đặt ssh và dùng tài khoản admin để remote access over putty .....	5
1.1. PuTTY là gì? .....	5
1.2. Đặc trưng của PuTTY .....	6
1.3. Download và cài đặt tool OpenSSH .....	6
1.4. Kiểm tra xem OpenSSH đã cài đặt thành công: .....	7
1.5. Lấy thông tin tài khoản đăng nhập Windows .....	7
1.6. Cài đặt Putty .....	7
1.7. Mở cổng 22 trên firewall .....	7
2. Cài đặt VPN Tailscale và Zerotier: để remote qua VPN Tunnel .....	7
2.1. Tailscale là gì?.....	8
2.2. Điểm mạnh của Tailscale .....	8
2.3. Cài đặt Tailscale .....	8
2.4. ZeroTier là gì?.....	9
2.5. Điểm mạnh của ZeroTier .....	9
2.6. Cài đặt Zerotier .....	10
3. Cài đặt Node.js .....	11
3.1. Node.js là gì?.....	11
3.2. Điểm mạnh của Node.js .....	11
3.3. Nhược điểm của Node.js.....	11
3.4. Các bước cài đặt Node.js trên windows.....	11
4. Cài đặt Node-RED: .....	12
4.1. Node-RED là gì?.....	12
4.2. Đặc trưng của Node-RED .....	12
4.3. Cài đặt Node-RED .....	12
4.4. Cấu hình để nodered luôn tự động chạy cùng windows .....	12
4.5. Kiểm tra Node-RED có đang chạy ? .....	14
4.6. Cấu hình Nodered .....	14
5. Cài đặt MariaDB .....	14
5.1. MariaDB là gì?.....	14
5.2. Đặc trưng của MariaDB .....	15

5.3. Các bước cài đặt MariaDB trên Windows .....	15
5.4. Các công cụ quản trị MariaDB .....	15
5.5. Áp dụng:.....	15
5.6. Tổ chức dữ liệu cho Web IOT.....	16
5.7. Tạo database và các table bằng tool HeidiSQL .....	17
6. Cài đặt influxDB 2.7 .....	18
6.1. InfluxDB là gì?.....	18
6.2. Đặc trưng của InfluxDB.....	18
6.3. Các bước cài đặt InfluxDB trên Windows .....	19
6.4. InfluxDB-CLI là gì?.....	20
6.5. Cài đặt InfluxDB-CLI trên Windows.....	21
7. Cài đặt Grafana .....	21
7.1. Grafana là gì?.....	21
7.2. Tác dụng của Grafana .....	21
7.3. Ưu điểm.....	22
7.4. Nhược điểm.....	22
7.5. Cách cài đặt Grafana trên Windows .....	22
7.6. Cấu hình để chạy grafana qua sub path .....	23
8. Cài đặt Emqx : MQTT .....	24
8.1. MQTT là gì?.....	24
8.2. EMQX là gì? .....	24
8.3. Đặc trưng của EMQX .....	24
8.4. Điểm mạnh.....	25
8.5. Điểm yếu .....	25
8.6. Cài đặt EMQX trên Windows .....	25
9. Cài đặt Python và các thư viện cần thiết.....	27
9.1. Python là gì?.....	27
9.2. Ưu điểm của Python.....	27
9.3. FastAPI là gì?.....	27
9.4. Ưu điểm của FastAPI.....	28
9.5. Cài đặt Python trên Windows .....	28
9.6. Cài đặt FastAPI .....	28
10. Viết code Back-End: Python+FastAPI+MQTT .....	28
10.1. Mục tiêu .....	28

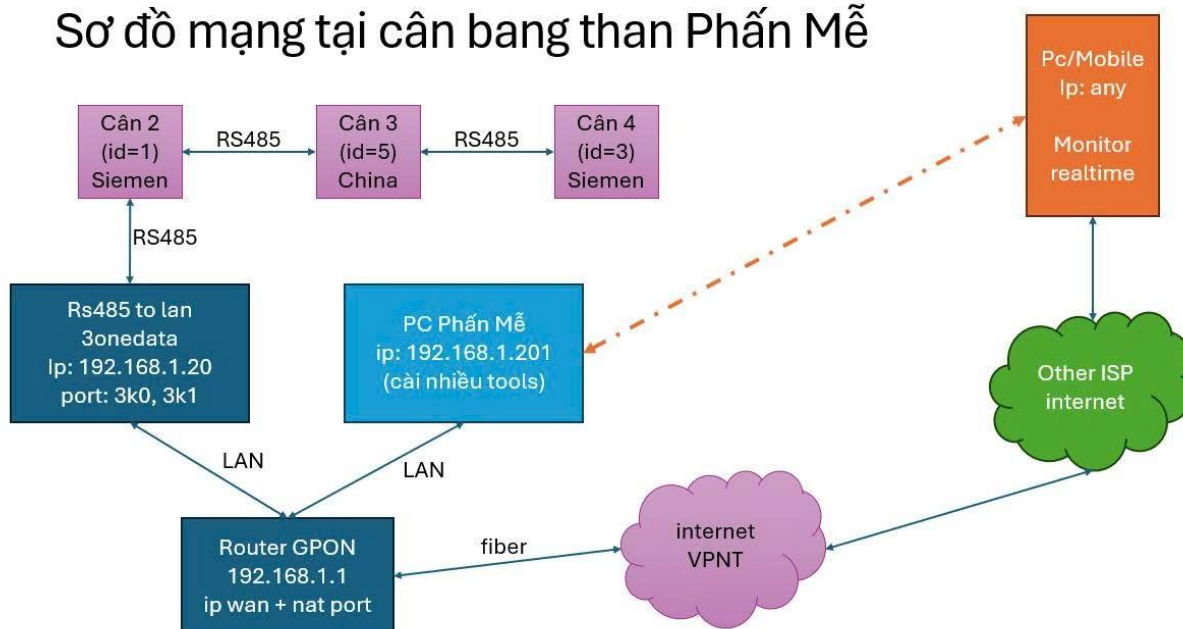
10.2. Thuật toán (tóm tắt hoạt động run-time).....	29
10.3. full source code python: sử dụng FastAPI và MQTT.....	29
10.4. Cấu hình chạy tự động như dịch vụ (run as service) .....	30
11. Cài đặt web server Apache.....	30
11.1. Apache Web Server là gì? .....	30
11.2. Đặc trưng của Apache .....	30
11.3. Ưu điểm.....	30
11.4. Nhược điểm (so với web server hiện đại như Nginx).....	30
11.5. Các bước cài đặt Apache trên Windows .....	31
11.6. Cấu hình web cùng các dịch vụ backend và grafana .....	31
12. Viết code Front-End : Html+JS+CSS .....	31
12.1. Mục tiêu của code (toàn trang) .....	31
12.2. Thuật toán hoạt động.....	32
12.3. Ý nghĩa của hai selector CSS.....	32
12.4. Full source-code html css js.....	32
13. Tạo các đồ thị trên Grafana.....	32
13.1. Kết nối Grafana với InfluxDB 2.7 .....	32
13.2. Tạo Dashboard và Panel (đồ thị) .....	33
13.3. Lưu Dashboard.....	33
13.4. Dùng đồ thị Grafana trên web (iframe): .....	33
14. Đăng ký domain trên DuckDNS .....	33
14.1. DuckDNS là gì? .....	33
14.2. Các bước đăng ký free host trên DuckDNS.....	33
14.3. Token của DuckDNS để làm gì?.....	34
15. Tạo form đăng nhập .....	34
16. Truy cập gpon router để mở port 80 443 .....	34
17. Tool acme để tạo chứng chỉ ssl, cấu hình https trên apache .....	36
17.1. SSL là gì? .....	36
17.2. Tại sao HTTPS lại an toàn? .....	36
17.3. Tool ACME là gì? .....	36
17.4. Chức năng của ACME tool.....	36
17.5. Cách dùng ACME cho web chạy với Apache.....	36

# BÀI TOÁN: XÂY DỰNG WEB IOT MONITOR+REPORT

Có 1 máy tính (desktop) nằm tại 1 công ty. Cần sử dụng máy tính này làm web server. Cần cài đặt công cụ để có thể làm việc từ xa với máy tính này.

Sơ đồ mạng tại công ty:

## Sơ đồ mạng tại cân bang than Phấn Mễ



Có các giải pháp để cài đặt/chỉnh sửa từ xa:

- + Sử dụng UltraView: khi remove vào là bị phát hiện
- + Sử dụng TeamView: khi remove vào là bị phát hiện
- + Sử dụng Remote Desktop (có sẵn): chiếm quyền điều khiển của windows
- + Sử dụng OpenSSH để remote access qua putty: im lặng tuyệt đối, hữu ích nhất khi làm việc từ xa

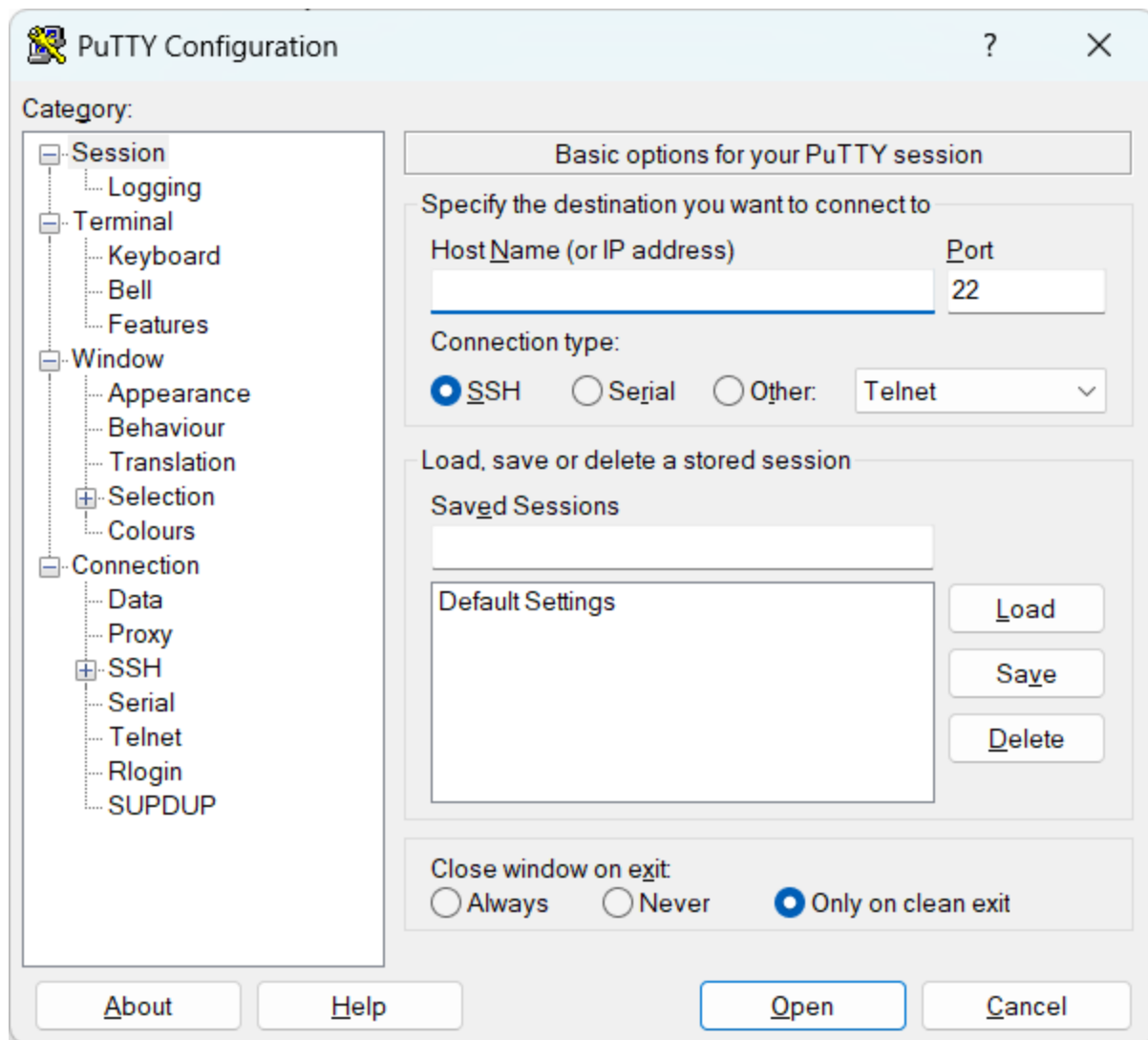
## 1. Cài đặt ssh và dùng tài khoản admin để remote access over putty

### 1.1. PuTTY là gì?

PuTTY là một phần mềm miễn phí, mã nguồn mở dùng để kết nối từ xa đến máy chủ qua nhiều giao thức mạng:

- SSH (Secure Shell) – phổ biến nhất, để quản trị Linux/Unix server.
- Telnet – kiểu cũ, ít an toàn.
- Rlogin, Serial (COM port), Raw TCP.

Nói ngắn gọn: PuTTY là cửa sổ để bạn “ngồi” ở máy Windows mà điều khiển server ở nơi khác.



## 1.2. Đặc trưng của PuTTY

- Gọn nhẹ, chạy ngay không cần cài đặt (có bản .exe portable).
- Hỗ trợ SSH key (public/private key) ngoài password.
- Có PuTTYgen (tạo SSH key), Pageant (agent lưu key).
- Cho phép lưu nhiều profile kết nối (host, port, user).
- Dùng được cả để debug thiết bị phần cứng qua Serial (RS232/COM).

## 1.3. Download và cài đặt tool OpenSSH

Thực hiện trên máy công ty

Bản mới nhất tại: <https://github.com/powershell/win32-openssh/releases>

Cách dùng: (Tự tìm hiểu)

#### 1.4. Kiểm tra xem OpenSSH đã cài đặt thành công:

Thực hiện trên máy công ty, Mở Cmd, nhập lệnh: `netstat -ano | findstr ":22 "`  
Nếu thấy 0.0.0.0:22 LISTENING và [::]:22 LISTENING là OK

#### 1.5. Lấy thông tin tài khoản đăng nhập Windows

Thực hiện trên máy công ty, lấy username và mật khẩu của tài khoản windows. Nếu không nhớ thì WIN+R: rồi nhập **compmgmt.msc** để enable user Administrator, nên đổi thành admin hoặc tên riêng, set pw mới.  
Thông tin này được dùng để remote access từ máy khác sang máy này.

#### 1.6. Cài đặt Putty

Thực hiện trên laptop. Cài đặt [putty](https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html),

Down tại: <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

Nếu cùng LAN, dùng putty truy cập ssh vào máy công ty, tạo shortcut:

`"C:\Program Files\PuTTY\putty.exe" admin@192.168.1.2 -pw 12345678`

Trong đó: (thay 3 giá trị bằng thông số riêng của bạn)

- + Tên đăng nhập vào windows: admin
- + Mật khẩu đăng nhập: 12345678
- + ip của máy tính đích: 192.168.1.2 (ip máy đã cài OpenSSH ở bước 1.1)
- +

#### 1.7. Mở cổng 22 trên firewall

- + Mở cổng 22 trên máy công ty để cho phép từ xa kết nối đến OpenSSH.
- + Chạy CMD với quyền Administrator, copy lệnh sau, rồi chuột phải để paste vào cmd:
- + `netsh advfirewall firewall add rule name="Allow SSH (22)" dir=in action=allow protocol=TCP localport=22`

## 2. Cài đặt VPN Tailscale và Zerotier: để remote qua VPN Tunnel

Có các cách truy cập vào OpenSSH:

- + Truy cập vào máy tính công ty trong cùng mạng LAN: cần mở cổng 22 trong firewall
- + Truy cập từ xa (qua internet) trực tiếp vào máy tính công ty trong LAN : cần mở port 22 trong firewall, và Nat port trên router (phức tạp)
- + Tạo VPN Tunnel (đường hầm VPN) trên máy ở công ty: cần mở cổng 22 trong firewall, không cần Nat port trên router => hữu ích nhất khi làm việc từ xa.

VPN Tunnel có Tailscale và Zerotier là dễ dùng, miễn phí. Sau khi cài đặt và cấu hình thành công trên máy công ty, và trên laptop: 2 máy này sẽ trao đổi được files với nhau mà không cần cùng LAN. Chỉ cần có internet là OK.

## 2.1. Tailscale là gì?

Tailscale là một dịch vụ **VPN dạng mesh (mạng ngang hàng)**, xây dựng trên giao thức **WireGuard**.

Thay vì dựng VPN thủ công, Tailscale biến mọi thiết bị (PC, laptop, điện thoại, server, IoT) thành **một mạng riêng an toàn** chỉ trong vài phút cài đặt.

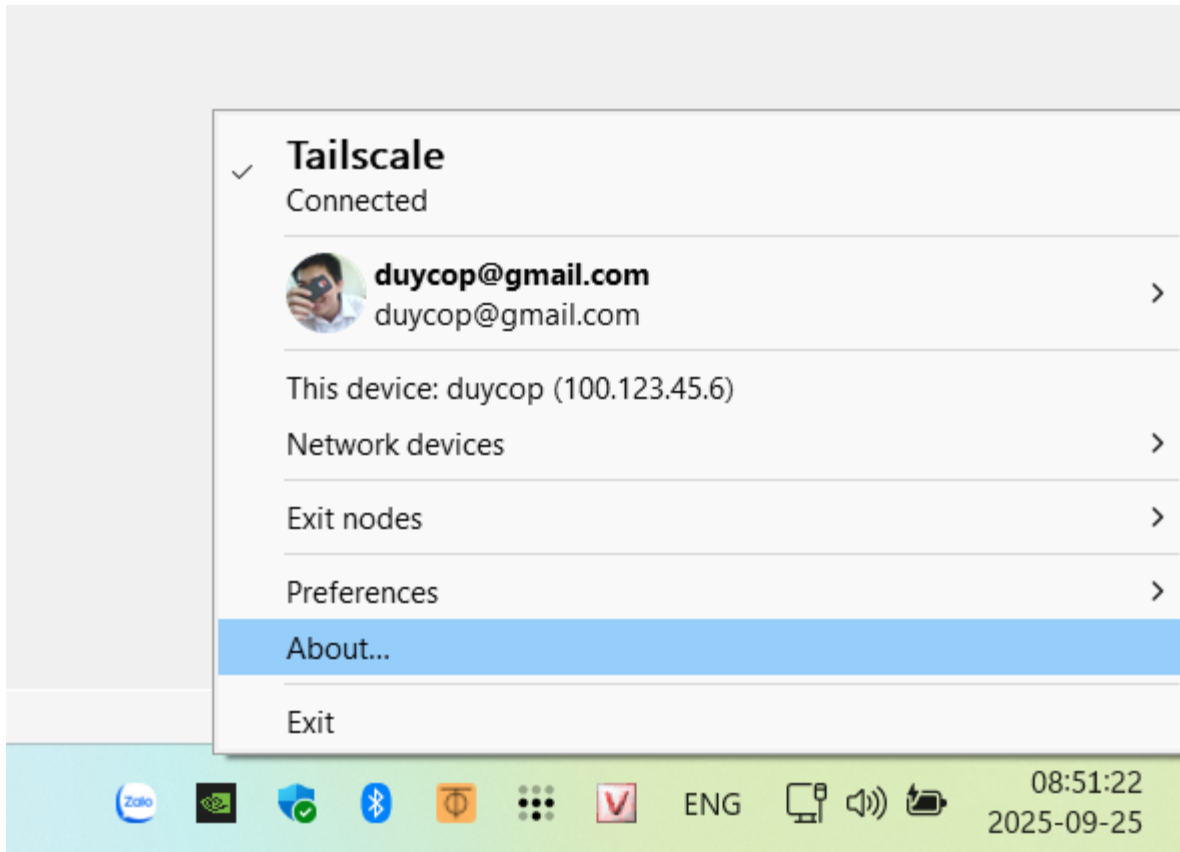
## 2.2. Điểm mạnh của Tailscale

- **Rất dễ dùng:** chỉ cần đăng nhập bằng Google/Microsoft/GitHub account là xong, không phải cấu hình router/firewall phức tạp.
- **Zero-config NAT traversal:** các máy ở sau NAT, firewall vẫn tự tìm và kết nối được (peer-to-peer).
- **Bảo mật cao:** dựa trên **WireGuard** (mã hóa nhanh, an toàn).
- **Đa nền tảng:** Windows, macOS, Linux, Android, iOS, thậm chí cả router và Docker.
- **Access Control:** có ACL (Access Control List) để giới hạn ai được truy cập máy nào.
- **Tích hợp dễ dàng:** kết nối server cloud (AWS, GCP, Azure) với máy nội bộ mà không cần mở port.
- **Free plan:** đủ xài cho cá nhân hoặc nhóm nhỏ.

## 2.3. Cài đặt Tailscale

- Download [Tailscale](https://tailscale.com/download) tại <https://tailscale.com/download>
- Chọn phiên bản phù hợp với hệ điều hành của bạn.
- Nhược điểm: Giới hạn 100 node cho 1 tài khoản gmail.
- Cài đặt (next all) xong cần cấu hình:
  - + Login vào Tailscale qua giao diện UI
  - + Signup bằng cách link với google account có trước đó (đừng quên pw tài khoản google này)
  - + Disable key expiry (để không bị hết hạn chia sẻ)
  - + Chia sẻ quyền truy cập và copy link chia sẻ
  - + Tại máy ở xa (laptop tại nhà) đăng nhập Tailscale khác và truy cập link chia sẻ, ghi nhớ IP tailscale cấp phát. Sử dụng ip này tại laptop sẽ truy cập được vào desktop công ty như trong cùng mạng Lan với desktop công ty.
  - + Giả sử ip tailscale của máy tính công ty là **123.45.67.89** thì shortcut trên laptop tại nhà để truy cập vào máy tính công ty sẽ là:  
**"C:\Program Files\PuTTY\putty.exe" admin@123.45.67.89 -pw 12345678**





## 2.4. ZeroTier là gì?

ZeroTier là một **nền tảng mạng ảo (virtual network overlay)**. Nó tạo ra một **LAN ảo** chạy trên Internet, cho phép các thiết bị ở nhiều nơi khác nhau (PC, server, điện thoại, IoT) kết nối lại với nhau như thể đang chung một mạng nội bộ.

Khác với VPN truyền thống (kiểu site-to-site), ZeroTier linh hoạt hơn: nó vừa là **VPN**, vừa giống như một **SD-WAN mini** (mạng diện rộng ảo).

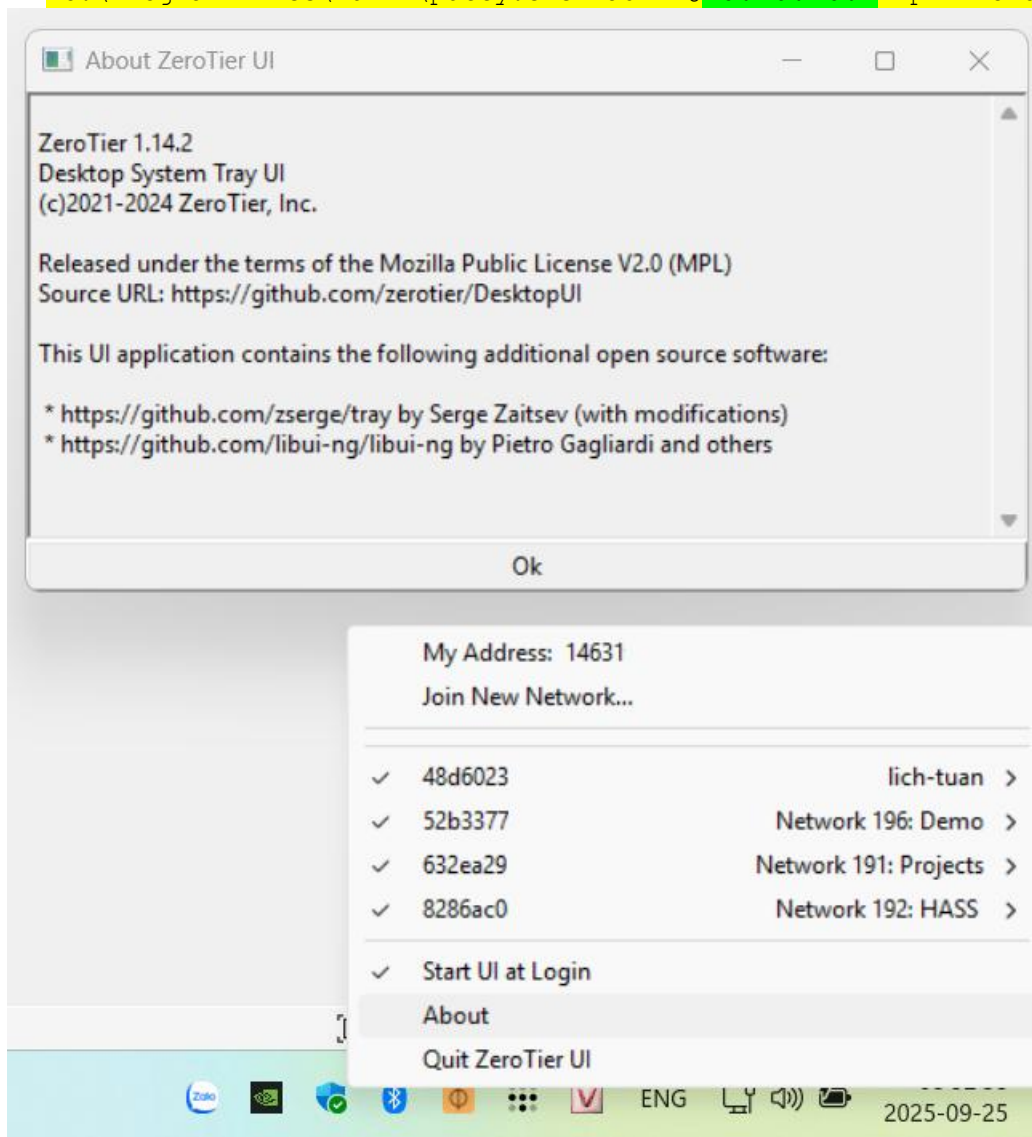
## 2.5. Điểm mạnh của ZeroTier

- **Dễ dùng:** cài phần mềm, đăng nhập, join vào “network ID” là xong.
- **NAT traversal thông minh:** thiết bị sau NAT/router vẫn kết nối được (peer-to-peer).
- **Đa nền tảng:** Windows, Linux, macOS, Android, iOS, thậm chí cả router, VM, Docker.
- **Giống LAN thật:** các máy kết nối xong có thể ping, chia sẻ file, chạy game LAN, SSH, RDP như mạng nội bộ.
- **Quản lý tập trung:** có web console để quản lý các node, quyền truy cập.
- **Mã hóa end-to-end:** an toàn, riêng tư.

- **Free cho cá nhân/nhóm nhỏ:** bản miễn phí cho phép tạo 1 mạng ảo với tối đa 25 node.

## 2.6. Cài đặt Zerotier

- Download [Zerotier](https://www.zerotier.com/download/) tại <https://www.zerotier.com/download/>
- Cài đặt (next all)
- Nhược điểm: tối đa 25 node (khá ít, vẫn đủ dùng)
- Cấu hình Zerotier:
  - + Trên laptop, tạo Zerotier network bằng cách link với google account
  - + Trên máy công ty, Zerotier cho join vào network đã tạo ở bước trên
  - + Trên mạng này gán ip cho máy công ty, ví dụ: **10.10.10.2**, khi đó từ laptop tạo shortcut để truy cập vào máy công ty:  
`"C:\Program Files\PuTTY\putty.exe" admin@10.10.10.2 -pw 12345678`



### 3. Cài đặt Node.js

#### 3.1. Node.js là gì?

Node.js là **môi trường chạy JavaScript phía server** (server-side runtime), xây dựng trên **V8 Engine** của Google Chrome.

Thay vì chỉ dùng JS trong trình duyệt, Node.js cho phép viết **ứng dụng backend, API, server, IoT, real-time** bằng JavaScript.

#### 3.2. Điểm mạnh của Node.js

- **Non-blocking I/O, event-driven**: xử lý song song hàng ngàn request cùng lúc, rất hợp real-time (chat, game, IoT, API).
- **Một ngôn ngữ từ client → server**: chỉ dùng JavaScript cho cả frontend và backend.
- **Hệ sinh thái npm khổng lồ**: hàng triệu thư viện sẵn có.
- **Cross-platform**: chạy trên Windows, Linux, macOS.
- **Nhanh**: dựa trên V8 Engine, tối ưu tốt.

#### 3.3. Nhược điểm của Node.js

- **Single-threaded**: khó xử lý tác vụ CPU nặng (AI, tính toán khoa học).
- **Callback hell**: dễ rối nếu code bất đồng bộ không quản lý tốt (dù async/await đã cải thiện).
- **Không mạnh bằng Java/Python trong xử lý số liệu lớn**.
- **Bảo mật**: phụ thuộc nhiều vào thư viện npm bên ngoài, dễ gặp lỗ hổng.

#### 3.4. Các bước cài đặt Node.js trên windows

- Cài đặt trên máy công ty
- Download **Node.js** , tại <https://nodejs.org/en/download>
- Nên chọn phiên bản LTS (long-term support: hỗ trợ lâu dài)
- Từ laptop (máy cá nhân, ở nhà) remote qua ssh bằng putty vào máy công ty, để download và cài đặt sử dụng lệnh sau:

+ Download file cài đặt nodejs về máy cty, kết quả được file **nodejs.msi**:

Laptop chạy putty kết nối tới máy công ty, rồi nhập lệnh:

```
curl https://nodejs.org/dist/v22.19.0/node-v22.19.0-x64.msi -o nodejs.msi
```

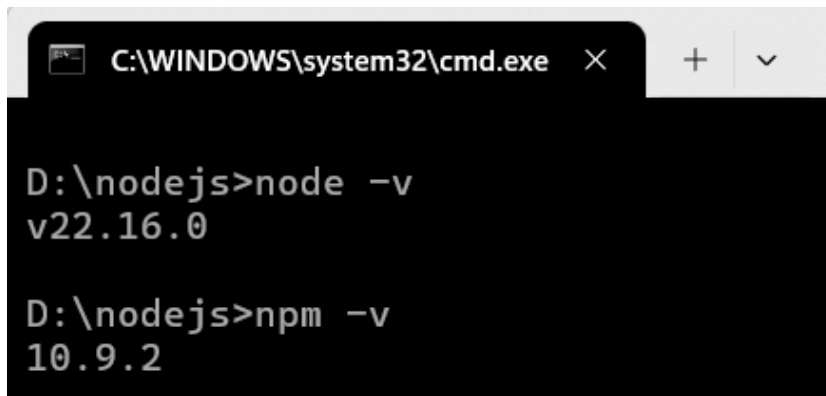
+ Để cài đặt nodejs vào thư mục **d:\nodejs** trên máy công ty, từ putty chạy lệnh:

```
msiexec /i nodejs.msi INSTALLDIR="D:\nodejs" /qn
```

ý nghĩa tham số:

/qn: để cài đặt im lặng.

INSTALLDIR là thư mục cài đặt, nên để trong nodejs



```
C:\WINDOWS\system32\cmd.exe X + v
D:\nodejs>node -v
v22.16.0
D:\nodejs>npm -v
10.9.2
```

## 4. Cài đặt Node-RED:

### 4.1. Node-RED là gì?

Node-RED là **một công cụ lập trình trực quan (low-code)** do IBM phát triển, chạy trên **Node.js**.

Nó dùng giao diện kéo-thả trên trình duyệt để **kết nối các “node”** (khối chức năng) thành một “flow” (luồng xử lý dữ liệu).

### 4.2. Đặc trưng của Node-RED

- **Flow-based programming:** thay vì viết code dài dòng, chỉ cần kéo-thả các node (Input, Output, Function, Debug...).
- **IoT-friendly:** hỗ trợ sẵn MQTT, HTTP, WebSocket, Modbus, Database...
- **Mở rộng dễ dàng:** có hơn 4000 node đóng góp từ cộng đồng.
- **Web UI trực quan:** chạy trên browser, chỉnh sửa và deploy ngay.
- **Cross-platform:** chạy trên PC, server, Raspberry Pi, Docker.

### 4.3. Cài đặt Node-RED

- Cài đặt trên máy công ty
- Nodered sẽ thu thập qua rs485 (LAN) => lưu db giá trị tức thời và lịch sử
- Putty đang SSH vào máy tính công ty:
  - + Vào thư mục D:\nodejs : gõ 2 lệnh sau  
d:  
cd d:\nodejs
  - + Cài đặt nodered bằng lệnh sau:  
npm install -g --unsafe-perm node-red --prefix "D:\nodejs\nodered"

### 4.4. Cấu hình để nodered luôn tự động chạy cùng windows

- Thực hiện trên laptop, sau đó sẽ copy sang máy công ty
- Download công cụ [nssm](#), công cụ này giúp biến 1 ứng dụng thành dạng service, mà service có thể tự động chạy khi windows khởi động, thậm chí chạy luôn trước cả khi user login, không cần login vẫn chạy.

- Sử dụng công cụ [WinSCP](#) để copy file từ laptop vào máy tính công ty thông qua giao thức SFTP mà OpenSSH hỗ trợ

- Giải nén ra file nssm.exe, dùng [WinSCP](#) để copy file nssm.exe vào path trên máy công ty: **d:\nodejs\nodered\nssm.exe**

- dùng [WinSCP](#) để edit file **d:\nodejs\nodered\node-red.cmd** với nội dung:

```
@echo off
REM fix path
set PATH=D:\nodejs;%PATH%
REM Run Node-RED
node "D:\nodejs\nodered\node_modules\node-red\red.js" -u "D:\nodejs\nodered\work" %*
```

- Chạy lệnh từ putty để tạo service:

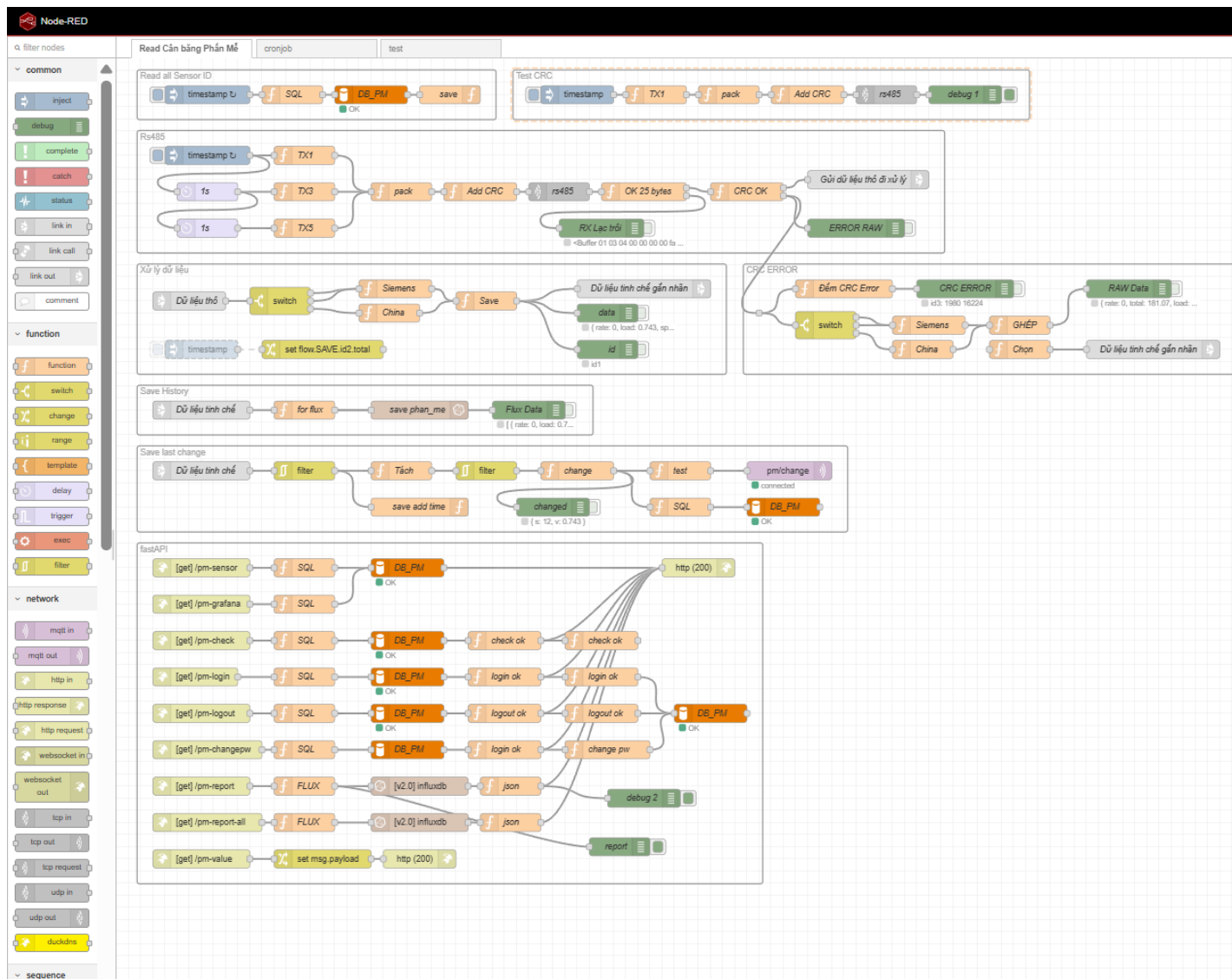
d:

```
cd d:\nodejs\nodered
```

```
nssm.exe install al-nodered "D:\nodejs\nodered\node-red.cmd"
```

- Chạy service, từ putty gõ lệnh:

```
sc start al-nodered
```



## 4.5. Kiểm tra Node-RED có đang chạy ?

- Thực hiện trên máy laptop
- Mở trình duyệt, nhập url: `http://<ip tailscale>:1880`
- Nếu vào được nodered là OK

## 4.6. Cấu hình Nodered

- Cấu hình Nodered để yêu cầu đăng nhập trước khi sử dụng:

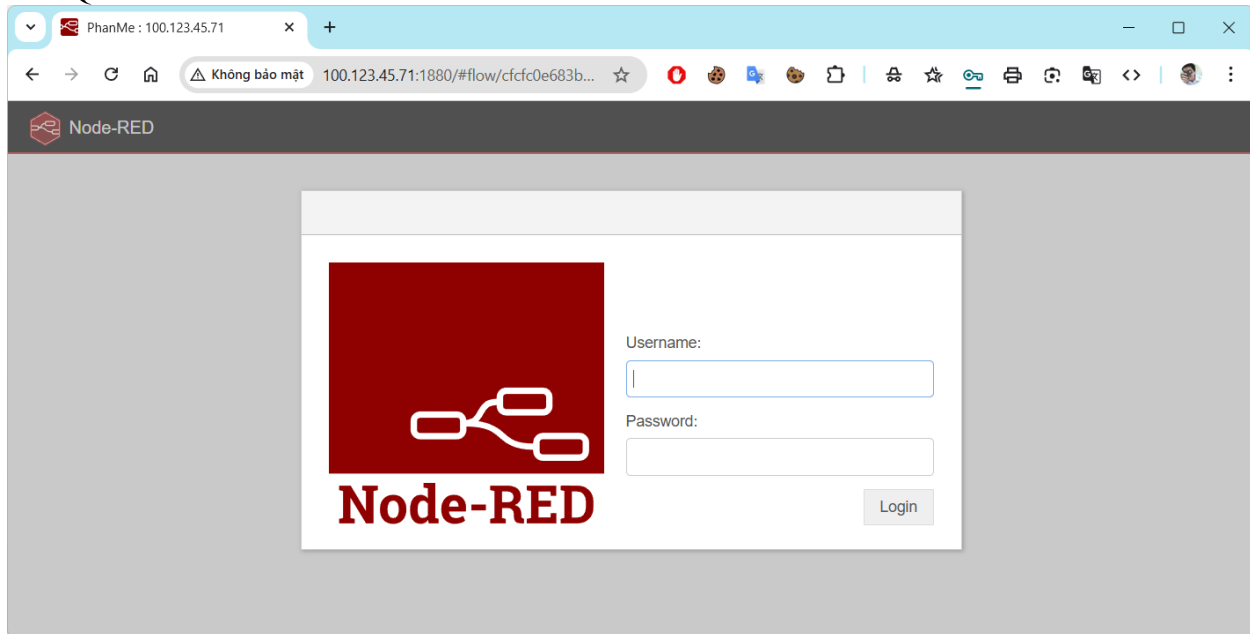
Sử dụng công cụ [WinSCP](#) để edit file: `d:\nodejs\nodered\work\settings.js`

Tìm đến **adminAuth**: bỏ các comment `//` ở đầu dòng, để thiết lập mật khẩu sử dụng [tool](#) này đã mã hoá, lấy được chuỗi mã hoá thay vào trường password

Ví dụ:

```
adminAuth: {  
  type: "credentials",  
  users: [{  
    username: "admin",  
    password: "$1y$23$4KbU5I67l8df9iYYdlvdru2AwTG/z3vc4TgM5Exo6sRDs7hMy8Dya",  
    permissions: "*"   
  }]  
},
```

KẾT QUẢ:



## 5. Cài đặt MariaDB

### 5.1. MariaDB là gì?

MariaDB là **hệ quản trị cơ sở dữ liệu quan hệ (RDBMS)**, phát triển từ **MySQL** (do chính tác giả MySQL tách ra sau khi Oracle mua Sun). Nó **hoàn toàn open source** và tương thích ngược với MySQL.

## 5.2. Đặc trưng của MariaDB

- **Tương thích MySQL:** hầu hết cú pháp, API, client của MySQL chạy được trên MariaDB.
- **Mã nguồn mở 100%:** không lo vấn đề giấy phép thương mại như MySQL.
- **Hiệu năng cao:** tối ưu query, storage engine tốt hơn (ví dụ Aria, XtraDB).
- **Bảo mật tốt:** có thêm tính năng như PAM, LDAP authentication plugin.
- **Replication:** hỗ trợ master-slave, Galera Cluster cho HA (high availability).
- **Ổn định lâu dài:** được cộng đồng + MariaDB Foundation duy trì.

## 5.3. Các bước cài đặt MariaDB trên Windows

### 1. Tải về

- Vào <https://mariadb.org/download/>
- Chọn Windows (installer .msi).

### 2. Chạy installer

- Nhấn Next → Accept → chọn **Typical Setup**.
- Đặt password cho tài khoản root.
- Chọn Install as Service để MariaDB chạy ngầm (service tự khởi động).

### 3. Hoàn tất cài đặt

- Sau khi cài, MariaDB service sẽ chạy (port mặc định **3306**).
- Kiểm tra bằng CMD:
- `mysql -u root -p`

### 4. Cấu hình thêm (tùy chọn)

- File cấu hình chính: `my.ini` hoặc `my.cnf` trong thư mục cài đặt.
- Có thể bật log query, tối ưu buffer pool, chỉnh port.

## 5.4. Các công cụ quản trị MariaDB

- **mysql CLI** (dùng sẵn): chạy query trực tiếp trong terminal.
- **HeidiSQL:** gọn nhẹ, dễ dùng, miễn phí.
- **DBeaver:** mạnh, đa DB, hỗ trợ cả MariaDB, PostgreSQL, Oracle...
- **phpMyAdmin:** web-based, tiện quản trị qua browser.
- **Navicat / DataGrip (JetBrains):** thương mại, nhiều tính năng nâng cao.

## 5.5. Áp dụng:

Download phiên bản mới nhất tại <https://mariadb.com/downloads/>

Tại thời điểm document này thì bản mới nhất là [MariaDB version 12.0.2](#)

Chú ý: Vì MariaDB lưu trữ dữ liệu vào thư mục cài đặt, để an toàn nên cài đặt vào ổ D, nếu cài lại windows thì không bị mất dữ liệu.



```
Administrator: C:\Windows\system32\conhost.exe - mysql -u root -p

admin@DESKTOP-GKTRJ7R D:\mariadb\bin>mysql -u root -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 69
Server version: 12.0.2-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use phan_me
Database changed
MariaDB [phan_me]> select * from sensor;
+----+-----+-----+-----+-----+-----+-----+
| id | tid | name  | unit  | value | time                | note |
+----+-----+-----+-----+-----+-----+-----+
| 11 | 1   | Rate  | Tần/h | 0      | 2025-09-24 18:46:26 | NULL |
| 12 | 1   | Load | kg/m   | 0.76   | 2025-09-25 09:01:07 | NULL |
| 13 | 1   | Speed | m/s    | 0      | 2025-09-24 18:46:26 | NULL |
| 14 | 1   | Total | Tần    | 454385.03 | 2025-09-24 18:44:44 | NULL |
| 15 | 1   | Total2 | Tần    | 4544011.46 | 2025-09-24 18:44:50 | NULL |
| 21 | 2   | Rate  | Tần/h | 0      | 2025-09-23 10:46:11 | NULL |
| 22 | 2   | Load | kg/m   | 0.8     | 2025-09-25 09:01:03 | NULL |
| 23 | 2   | Speed | m/s    | 1       | 2025-09-13 00:31:48 | NULL |
| 24 | 2   | Total | Tần    | 181.07  | 2025-09-23 10:45:49 | NULL |
| 25 | 2   | Total2 | Tần    | 181.07  | 2025-09-23 10:45:49 | NULL |
| 31 | 3   | Rate  | Tần/h | 0      | 2025-09-25 08:01:35 | NULL |
| 32 | 3   | Load | kg/m   | -1.155  | 2025-09-25 09:01:02 | NULL |
| 33 | 3   | Speed | m/s    | 0       | 2025-09-25 08:01:35 | NULL |
| 34 | 3   | Total | Tần    | 121629.1 | 2025-09-24 18:46:03 | NULL |
| 35 | 3   | Total2 | Tần    | 121932.3 | 2025-09-24 18:46:03 | NULL |
+----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.001 sec)

MariaDB [phan_me]>
```

## 5.6. Tổ chức dữ liệu cho Web IOT

Sử dụng công cụ **HeidiSQL** hoặc **phpMyAdmin** để quản trị.

Mục tiêu:

- Tạo csdl mới (đặt tên liên quan đến dự án)
- Tạo các bảng để:
  - o Phục vụ việc login
  - o Phục vụ lưu log (log đăng nhập, log các lỗi phát sinh, ...)
  - o Lưu trữ dữ liệu tức thời của các cảm biến
  - o Lưu trữ id của biểu đồ trực quan hoá dữ liệu trên grafana

Với bài toán cụ thể này, cần các bảng:



STT	Tên bảng	Ý nghĩa	Các trường
1	Sensor	Lưu thông tin cảm biến	(#id, name, unit, value, time)
2	Login	Lưu thông tin đăng nhập	(#id, uid, pwd, name, role)
3	Log	Lưu thông tin Log	(#id, tag, message)
4	Grafana	Lưu gid các biểu đồ	(#id, gid, ids, name)
5	Cookie	Lưu ck phiên đăng nhập	(#id, uid, ck, time)

Với bảng Sensor: Số dòng dữ liệu được insert vào ngay từ đầu, sau đó ko thay đổi số lượng. Khi sensor có dữ liệu mới chỉ dùng lệnh Update để cập nhật VALUE và TIME.

The screenshot shows the HeidiSQL interface with the 'sensor' table selected. The table structure is as follows:

#	1 id	2 tid	3 name	4 unit	5 value	6 time	7 note
1	11	1	Rate	Tấn/h	0	2025-09-24 18:46:26	(NULL)
2	12	1	Load	kg/m	0.739	2025-09-25 09:04:25	(NULL)
3	13	1	Speed	m/s	0	2025-09-24 18:46:26	(NULL)
4	14	1	Total	Tấn	454,385.03	2025-09-24 18:44:44	(NULL)
5	15	1	Total2	Tấn	4,544,011.46	2025-09-24 18:44:50	(NULL)
6	21	2	Rate	Tấn/h	0	2025-09-23 10:46:11	(NULL)
7	22	2	Load	kg/m	0.81	2025-09-25 09:04:24	(NULL)
8	23	2	Speed	m/s	1	2025-09-13 00:31:48	(NULL)
9	24	2	Total	Tấn	181.07	2025-09-23 10:45:49	(NULL)
10	25	2	Total2	Tấn	181.07	2025-09-23 10:45:49	(NULL)
11	31	3	Rate	Tấn/h	0	2025-09-25 08:01:35	(NULL)
12	32	3	Load	kg/m	-1.149	2025-09-25 09:03:17	(NULL)
13	33	3	Speed	m/s	0	2025-09-25 08:01:35	(NULL)
14	34	3	Total	Tấn	121,629.1	2025-09-24 18:46:03	(NULL)
15	35	3	Total2	Tấn	121,932.3	2025-09-24 18:46:03	(NULL)

## 5.7. Tạo database và các table bằng tool HeidiSQL

```
CREATE DATABASE IF NOT EXISTS `phan_me`;
USE `phan_me`;

-- Dumping structure for table phan_me.grafana
CREATE TABLE IF NOT EXISTS `grafana` (
  `id` int(11) NOT NULL,
  `gid` varchar(255) NOT NULL,
  `ids` varchar(255) NOT NULL,
  `name` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- Dumping structure for table phan_me.log
CREATE TABLE IF NOT EXISTS `log` (
  `ID` int(11) NOT NULL AUTO_INCREMENT,
```

```

`key` longtext DEFAULT NULL,
`message` longtext DEFAULT NULL,
`time` datetime DEFAULT current_timestamp(),
PRIMARY KEY (`ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- Dumping structure for table phan_me.sensor
CREATE TABLE IF NOT EXISTS `sensor` (
  `id` int(11) NOT NULL,
  `tid` int(11) DEFAULT NULL,
  `name` varchar(50) DEFAULT NULL,
  `unit` varchar(10) DEFAULT NULL,
  `value` double DEFAULT NULL,
  `time` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- Dumping structure for table phan_me.users
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `uid` varchar(17) DEFAULT NULL,
  `pwd` varchar(50) DEFAULT NULL,
  `name` varchar(50) DEFAULT NULL,
  `role` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uid` (`uid`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- Dumping structure for table phan_me.cookie
CREATE TABLE IF NOT EXISTS `cookie` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `uid` varchar(50) DEFAULT NULL,
  `ck` varchar(50) DEFAULT NULL,
  `time` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

```

## 6. Cài đặt influxDB 2.7

### 6.1. InfluxDB là gì?

InfluxDB là **cơ sở dữ liệu dạng time-series (chuỗi thời gian)**, tối ưu cho việc lưu trữ và truy vấn dữ liệu có dấu thời gian (timestamp). Nó thường được dùng trong **IoT, monitoring, logging, metrics** (kiểu như dữ liệu sensor, log hệ thống, performance server).

### 6.2. Đặc trưng của InfluxDB

- **Time-series DB:** mọi bản ghi đều gắn timestamp.
- **Không cần schema:** tạo measurement (giống table) rồi insert dữ liệu trực tiếp.
- **Query mạnh mẽ:** dùng InfluxQL hoặc Flux để truy vấn (kiểu SQL nhưng tối ưu cho time-series).
- **Retention Policy:** có thể tự động xóa dữ liệu cũ, tiết kiệm dung lượng.
- **Hiệu năng cao:** ghi được hàng triệu points/giây, đọc nhanh.

### 6.3. Các bước cài đặt InfluxDB trên Windows

#### 1. Tải về

- Vào trang chính thức: <https://portal.influxdata.com/downloads/>
- Chọn bản **InfluxDB 2.x for Windows** (file .zip).

#### 2. Giải nén

- Giải nén vào thư mục, ví dụ: D:\influxdb.

#### 3. Chạy server lần đầu

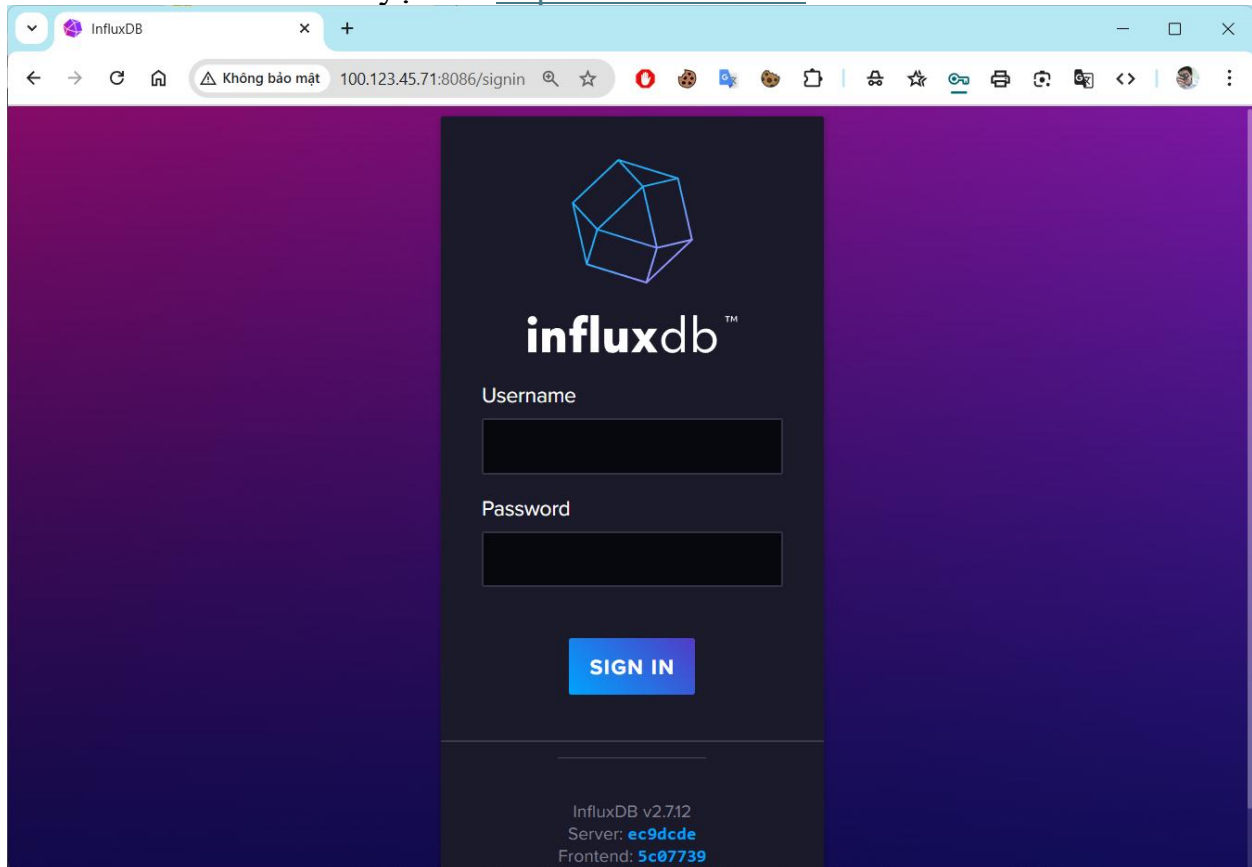
- Mở **PowerShell** hoặc **CMD** trong thư mục vừa giải nén.
- Chạy lệnh: `influxd.exe`
- Server sẽ khởi động tại <http://localhost:8086>.

#### 4. Tạo service tự động chạy với nssm

- Tạo file **setup\_influxdb.bat** với nội dung (3 dòng):  
`nssm install A1-influxdb "D:\influxdb\influxd.exe"`  
`nssm set A1-influxdb AppDirectory "D:\influxdb"`  
`nssm set A1-influxdb Description "A1-influxdb"`

#### 5. Khởi tạo (setup)

- Mở trình duyệt vào <http://localhost:8086>.



- Tạo:
  - username, password
  - organization name

- bucket name (bucket = database + retention policy).

## 6. Cài đặt PATH (tùy chọn)

- Thêm thư mục influxdb vào biến môi trường PATH để có thể dùng influx ở mọi nơi.

## 7. Chạy client CLI

- Dùng lệnh:
- influx.exe
- Đăng nhập bằng token vừa tạo.

## 8. (Tùy chọn) Cài đặt như service

- Có thể dùng nssm.exe để chạy influxd.exe như Windows Service (tự chạy khi khởi động máy).

## 6.4. InfluxDB-CLI là gì?

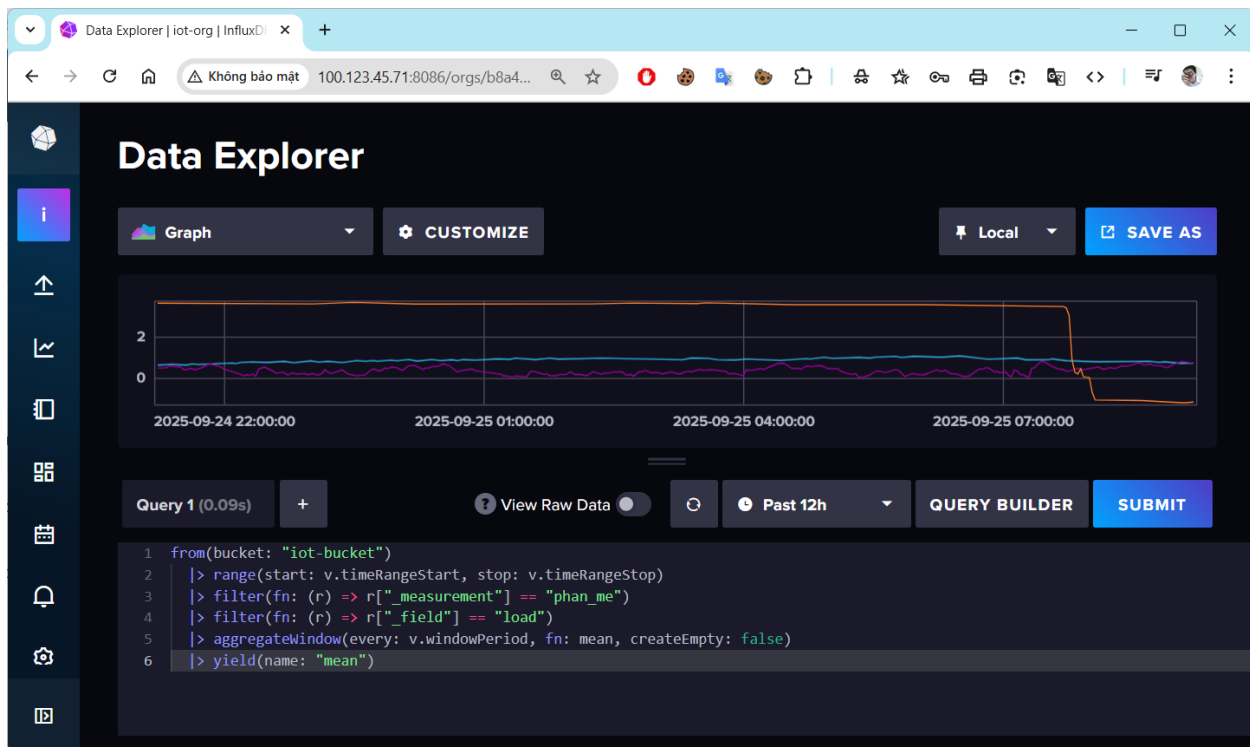
InfluxDB-CLI là **công cụ dòng lệnh (Command Line Interface)** để làm việc với **InfluxDB 2.x**.

Nó cho phép:

- Quản lý **organization, bucket, user, token**.
- Insert dữ liệu trực tiếp (write).
- Query dữ liệu bằng **Flux** hoặc InfluxQL.
- Backup/restore dữ liệu.

Nói nôm na: **influx-cli** là client nói chuyện với **influxd** (server).





## 6.5. Cài đặt InfluxDB-CLI trên Windows

### 1. Tải CLI

- Vào <https://portal.influxdata.com/downloads/>.
- Kéo xuống mục **InfluxDB CLI (influx)** → chọn Windows → tải file .zip.

### 2. Giải nén

- Ví dụ giải nén vào D:\influxdb-cli.

### 3. Cách dùng: (tự tìm hiểu)

## 7. Cài đặt Grafana

### 7.1. Grafana là gì?

Grafana là **nền tảng trực quan hóa dữ liệu mã nguồn mở**. Nó kết nối đến nhiều nguồn dữ liệu khác nhau (InfluxDB, Prometheus, MySQL, PostgreSQL, Elasticsearch...) và hiển thị dưới dạng **dashboard, chart, alert**.

### 7.2. Tác dụng của Grafana

- Biến dữ liệu khô khan thành **biểu đồ, dashboard đẹp mắt**.
- Theo dõi hệ thống, IoT, server, app real-time.
- Gửi cảnh báo (alert) qua email, Telegram, Slack khi vượt ngưỡng.
- Kết hợp nhiều nguồn dữ liệu trên cùng một dashboard.

### 7.3. Ưu điểm

- **Đa dạng nguồn dữ liệu:** hầu hết DB phổ biến đều kết nối được.
- **Dashboard mạnh mẽ:** kéo-thả, custom linh hoạt.
- **Alerting:** cảnh báo real-time.
- **Mã nguồn mở + miễn phí,** có bản Enterprise nếu cần tính năng nâng cao.
- **Cộng đồng lớn,** nhiều plugin có sẵn.

### 7.4. Nhược điểm

- **Chỉ trực quan hóa, không phải DB** → vẫn cần cơ sở dữ liệu lưu trữ (InfluxDB, Prometheus...).
- **Hơi nặng** nếu chạy trên Windows yếu.
- **Cấu hình alert nâng cao** đôi khi phức tạp.
- **Không phải ETL tool** (không giỏi xử lý dữ liệu, chỉ hiển thị).

### 7.5. Cách cài đặt Grafana trên Windows

#### 1. Tải về

- Vào <https://grafana.com/grafana/download>.
- Chọn **Windows installer (.msi)**.

#### 2. Cài đặt

- Chạy file .msi → Next → Install.
- Grafana sẽ cài thành **Windows service**.

#### 3. Khởi động

- Sau khi cài, Grafana service tự chạy (port mặc định **3000**).
- Mở trình duyệt: <http://localhost:3000>.

#### 4. Login lần đầu

- Tài khoản mặc định:
  - **user:** admin
  - **password:** admin (sẽ bắt đổi pass).

#### 5. Thêm Data Source

- Vào **Configuration** → **Data Sources** → chọn InfluxDB, MySQL, Prometheus...
- Nhập thông tin kết nối DB.

#### 6. Tạo Dashboard

- Vào **Create** → **Dashboard** → Add Panel → chọn loại biểu đồ (time-series, gauge, pie chart...).

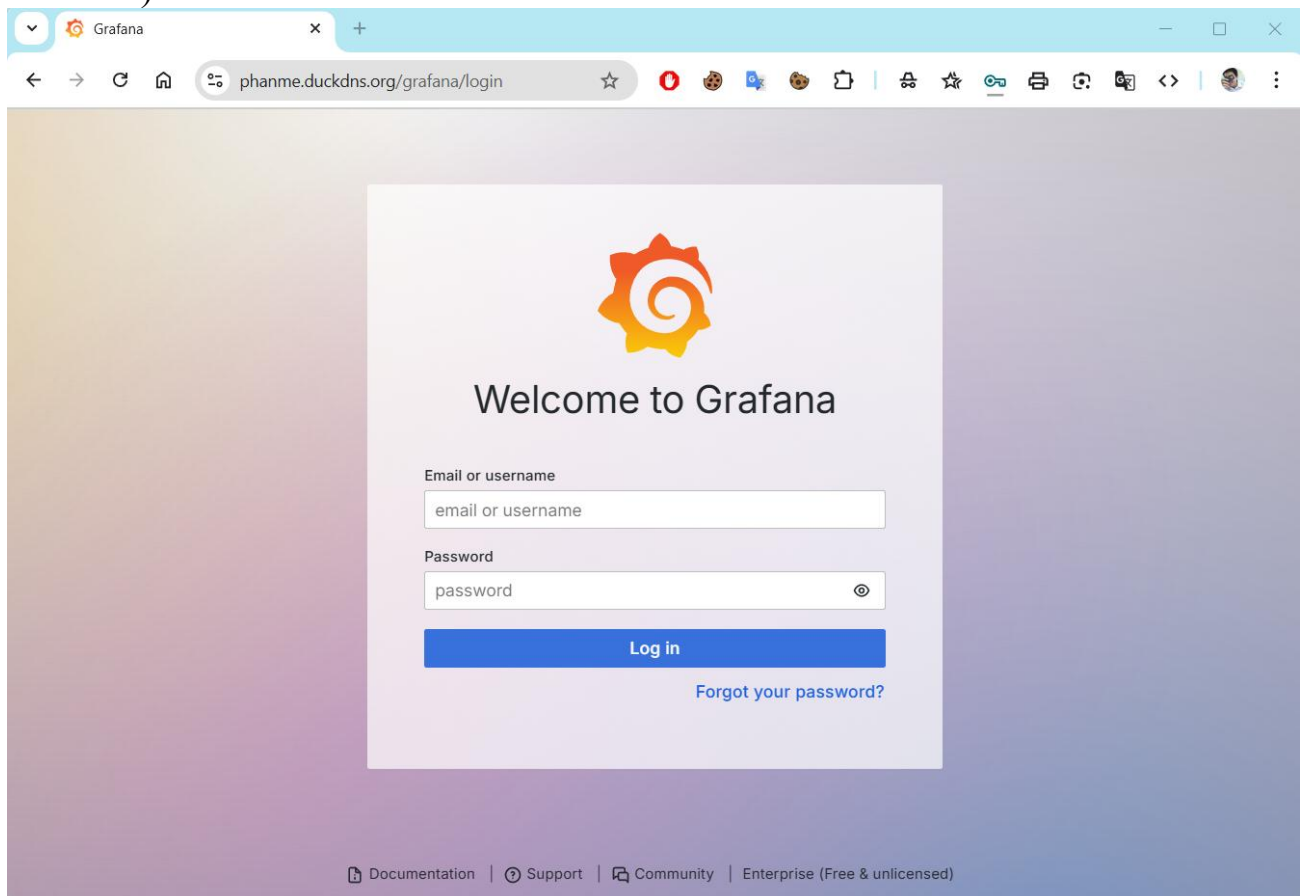
## 7.6. Cấu hình để chạy grafana qua sub path

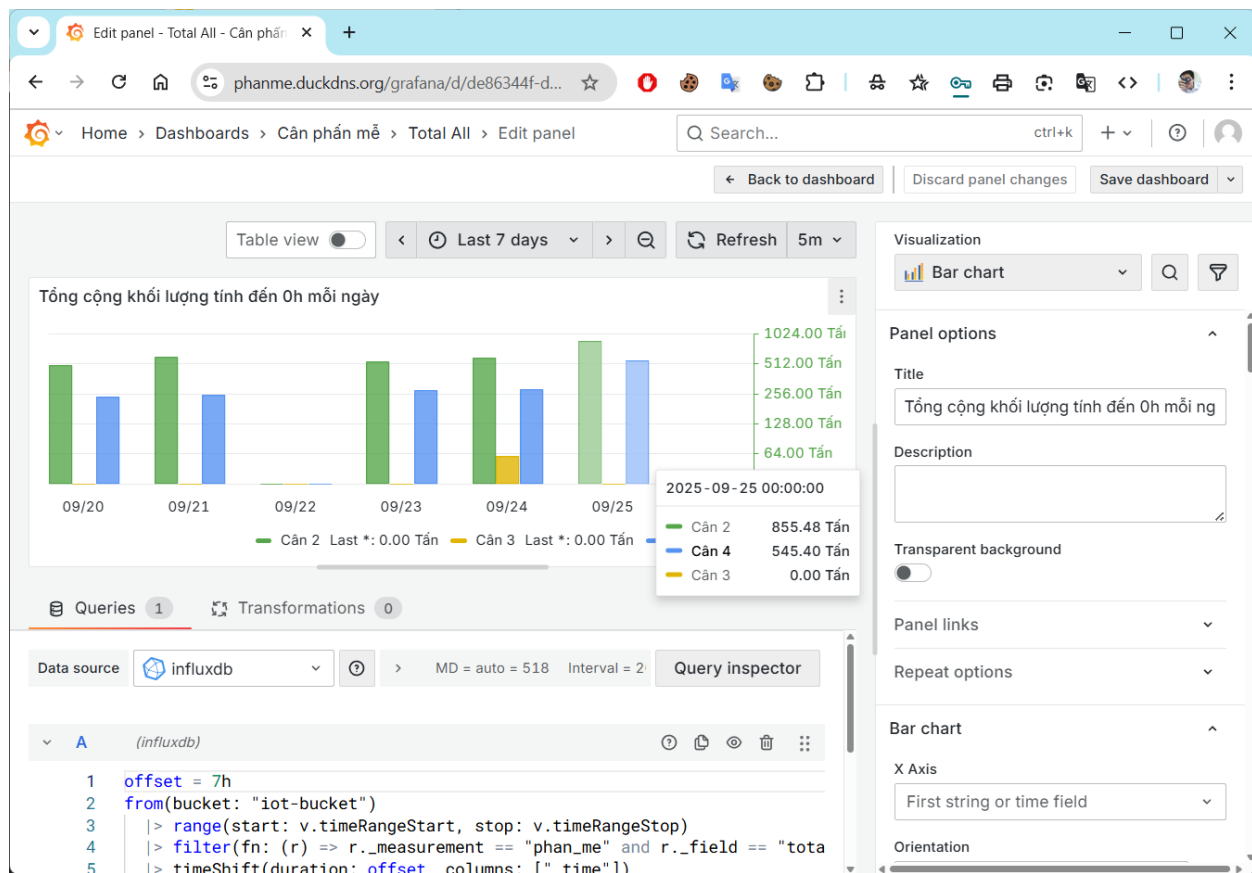
- Chỉnh file grafana.ini:

```
[server]
# port mặc định là 3000
http_port = 3000

# URL công khai của Grafana (subdomain)
domain = grafana.example.com
root_url = %(protocol)s://%(domain)s/
# nếu chạy sau reverse proxy thì bật
serve_from_sub_path = true
```

- Dùng Reverse Proxy (khuyến nghị): tùy webserver iis,nginx, apache (tự tìm hiểu)





## 8. Cài đặt Emqx : MQTT

### 8.1. MQTT là gì?

- **MQTT (Message Queuing Telemetry Transport)** là giao thức nhắn tin **publish/subscribe** cực nhẹ, tối ưu cho IoT.
- Dữ liệu gửi đi theo **topic**, client nào subscribe topic thì sẽ nhận dữ liệu.
- Rất tiết kiệm băng thông, dùng được cả trong môi trường mạng yếu, mất kết nối tạm thời.

### 8.2. EMQX là gì?

- **EMQX (Erlang MQTT Broker)** là **máy chủ (broker)** triển khai giao thức MQTT.
- Nó đóng vai trò trung gian: nhận dữ liệu từ **publisher** (sensor, thiết bị IoT) và phân phát cho **subscriber** (ứng dụng, client).

### 8.3. Đặc trưng của EMQX

- **Hỗ trợ chuẩn MQTT 3.1.1 và 5.0.**
- **Hiệu năng cao:** hàng triệu kết nối đồng thời.
- **Cluster:** có thể scale ngang nhiều node.



- **Tích hợp nhiều giao thức:** MQTT, WebSocket, HTTP API, CoAP, gRPC.
- **Quản trị web UI:** giám sát kết nối, topic, message.
- **Extensible:** plugin authentication, rule engine, bridge qua Kafka, RabbitMQ, InfluxDB, MySQL...

#### 8.4. Điểm mạnh

- **Cực kỳ mạnh về scale** (hàng triệu client).
- **Giao diện quản lý Web UI đẹp.**
- **Plugin phong phú** (auth, DB, bridge).
- **Open-source, active community.**

#### 8.5. Điểm yếu

- **Nặng** hơn so với Mosquitto (do nhiều tính năng).
- **Tốn RAM/CPU** hơn nếu chỉ chạy vài trăm sensor.
- Bản **Enterprise** mới có đủ tính năng cao cấp (HA cluster, observability full).

#### 8.6. Cài đặt EMQX trên Windows

##### 1. Tải về

- Vào <https://www.emqx.io/downloads>.
- Chọn bản **Windows (zip)**.

##### 2. Giải nén

- Ví dụ giải nén vào D:\emqx.

##### 3. Chạy EMQX

- Mở **PowerShell** hoặc **CMD** trong thư mục: `emqx start`
- Kiểm tra trạng thái: `emqx ctl status`
- Cài đặt để Emqx tự động chạy: `emqx install`

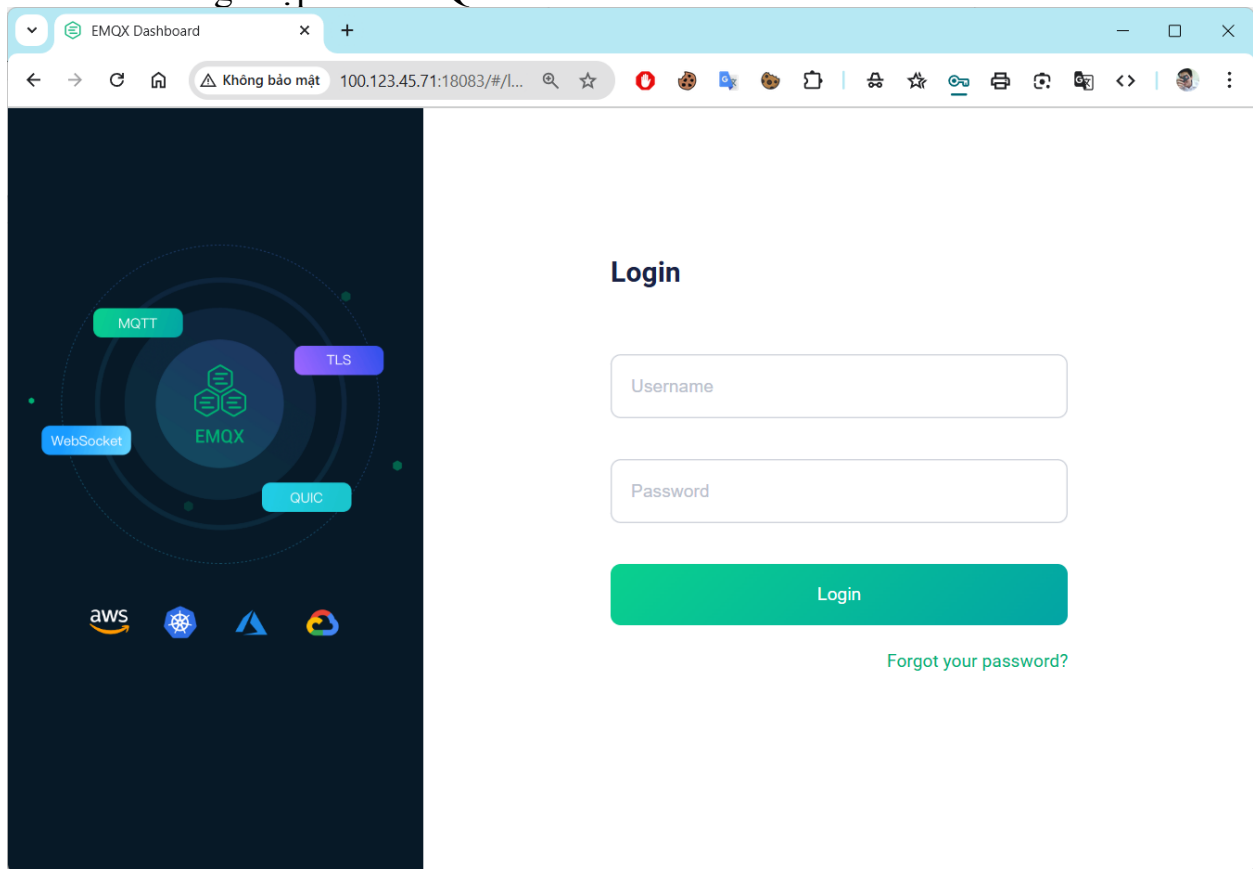
##### 4. Truy cập Dashboard

- Mở trình duyệt: <http://localhost:18083>.
- Login mặc định:
  - user: admin
  - pass: public

##### 5. Test MQTT

- Cài MQTT client (ví dụ **MQTTX**, **MQ Explorer**, hoặc **mosquitto\_pub/sub**).
- Publish thử: `mosquitto_pub -h localhost -p 1883 -t test/topic -m "Hello IoT"`
- Subscribe thử: `mosquitto_sub -h localhost -p 1883 -t test/topic`

## Màn hình đăng nhập vào EMQX



EMQX Dashboard

Không bảo mật 100.123.45.71:18083/#/l...

### Login

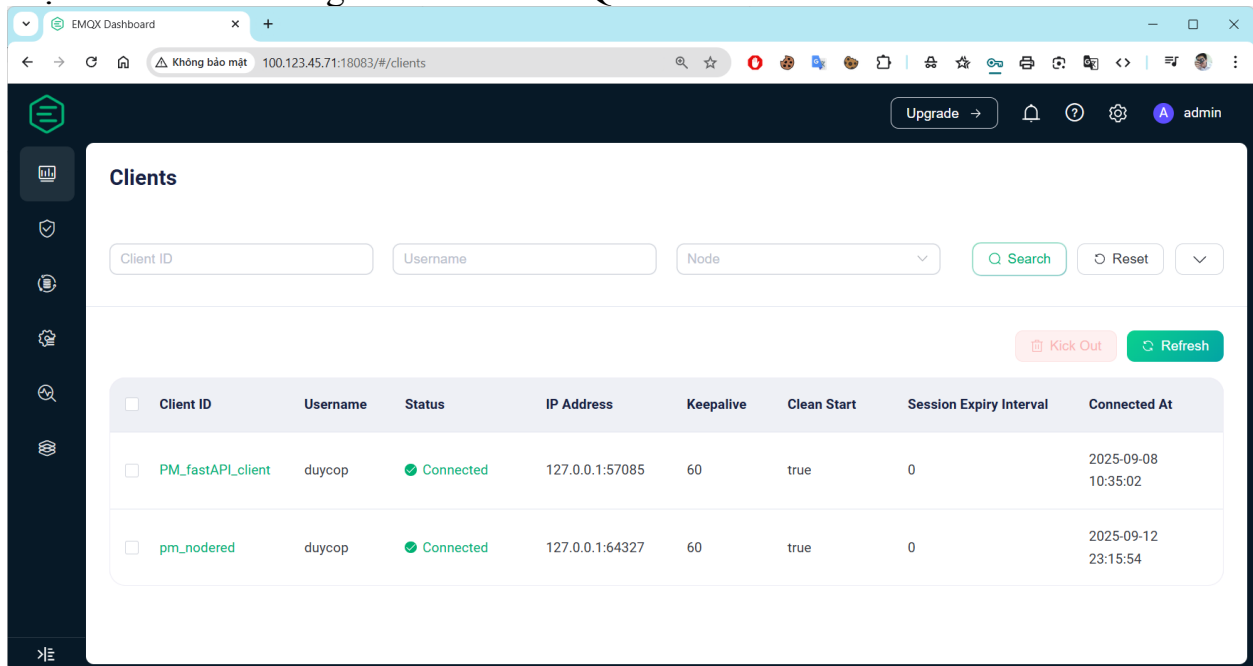
Username

Password

Login

[Forgot your password?](#)

## Liệt kê các client đang kết nối vào EMQX



EMQX Dashboard

Không bảo mật 100.123.45.71:18083/#/clients

Upgrade →

### Clients

Client ID Username Node Search Reset

Kick Out Refresh

Client ID	Username	Status	IP Address	Keepalive	Clean Start	Session Expiry Interval	Connected At
<input type="checkbox"/> PM_fastAPI_client	duycop	Connected	127.0.0.1:57085	60	true	0	2025-09-08 10:35:02
<input type="checkbox"/> pm_nodered	duycop	Connected	127.0.0.1:64327	60	true	0	2025-09-12 23:15:54

## 9. Cài đặt Python và các thư viện cần thiết

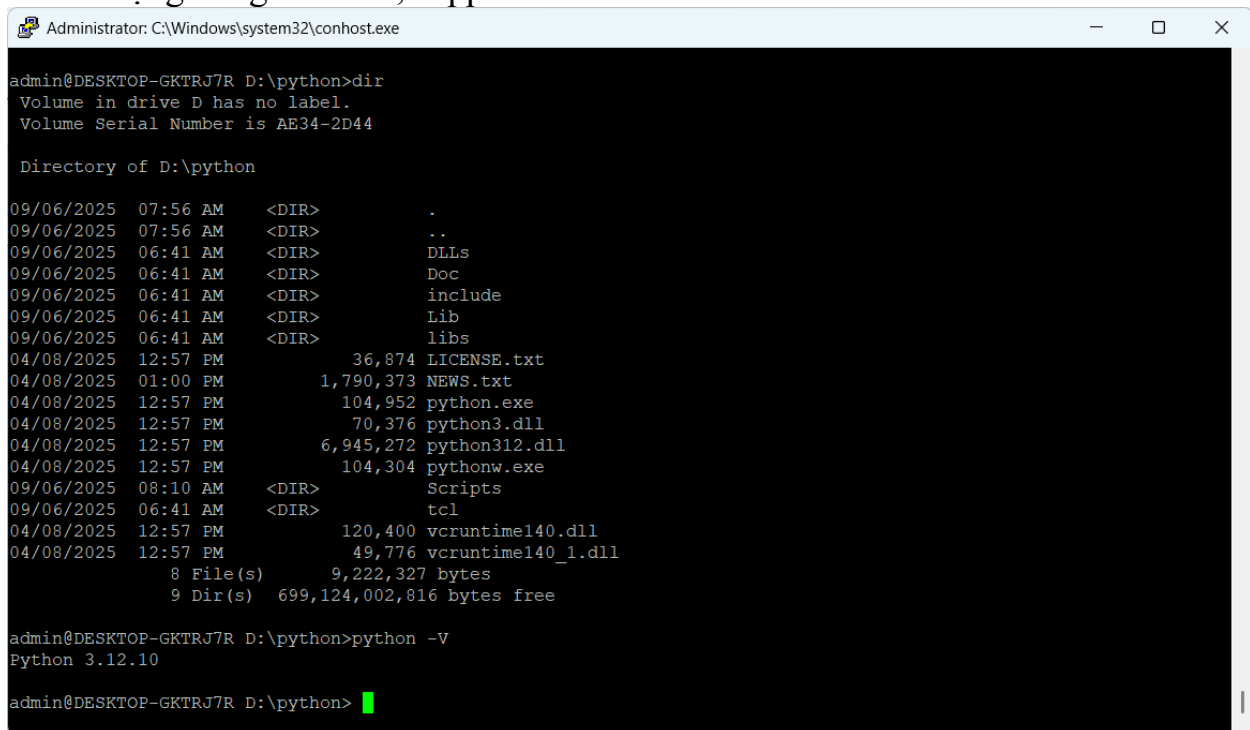
### 9.1. Python là gì?

Python là một **ngôn ngữ lập trình thông dịch, đa năng, dễ học**, nổi tiếng vì cú pháp gọn gàng, đọc như tiếng Anh. Nó được dùng rộng rãi trong:

- Web (Django, FastAPI, Flask)
- AI/ML (TensorFlow, PyTorch)
- Tự động hóa, scripting
- IoT, khoa học dữ liệu

### 9.2. Ưu điểm của Python

- Cú pháp dễ hiểu, học nhanh.
- Thư viện phong phú (AI, Web, Big Data, IoT...).
- Chạy đa nền tảng (Windows, Linux, macOS).
- Cộng đồng siêu lớn, support tốt.



```
Administrator: C:\Windows\system32\conhost.exe
admin@DESKTOP-GKTRJ7R D:\python>dir
Volume in drive D has no label.
Volume Serial Number is AB34-2D44

Directory of D:\python

09/06/2025  07:56 AM    <DIR>          .
09/06/2025  07:56 AM    <DIR>          ..
09/06/2025  06:41 AM    <DIR>          DLLs
09/06/2025  06:41 AM    <DIR>          Doc
09/06/2025  06:41 AM    <DIR>          include
09/06/2025  06:41 AM    <DIR>          Lib
09/06/2025  06:41 AM    <DIR>          libs
04/08/2025  12:57 PM             36,874 LICENSE.txt
04/08/2025  01:00 PM          1,790,373 NEWS.txt
04/08/2025  12:57 PM          104,952 python.exe
04/08/2025  12:57 PM           70,376 python3.dll
04/08/2025  12:57 PM          6,945,272 python312.dll
04/08/2025  12:57 PM          104,304 pythonw.exe
09/06/2025  08:10 AM    <DIR>          Scripts
09/06/2025  06:41 AM    <DIR>          tcl
04/08/2025  12:57 PM          120,400 vcruntime140.dll
04/08/2025  12:57 PM           49,776 vcruntime140_1.dll
               8 File(s)          9,222,327 bytes
               9 Dir(s)  699,124,002,816 bytes free

admin@DESKTOP-GKTRJ7R D:\python>python -V
Python 3.12.10

admin@DESKTOP-GKTRJ7R D:\python>
```

### 9.3. FastAPI là gì?

FastAPI là **framework Python để xây dựng API (RESTful/GraphQL)**, nổi bật nhờ:

- **Nhanh:** hiệu năng gần ngang NodeJS/Go nhờ dùng **ASGI + Starlette + Pydantic**.
- **Tự động sinh tài liệu API** (Swagger UI, ReDoc).
- **Hỗ trợ async/await** → xử lý song song ngon.
- **Type hint rõ ràng** → code an toàn, dễ bảo trì.

## 9.4. Ưu điểm của FastAPI

- Phát triển API nhanh, ít code.
- Tích hợp validation dữ liệu tự động.
- Tài liệu API “auto đẹp” không cần viết tay.
- Hợp cho microservices, IoT backend, AI API.

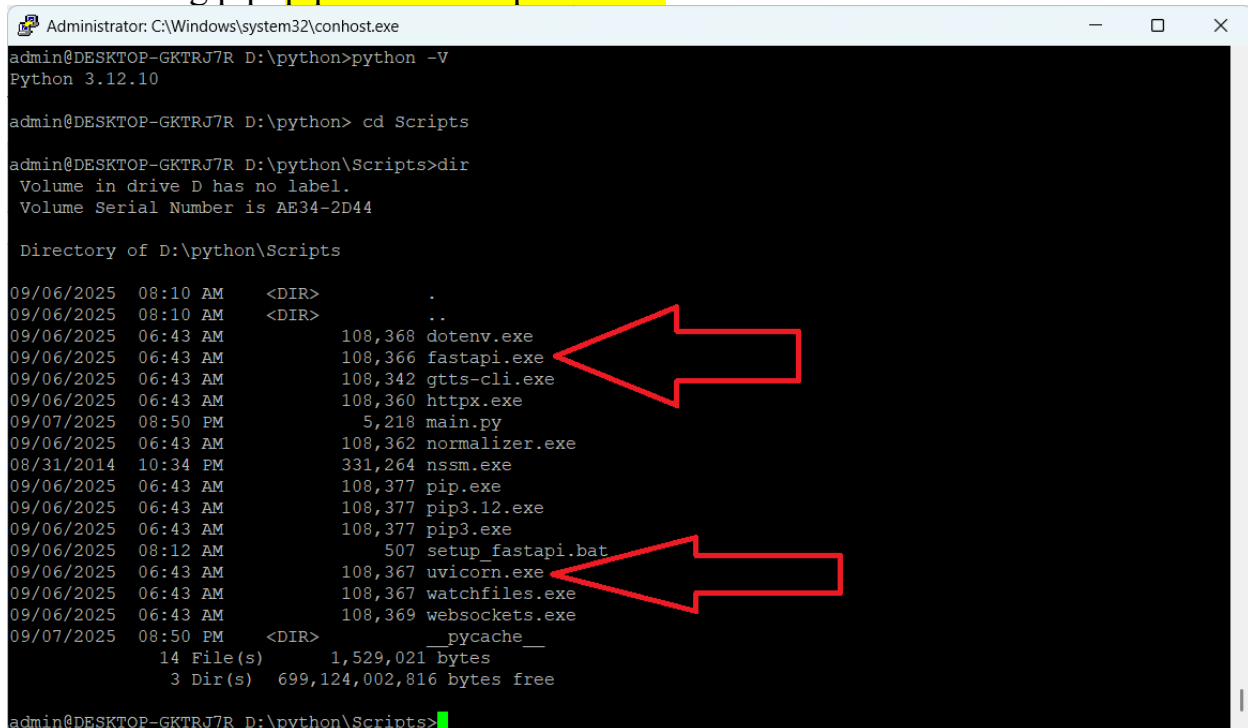
## 9.5. Cài đặt Python trên Windows

1. Tải Python từ [python.org/downloads](https://python.org/downloads).
2. Chạy installer → tick “Add Python to PATH” → Install.
3. Kiểm tra:
4. `python --version`
5. `pip --version`

## 9.6. Cài đặt FastAPI

FastAPI chạy trên ASGI server (Uvicorn).

Cài bằng pip: `pip install fastapi uvicorn`



```
Administrator: C:\Windows\system32\conhost.exe
admin@DESKTOP-GKTRJ7R D:\python>python -V
Python 3.12.10

admin@DESKTOP-GKTRJ7R D:\python> cd Scripts

admin@DESKTOP-GKTRJ7R D:\python\Scripts>dir
Volume in drive D has no label.
Volume Serial Number is AE34-2D44

Directory of D:\python\Scripts

09/06/2025  08:10 AM    <DIR>          .
09/06/2025  08:10 AM    <DIR>          ..
09/06/2025  06:43 AM             108,368 dotenv.exe
09/06/2025  06:43 AM             108,366 fastapi.exe
09/06/2025  06:43 AM             108,342 gtts-cli.exe
09/06/2025  06:43 AM             108,360 httpx.exe
09/07/2025  08:50 PM               5,218 main.py
09/06/2025  06:43 AM             108,362 normalizer.exe
08/31/2014  10:34 PM             331,264 nssm.exe
09/06/2025  06:43 AM             108,377 pip.exe
09/06/2025  06:43 AM             108,377 pip3.12.exe
09/06/2025  06:43 AM             108,377 pip3.exe
09/06/2025  08:12 AM               507 setup_fastapi.bat
09/06/2025  06:43 AM             108,367 uvicorn.exe
09/06/2025  06:43 AM             108,367 watchfiles.exe
09/06/2025  06:43 AM             108,369 websockets.exe
09/07/2025  08:50 PM    <DIR>          __pycache__
                14 File(s)            1,529,021 bytes
                 3 Dir(s)         699,124,002,816 bytes free

admin@DESKTOP-GKTRJ7R D:\python\Scripts>
```

## 10. Viết code Back-End: Python+FastAPI+MQTT

### 10.1. Mục tiêu

- Là cầu nối giữa MQTT broker, WebSocket clients và một HTTP API sensor (SENSOR\_API\_BASE).
- Nhận message từ topic pm/change (MQTT) → phát (broadcast) tới tất cả client kết nối qua /ws.

- Expose HTTP proxy (catch-all GET/POST) để chuyển tiếp yêu cầu đến SENSOR\_API\_BASE.
- Theo dõi số client online, gửi count qua WebSocket và **xuất bản** count lên MQTT topic pm/online.
- Chạy MQTT client trong **thread nền**; dùng event loop asyncio chính cho WebSocket và broadcast.

## 10.2. Thuật toán (tóm tắt hoạt động run-time)

1. **Startup:** lưu `main_loop = asyncio.get_running_loop()` và tạo thread daemon chạy `mqtt_thread(loop)`.
2. **mqtt\_thread:** khởi tạo `paho.mqtt.client.Client`, set user/pass, cố kết nối trong vòng lặp (retry mỗi 3s nếu fail), subscribe MQTT\_TOPIC\_CHANGE\_SUB, rồi `loop_start()` để xử lý mạng MQTT.
3. **Khi MQTT nhận message (on\_message):** decode payload → dùng `asyncio.run_coroutine_threadsafe(broadcast_message(message), main_loop)` để đẩy nhiệm vụ broadcast vào event loop chính.
4. **broadcast\_message(message):** lấy `clients_lock`, duyệt `connected_clients`, gửi `ws.send_text(message)` cho từng client; nếu client bị disconnect thì xóa khỏi set.
5. **broadcast\_online\_count():** tạo JSON `{"online": N}` → gửi tới mọi WebSocket client; nếu `mqttc.is_connected()` thì publish JSON đó lên topic pm/online.
6. **on\_disconnect (MQTT):** nếu mất kết nối bất thường (`rc != 0`) thì sleep + retry reconnect (vòng lặp retry + resubscribe).
7. **WebSocket /ws:** accept kết nối, thêm ws vào `connected_clients` (dưới lock), gửi lời chào, gọi `broadcast_online_count()`. Vòng lặp `receive_text()` giữ kết nối; khi `WebSocketDisconnect` xảy ra thì remove client và cập nhật online count.
8. **HTTP proxy endpoints (GET/POST catch-all):** ghép đường dẫn vào SENSOR\_API\_BASE rồi gọi `get_api()` (dùng `httpx.AsyncClient().get()`), trả JSON; (POST hiện tại đọc body bytes và **ghép thành query string** nếu có — lưu ý hành vi này hơi khác thường).
9. **Đồng bộ/đa luồng:** paho-mqtt chạy trong thread riêng; mọi tương tác với WebSocket xảy ra trong asyncio loop; giao tiếp giữa 2 môi trường dùng `run_coroutine_threadsafe`.

## 10.3. full source code python: sử dụng FastAPI và MQTT

Tham khảo full source code trên [github](#)

## 10.4. Cấu hình chạy tự động như dịch vụ (run as service)

Tạo file cài đặt tên setup\_fastapi.bat với nội dung:

```
@echo off
set SERVICE_NAME=A1_fastAPI
set NSSM_PATH=D:\python\Scripts\nssm.exe
set PYTHON_PATH=D:\python\python.exe
set APP_DIR=D:\python\Scripts
set ARGS=-m uvicorn main:app --host "0.0.0.0" --port 8003

echo Installing %SERVICE_NAME% ...
%NSSM_PATH% install %SERVICE_NAME% "%PYTHON_PATH%" %ARGS%

echo Setting AppDirectory to %APP_DIR% ...
%NSSM_PATH% set %SERVICE_NAME% AppDirectory %APP_DIR%

echo Starting service %SERVICE_NAME% ...
%NSSM_PATH% start %SERVICE_NAME%

echo Done.
Pause
```

## 11. Cài đặt web server Apache

### 11.1. Apache Web Server là gì?

Apache HTTP Server (thường gọi ngắn là **Apache**) là **máy chủ web mã nguồn mở**, lâu đời và phổ biến nhất thế giới. Nó chạy trên đa nền tảng (Linux, Windows, macOS) và có thể phục vụ **website tĩnh (HTML, CSS, JS)** hoặc **website động** thông qua module (PHP, Python, Perl...).

### 11.2. Đặc trưng của Apache

- **Module-based**: có thể bật/tắt tính năng (SSL, Proxy, Rewrite, PHP...).
- **Cross-platform**: chạy được trên Windows, Linux, Unix, macOS.
- **Flexible config**: cấu hình bằng file httpd.conf hoặc .htaccess.
- **Hỗ trợ giao thức chuẩn**: HTTP/1.1, HTTP/2, HTTPS (SSL/TLS).
- **Khả năng mở rộng**: chạy từ website nhỏ tới hệ thống lớn.

### 11.3. Ưu điểm

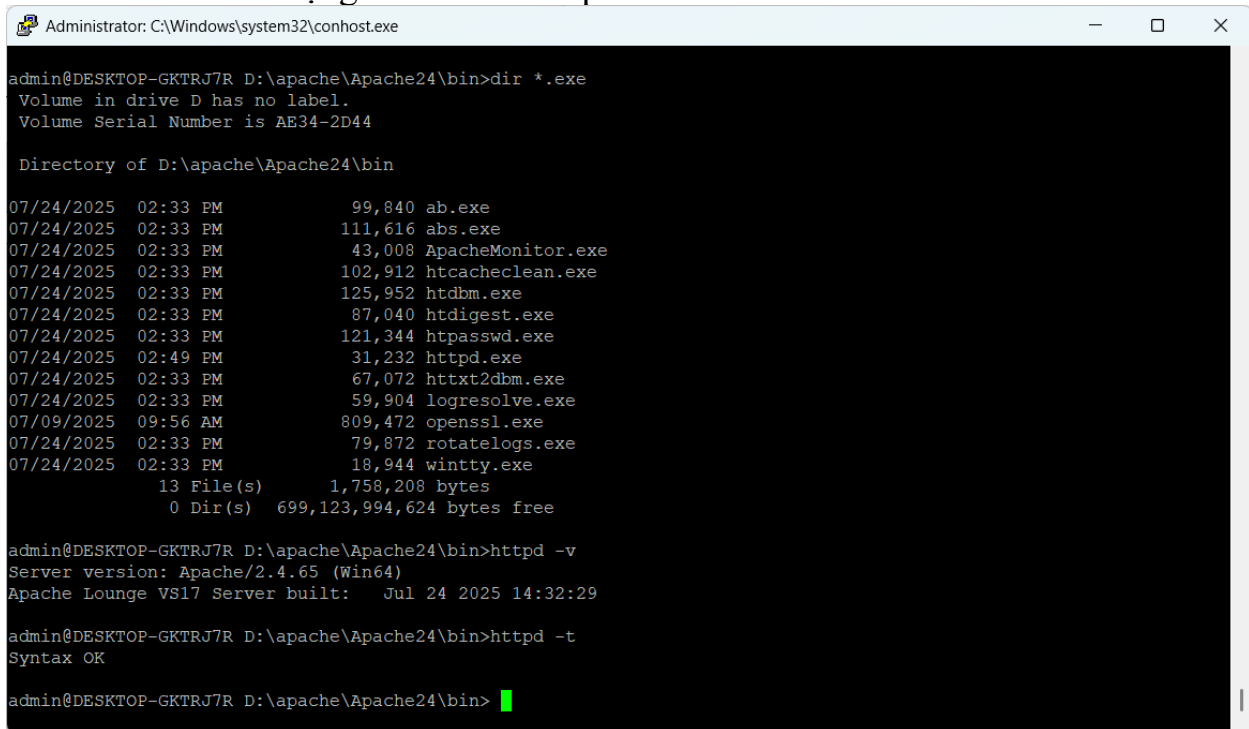
- **Miễn phí & mã nguồn mở**.
- **Cộng đồng lớn, tài liệu đầy đủ**.
- **Dễ dùng, dễ cấu hình**.
- **Ổn định, đáng tin cậy** (chạy nhiều năm mà không crash).
- **Tích hợp tốt với PHP/MySQL** → LAMP/WAMP stack.
- **Hỗ trợ nhiều module** (rewrite URL, load balancing, proxy...).

### 11.4. Nhược điểm (so với web server hiện đại như Nginx)

- **Tốn RAM/CPU** hơn khi số lượng kết nối đồng thời rất lớn.
- **Hiệu năng kém hơn** trong môi trường high-concurrency.
- **Cấu hình có thể phức tạp** với newbie.

## 11.5. Các bước cài đặt Apache trên Windows

1. Tải từ trang chính thức: <https://httpd.apache.org/download.cgi>.
2. Giải nén và chạy Apache (thường dùng bản Win64 VC build).
3. Chỉnh cấu hình trong file httpd.conf (thư mục conf).
4. Start Apache bằng httpd.exe hoặc cài làm Windows Service:
  - Cài làm service: `httpd -k install`
  - Khởi động nó lần đầu: `httpd -k start`



```
Administrator: C:\Windows\system32\conhost.exe
admin@DESKTOP-GKTRJ7R D:\apache\Apache24\bin>dir *.exe
Volume in drive D has no label.
Volume Serial Number is AE34-2D44

Directory of D:\apache\Apache24\bin

07/24/2025  02:33 PM                99,840  ab.exe
07/24/2025  02:33 PM               111,616  abs.exe
07/24/2025  02:33 PM               43,008  ApacheMonitor.exe
07/24/2025  02:33 PM               102,912  htcacheclean.exe
07/24/2025  02:33 PM               125,952  htdbm.exe
07/24/2025  02:33 PM               87,040  htdigest.exe
07/24/2025  02:33 PM               121,344  httpasswd.exe
07/24/2025  02:49 PM               31,232  httpd.exe
07/24/2025  02:33 PM               67,072  httpd24.exe
07/24/2025  02:33 PM               59,904  logresolve.exe
07/09/2025  09:56 AM              809,472  openssl.exe
07/24/2025  02:33 PM               79,872  rotatelog.exe
07/24/2025  02:33 PM               18,944  winntty.exe
      13 File(s)          1,758,208 bytes
       0 Dir(s)        699,123,994,624 bytes free

admin@DESKTOP-GKTRJ7R D:\apache\Apache24\bin>httpd -v
Server version: Apache/2.4.65 (Win64)
Apache Lounge VS17 Server built:   Jul 24 2025 14:32:29

admin@DESKTOP-GKTRJ7R D:\apache\Apache24\bin>httpd -t
Syntax OK

admin@DESKTOP-GKTRJ7R D:\apache\Apache24\bin>
```

## 11.6. Cấu hình web cùng các dịch vụ backend và grafana

Tham khảo trên [github](#)

## 12. Viết code Front-End : Html+JS+CSS

### 12.1. Mục tiêu của code (toàn trang)

- Trang web này là hệ thống giám sát **cân bằng tải**.
- Dữ liệu được lấy từ backend (qua WebSocket hoặc AJAX).
- **phanme.js** chịu trách nhiệm:
  - Nhận dữ liệu realtime từ server.
  - Render thông tin ra các khối `.sensor-can-X` và `.sensor-table-X`.
  - Điều khiển trạng thái hiển thị hình ảnh băng tải: đang chạy (**run**) hay dừng (**stop**).
- **phanme.css** định nghĩa giao diện, có hai trạng thái động cho băng tải:
  - `.bang-tai-can.run` → hiển thị ảnh động (gif) để mô tả băng chuyền đang chạy.
  - `.bang-tai-can.stop` → hiển thị ảnh tĩnh (png) khi băng chuyền dừng.

## 12.2. Thuật toán hoạt động

### Luồng xử lý trong phanme.js

1. **Khởi tạo WebSocket**: khi load trang, phanme.js mở kết nối WS đến server để nhận dữ liệu sensor.
2. **Nhận dữ liệu sensor**: server gửi về JSON, gồm thông tin như: tên cân, trạng thái, tốc độ, tổng tải...
3. **Cập nhật giao diện**:
  - Chèn dữ liệu số liệu (tốc độ, lưu lượng, trọng lượng...) vào các thẻ tương ứng trong .sensor-can-\*
  - Kiểm tra trạng thái **có tải hay không** / **có chạy hay không**.
4. **Chọn class hiển thị bằng tải**:
  - Nếu sensor báo **cân đang chạy** → gán class .bang-tai-can run → CSS sẽ hiển thị ảnh động /images/bang\_tai\_run.gif.
  - Nếu sensor báo **cân dừng** → gán class .bang-tai-can stop → CSS hiển thị ảnh tĩnh /images/bang\_tai\_stop.png.
  - Việc chuyển đổi này thường là .addClass("run") / .removeClass("stop") hoặc ngược lại.
5. **Cập nhật liên tục** mỗi khi có dữ liệu mới từ WebSocket.

## 12.3. Ý nghĩa của hai selector CSS

- .bang-tai-can.run → nên là ảnh động mô tả bằng chuyển vận hành → cho người giám sát biết máy đang chạy.
- .bang-tai-can.stop → nên là ảnh tĩnh mô tả bằng chuyển dừng → cảnh báo hoặc cho biết trạng thái nghỉ.

## 12.4. Full source-code html css js

Tham khảo trên [github](#)

## 13. Tạo các đồ thị trên Grafana

Mục tiêu để Front-End hiển thị qua iframe

Các đồ thị lấy nguồn dữ liệu từ influxDB

### 13.1. Kết nối Grafana với InfluxDB 2.7

1. Vào **Configuration** → **Data sources** → **Add data source**.
2. Chọn **InfluxDB**.
3. Ở mục **Query Language** → chọn **Flux** (vì InfluxDB 2.x mặc định dùng Flux).
4. Điền:
  - **URL**: http://localhost:8086
  - **Organization**: tên org khi setup InfluxDB



- **Token:** token đã tạo trong InfluxDB
  - **Default Bucket:** tên bucket chứa dữ liệu sensor
5. Nhấn **Save & Test** → nếu OK là kết nối thành công.

### 13.2. Tạo Dashboard và Panel (đồ thị)

1. Vào menu + → **Dashboard** → **Add new panel**.
2. Chọn data source = **InfluxDB (Flux)**.
3. Ở khung truy vấn, nhập Flux query. Ví dụ hiển thị nhiệt độ 1h gần đây:  

```
from(bucket: "sensor_bucket")
  |> range(start: -1h)
  |> filter(fn: (r) => r._measurement == "temperature")
  |> filter(fn: (r) => r._field == "value")
  |> aggregateWindow(every: 1m, fn: mean, createEmpty: false)
  |> yield(name: "mean")
```
4. Bên phải chọn visualization = **Time series**.
5. Nhấn **Apply** → sẽ thấy biểu đồ nhiệt độ theo thời gian.

### 13.3. Lưu Dashboard

- Nhấn **Save dashboard** → đặt tên → từ nay mở lại sẽ thấy panel vừa tạo.

### 13.4. Dùng đồ thị Grafana trên web (iframe):

- Lưu gid của đồ thị vào csdl mariadb
- js lấy các gid này về khi trang tải lần 1
- Tạo url trỏ đến đồ thị, đưa url vào iframe để hiển thị (cần cấu hình sub path cho grafana trước đó)

## 14. Đăng ký domain trên DuckDNS

### 14.1. DuckDNS là gì?

DuckDNS là một dịch vụ **Dynamic DNS (DDNS) miễn phí**.

Nó cho phép bạn có một tên miền dạng xxxx.duckdns.org trỏ về IP nhà bạn.

Nếu IP thay đổi (do nhà mạng cấp DHCP), DuckDNS sẽ tự động cập nhật IP mới, nhờ đó bạn luôn truy cập được vào server ở nhà qua tên miền cố định.

### 14.2. Các bước đăng ký free host trên DuckDNS

1. Truy cập: <https://www.duckdns.org>
2. Đăng nhập bằng tài khoản Google, GitHub, Twitter hoặc Reddit.
3. Sau khi login, bạn sẽ thấy trang quản lý với **token** của bạn.
4. Trong mục **Domains**, nhập tên subdomain bạn muốn (ví dụ: myhome) → nhấn **add domain** → sẽ có myhome.duckdns.org.

5. DuckDNS sẽ hiển thị lệnh update, ví dụ dùng curl hoặc script, để bạn cập nhật IP định kỳ.

### 14.3. Token của DuckDNS để làm gì?

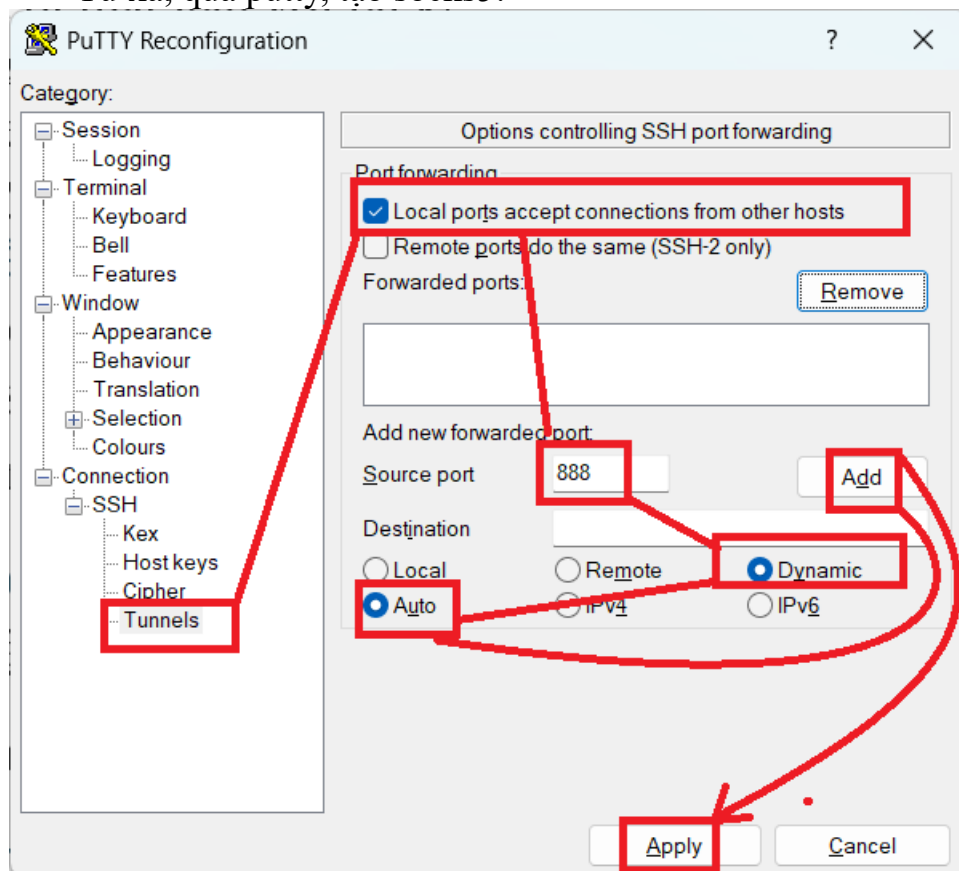
- **Token** là mã bí mật gắn với tài khoản của bạn.
- Nó dùng để **xác thực khi update IP** cho domain.
- Sử dụng token này trên Node-RED để tự động cập nhật ip động cho domain (tự làm, gợi ý sử dụng thư viện [node-red-contrib-duckdns](#))

## 15. Tạo form đăng nhập

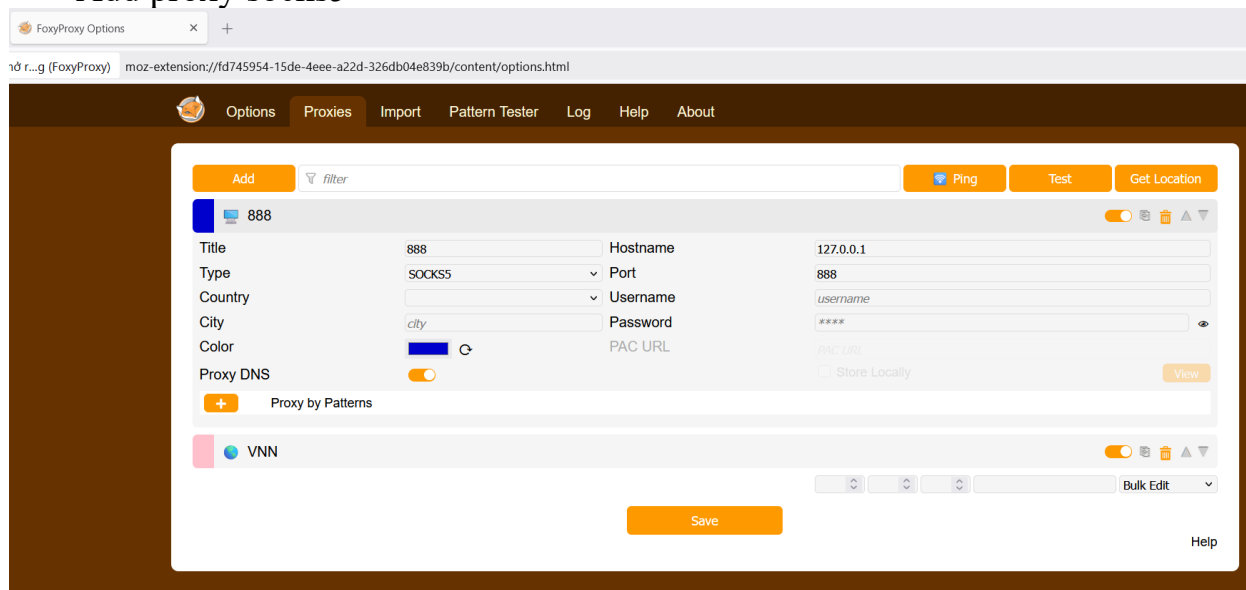
Lúc đầu làm đc hiển thị ok, sau đó thêm chức năng login để giới hạn, login thành công mới hiển thị thông số, do đó cần thêm bảng user, cookie; thêm nút logout đỏ, code trên [github](#)

## 16. Truy cập gpon router để mở port 80 443

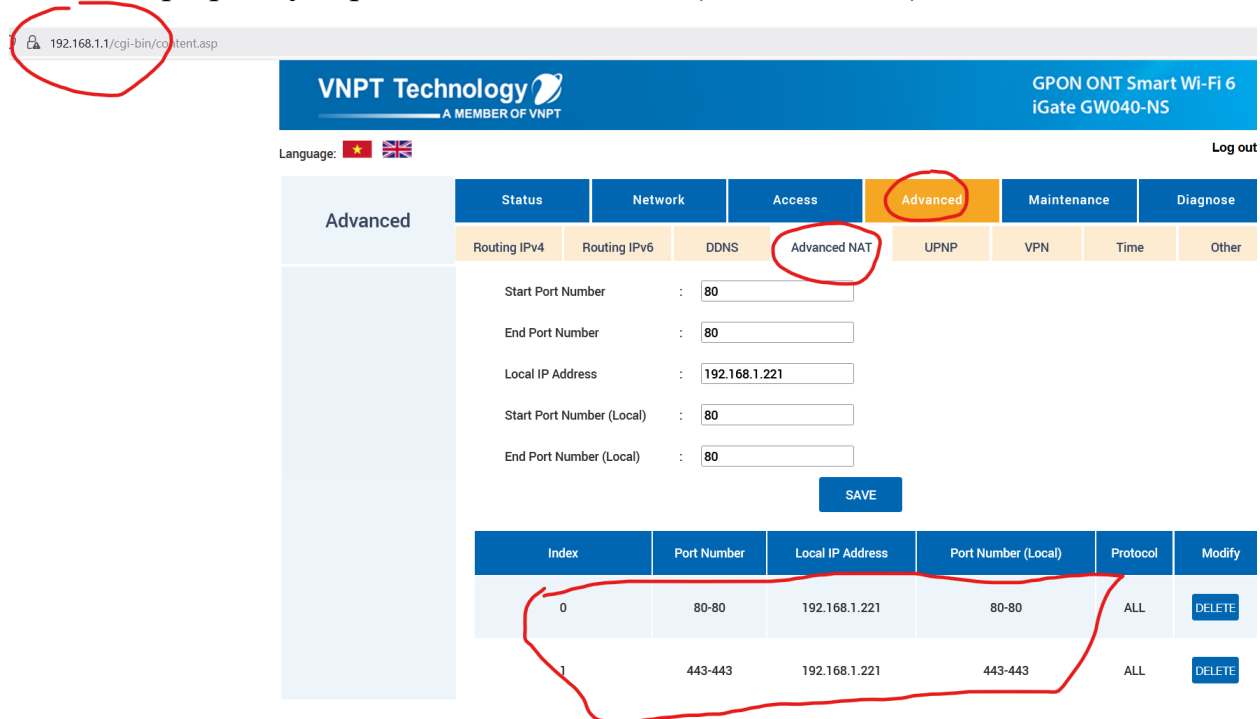
Tùy router, ở đây tập trung vào router gpon của vnpt  
Từ xa, qua putty, tạo socks5:



## Sử dụng trình duyệt FireFox, cài addon FoxyProxy Add proxy socks5



Tại laptop, truy cập từ firefox vào router (router ở xa xa)



Chú ý: 192.168.1.221 là ip của máy tính tại công ty.  
Với trường hợp của riêng bạn, ip có thể sẽ khác đi

## 17. Tool acme để tạo chứng chỉ ssl, cấu hình https trên apache

### 17.1. SSL là gì?

- **SSL (Secure Sockets Layer)** là giao thức mã hóa để bảo mật kết nối giữa trình duyệt và server.
- Thực tế bây giờ ta dùng **TLS (Transport Layer Security)**, nhưng mọi người vẫn quen gọi là SSL.
- Khi web có SSL/TLS → dùng HTTPS thay vì HTTP.

### 17.2. Tại sao HTTPS lại an toàn?

1. **Mã hóa (Encryption):** dữ liệu truyền đi được mã hóa → hacker không đọc được.
  2. **Xác thực (Authentication):** chứng chỉ SSL xác minh bạn đang nói chuyện với đúng server (không bị “giả mạo server”).
  3. **Toàn vẹn dữ liệu (Integrity):** ngăn dữ liệu bị sửa trong quá trình truyền.
- Vì thế HTTPS an toàn hơn HTTP, đặc biệt cho login, thanh toán, IoT, API.

### 17.3. Tool ACME là gì?

- **ACME (Automatic Certificate Management Environment)** là giao thức do Let's Encrypt phát triển để tự động xin/cấp/gia hạn chứng chỉ SSL miễn phí.
- Có nhiều tool client hỗ trợ ACME, nổi tiếng nhất:
  - **certbot** (Python)
  - **acme.sh** (bash script, rất nhẹ và phổ biến)

### 17.4. Chức năng của ACME tool

- Tự động:
  - Sinh private key và CSR.
  - Liên hệ với CA (ví dụ Let's Encrypt).
  - Thực hiện xác minh tên miền (HTTP-01, DNS-01).
  - Lấy chứng chỉ SSL.
  - Gia hạn chứng chỉ trước khi hết hạn.

### 17.5. Cách dùng ACME cho web chạy với Apache

#### 1. Cài acme

Cài trên windows: truy cập <https://www.win-acme.com/>

Download: [phần mềm](#)

Giải nén

#### 2. Xin chứng chỉ cho domain

chạy file wacs.exe với quyền admin, nhập lệnh

```
wacs.exe --target manual --host myhost.duckdns.org ^
--validation filesystem ^
--webroot "D:\apache\Apache24\htdocs" ^
--store pemfiles --pemfilespace "D:\apache\Apache24\cert\myhost" ^
--friendlyname "myhost.duckdns.org"
```

Trong D:\apache\Apache24\cert\myhost\ sẽ tạo ra 4 file:

```
myhost.duckdns.org-chain-only.pem
myhost.duckdns.org-chain.pem
myhost.duckdns.org-crt.pem
myhost.duckdns.org-key.pem
```

### 3. Cấu hình Apache: xem trên [github](#)

Hi vọng ai đọc đến đây thấy được tâm huyết của tác giả khi xây dựng tài liệu này => bắt tay vào thử nghiệm để có được kinh nghiệm./.

```
Administrator: C:\Windows\system32\conhost.exe
admin@DESKTOP-GKTRJ7R D:\mariadb\bin>sc query

SERVICE_NAME: A1-Node-RED
DISPLAY_NAME: A1-Node-RED
        TYPE               : 10   WIN32_OWN_PROCESS
        STATE                : 4     RUNNING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE       : 0     (0x0)
        SERVICE_EXIT_CODE    : 0     (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x0

SERVICE_NAME: A1_fastAPI
DISPLAY_NAME: A1_fastAPI
        TYPE               : 10   WIN32_OWN_PROCESS
        STATE                : 4     RUNNING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE       : 0     (0x0)
        SERVICE_EXIT_CODE    : 0     (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x0

SERVICE_NAME: A2-influxdb
DISPLAY_NAME: A2-influxdb
        TYPE               : 10   WIN32_OWN_PROCESS
        STATE                : 4     RUNNING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE       : 0     (0x0)
        SERVICE_EXIT_CODE    : 0     (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x0

SERVICE_NAME: Apache2.4
DISPLAY_NAME: Apache2.4
        TYPE               : 10   WIN32_OWN_PROCESS
        STATE                : 4     RUNNING
                                (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE       : 0     (0x0)
        SERVICE_EXIT_CODE    : 0     (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x0
```