

RK3568 设备上系统部署及使用说明

1. 文档目的

本文档用于说明如何在 RK3568 + Ubuntu 20.04 设备上部署 EMCP 电气量测控平台的前后端系统，并完成以下目标：

- 部署后端 FastAPI + Uvicorn 服务
- 部署前端静态页面
- 配置前后端开机自动启动
- 登录桌面后自动打开浏览器并访问主页面
- 给出现场快速设置和日常使用说明

2. 适用范围

适用对象：

- 硬件平台：RK3568
- 操作系统：Ubuntu 20.04
- 部署模式：前后端均部署在同一台设备本机

当前工程实际情况：

- 后端运行端口：8000
- 前端访问地址建议使用：`http://127.0.0.1:5173/`
- 前端默认请求后端地址：`http://127.0.0.1:8000/api`
- 前端默认请求头中携带：`X-API-Token: admin123`
- 默认安全校验密码：`admin123`
- 后端默认启用模拟采集：`use_mock_device = True`

说明：

- 当前前端代码默认调用本机 `127.0.0.1:8000` 的后端接口，因此推荐前后端部署在同一台 RK3568 设备上
- 如果后续需要将前后端拆分到不同设备，需要同步调整前端接口配置

3. 部署建议

推荐目录规划如下：

```
/opt/emcp/  
├─ backend/  
├─ frontend/  
└─ scripts/
```

建议使用以下方式部署：

- 后端：systemd 托管 uvicorn
- 前端：前端执行 `npm run build` 后，用轻量静态服务对 `dist/` 目录发布
- 浏览器自启动：使用桌面自动启动项 `~/.config/autostart/*.desktop`

4. 环境准备

4.1 安装系统依赖

在 RK3568 设备终端执行：

```
sudo apt update
sudo apt install -y python3.8 python3.8-venv python3-pip curl git
sudo apt install -y nodejs npm
```

说明：

- Ubuntu 20.04 默认可直接使用 Python 3.8
- 如果设备上的 nodejs 版本过低，建议升级到 Node.js 18 LTS 后再安装前端依赖

4.2 安装浏览器

如果设备带图形桌面，建议至少安装一个浏览器：

```
sudo apt install -y chromium-browser
```

如果系统仓库没有 chromium-browser，也可以使用：

```
sudo apt install -y firefox
```

4.3 拷贝项目代码

将工程目录拷贝到设备，例如：

```
sudo mkdir -p /opt/emcp
sudo chown -R $USER:$USER /opt/emcp
```

然后将项目复制到：

```
/opt/emcp/backend
/opt/emcp/frontend
```

如果是通过 git 拉取，也可以在 /opt/emcp 下执行：

```
git clone <你的仓库地址> /opt/emcp
```

如果项目目录本身已经包含 `backend` 和 `frontend`，则以实际目录结构为准。

5. 后端部署

5.1 创建虚拟环境

```
cd /opt/emcp/backend
python3.8 -m venv .venv
source .venv/bin/activate
python -m pip install --upgrade pip
```

5.2 安装后端依赖

```
cd /opt/emcp/backend
source .venv/bin/activate
pip install -e .
```

如需测试或调试工具，再安装：

```
pip install -e .[dev]
```

5.3 手动启动后端验证

```
cd /opt/emcp/backend
source .venv/bin/activate
python -m uvicorn app.main:app --host 0.0.0.0 --port 8000
```

验证方式：

- 本机健康检查：<http://127.0.0.1:8000/>
- 接口文档：<http://127.0.0.1:8000/docs>

如果能正常返回服务信息，说明后端部署成功。

6. 前端部署

6.1 安装前端依赖

```
cd /opt/emcp/frontend
npm install
```

6.2 构建前端

```
cd /opt/emcp/frontend
npm run build
```

构建完成后会生成：

```
/opt/emcp/frontend/dist
```

6.3 手动启动前端验证

推荐使用 Python 内置静态服务快速发布：

```
cd /opt/emcp/frontend
python3 -m http.server 5173 --directory dist --bind 0.0.0.0
```

验证地址：

- 前端页面：<http://127.0.0.1:5173/>

说明：

- 当前前端默认调用本机后端 <http://127.0.0.1:8000/api>
- 因此前端页面只要在设备本机打开，即可直接访问后端服务

7. 首次联调检查

建议按以下顺序进行：

1. 先启动后端
2. 再启动前端静态服务
3. 浏览器打开 <http://127.0.0.1:5173/>
4. 检查实时数据、设备状态、报警历史等页面是否能正常进入

当前项目已打通页面包括：

- 实时数据
- 设备状态
- 设备配置
- 通道配置
- 报警设置
- 报警历史
- 控制指令
- 系统设置

补充说明：

- `设备状态` 页面支持刷新
- `报警历史` 页面支持分页和刷新
- 配置保存前会弹出安全校验密码输入框
- 当前默认安全校验密码为 `admin123`

8. 配置开机自动启动

推荐使用 `systemd` 管理前后端服务。

8.1 配置后端服务

创建文件：

```
/etc/systemd/system/emcp-backend.service
```

内容如下：

```
[Unit]
Description=EMCP Backend Service
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
User=ubuntu
WorkingDirectory=/opt/emcp/backend
ExecStart=/opt/emcp/backend/.venv/bin/python -m uvicorn app.main:app --host 0.0.0.0 --port 8000
Restart=always
RestartSec=3

[Install]
WantedBy=multi-user.target
```

说明：

- `User=ubuntu` 需要替换为设备实际登录用户名
- 如需切换真实采集设备，可结合 `.env` 或配置文件调整后端参数

8.2 配置前端服务

创建文件：

```
/etc/systemd/system/emcp-frontend.service
```

内容如下：

```
[Unit]
```

```
Description=EMCP Frontend Static Service
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
User=ubuntu
WorkingDirectory=/opt/emcp/frontend
ExecStart=/usr/bin/python3 -m http.server 5173 --directory /opt/emcp/frontend/dist --bind 0.0.0.0
Restart=always
RestartSec=3

[Install]
WantedBy=multi-user.target
```

8.3 启用开机自启

执行：

```
sudo systemctl daemon-reload
sudo systemctl enable emcp-backend.service
sudo systemctl enable emcp-frontend.service
sudo systemctl start emcp-backend.service
sudo systemctl start emcp-frontend.service
```

查看状态：

```
sudo systemctl status emcp-backend.service
sudo systemctl status emcp-frontend.service
```

查看日志：

```
journalctl -u emcp-backend.service -f
journalctl -u emcp-frontend.service -f
```

9. 开机自动打开浏览器访问主页面

本功能依赖图形桌面环境。若设备只启动命令行、不进入桌面，则无法自动弹出浏览器。

9.1 前置条件

建议提前确认以下两项：

- 设备已安装浏览器
- 设备已启用桌面自动登录，或现场人员上电后会手动登录桌面

说明：

- 只有在用户登录图形桌面后，自动启动项才会执行

- 如果希望“上电后无需人工登录就自动打开页面”，需要同时开启系统自动登录

9.2 创建浏览器启动脚本

创建目录：

```
mkdir -p /opt/emcp/scripts
```

创建脚本文件：

```
/opt/emcp/scripts/start-emcp-browser.sh
```

内容如下：

```
#!/bin/bash
sleep 8

URL="http://127.0.0.1:5173/"

if command -v chromium-browser >/dev/null 2>&1; then
    chromium-browser --start-fullscreen "$URL" >/dev/null 2>&1 &
elif command -v firefox >/dev/null 2>&1; then
    firefox "$URL" >/dev/null 2>&1 &
else
    xdg-open "$URL" >/dev/null 2>&1 &
fi
```

赋予执行权限：

```
chmod +x /opt/emcp/scripts/start-emcp-browser.sh
```

说明：

- `sleep 8` 用于等待前后端服务启动完成
- 如果设备性能较低，可改为 `sleep 10` 或 `sleep 15`
- 如需全屏显示，优先使用 `chromium-browser --start-fullscreen`

9.3 配置桌面自动启动项

创建目录：

```
mkdir -p ~/.config/autostart
```

创建文件：

```
~/.config/autostart/emcp-browser.desktop
```

内容如下：

```
[Desktop Entry]
Type=Application
Name=EMCP Browser Autostart
Exec=/opt/emcp/scripts/start-emcp-browser.sh
X-GNOME-Autostart-enabled=true
Terminal=false
```

配置完成后，用户每次登录桌面时都会自动打开浏览器并访问：

```
http://127.0.0.1:5173/
```

10. 快捷设置说明

这一节适合现场交付时快速操作。

10.1 一次性完成部署

按以下顺序操作即可：

1. 拷贝项目到 `/opt/emcp`
2. 创建后端虚拟环境并安装依赖
3. 前端执行 `npm install` 和 `npm run build`
4. 创建两个 `systemd` 服务
5. 创建浏览器自动启动脚本
6. 创建 `~/config/autostart/emcp-browser.desktop`
7. 执行 `systemctl enable` 完成开机自启

10.2 最常用检查命令

检查后端状态：

```
sudo systemctl status emcp-backend.service
```

检查前端状态：

```
sudo systemctl status emcp-frontend.service
```

重启后端：

```
sudo systemctl restart emcp-backend.service
```

重启前端：

```
sudo systemctl restart emcp-frontend.service
```

重新打开浏览器主页：

```
/opt/emcp/scripts/start-emcp-browser.sh
```

10.3 修改主页地址

如果前端端口改为其他端口，例如 8080，需要同步修改以下两处：

- /opt/emcp/scripts/start-emcp-browser.sh 中的 URL
- 前端静态服务端口或访问地址

10.4 修改登录后自动打开行为

如果不希望每次登录都自动打开浏览器，可以删除或重命名：

```
~/ .config/autostart/emcp-browser.desktop
```

11. 日常使用说明

11.1 启动后访问方式

当设备开机完成并进入桌面后，系统会自动：

- 启动后端服务
- 启动前端静态服务
- 打开浏览器访问主页

默认主页地址：

```
http://127.0.0.1:5173/
```

11.2 页面使用提示

- 实时数据页：查看模拟或采集到的实时数据
- 设备状态页：支持手动刷新状态
- 报警历史页：支持分页和刷新
- 配置类页面：保存前需要输入安全校验密码

当前默认安全校验密码：

```
admin123
```

建议交付后尽快修改为现场正式密码。

12. 常见问题

12.1 浏览器没有自动打开

请检查：

- 是否进入了图形桌面
- `~/.config/autostart/emcp-browser.desktop` 是否存在
- `/opt/emcp/scripts/start-emcp-browser.sh` 是否有执行权限
- 浏览器是否已安装

12.2 浏览器打开了，但页面访问失败

请检查：

- 前端服务是否启动成功
- `5173` 端口是否被其他程序占用
- `/opt/emcp/frontend/dist` 是否已经构建生成

12.3 页面能打开，但接口不通

请检查：

- 后端服务是否启动成功
- `8000` 端口是否正常监听
- 前端默认接口地址是否仍为 `http://127.0.0.1:8000/api`

12.4 页面保存时报权限或校验失败

请检查：

- 默认请求头 `X-API-Token` 是否被修改
- 输入的安全校验密码是否正确
- 当前默认密码是否仍为 `admin123`

12.5 修改代码后页面没有变化

请检查是否重新执行了前端构建：

```
cd /opt/emcp/frontend
npm run build
sudo systemctl restart emcp-frontend.service
```

后端代码修改后则需要重启后端服务：

```
sudo systemctl restart emcp-backend.service
```

13. 推荐交付方式

建议现场交付时采用以下方式：

- 设备开机自动进入桌面
- 后端和前端均使用 `systemd` 自启动
- 浏览器自动全屏打开主页
- 保留一个终端窗口或 SSH 方式用于维护

如需进一步增强生产部署能力，后续可继续补充：

- `Nginx` 托管前端静态资源
- 反向代理后端接口
- HTTPS 证书
- 真实硬件采集驱动接入
- 自动备份配置与报警数据库