StarFive 2 编译 Home Assistant 2026.8.3

按实际操作顺序整理的命令清单 · Debian trixie / RISC-V riscv64

1 克隆源码与初始检查

```
git clone https://github.com/home-assistant/core.git
cd core
uname -m
cat /etc/os-release
free -h
df -h
```

2 添加 Swap（编译刚需）

```
sudo fallocate -l 4G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
free -h
```

3 安装系统依赖

3.1 正常 apt 安装 OK

```
sudo apt update
sudo apt install -y build-essential libssl-dev zlib1g-dev libbz2-dev libreadline-dev \
  libsqlite3-dev curl git libncursesw5-dev xz-utils tk-dev libxml2-dev libxmlsec1-dev \
  libffi-dev liblzma-dev
```

3.2 解决 libnss3-dev 冲突 dpkg 强装

```
wget http://ftp.de.debian.org/debian/pool/main/n/nss/libnss3_3.129-1_riscv64.deb
sudo dpkg -i --force-depends libnss3_3.129-1_riscv64.deb
wget http://ftp.de.debian.org/debian/pool/main/n/nss/libnss3-dev_3.129-1_riscv64.deb
sudo dpkg -i --force-depends libnss3-dev_3.129-1_riscv64.deb
sudo apt --fix-broken install
```

3.3 安装剩余开发包 OK

```
sudo apt install -y libbz2-dev libreadline-dev libsqlite3-dev liblzma-dev \
  libncursesw5-dev libgdbm-dev libnss3-dev uuid-dev libtiff-dev libwebp-dev \
  libxml2-dev libxslt1-dev libturbojpeg0-dev
```

4 安装 Python 3.14.2 (pyenv)

```
curl https://pyenv.run | bash
cat >> ~/.bashrc <<'EOF'
```

```
export PYENV_ROOT="$HOME/.pyenv"
export PATH="$PYENV_ROOT/bin:$PATH"
eval "$(pyenv init --path)"
eval "$(pyenv init -)"
EOF
source ~/.bashrc
pyenv install 3.14.2
cd ~/core
pyenv local 3.14.2
python --version
```

5 编译 FFmpeg 7.1

```
cd ~
wget https://ffmpeg.org/releases/ffmpeg-7.1.tar.xz
tar xf ffmpeg-7.1.tar.xz
cd ffmpeg-7.1
./configure --enable-shared --disable-static --prefix=/usr/local
make -j4
sudo make install
sudo ldconfig
```

6 切换 HA 源码版本并创建 venv

```
cd ~/core
git checkout 2026.8.3
pyenv local 3.14.2
python -m venv venv
```

```
source venv/bin/activate
pip install --upgrade pip setuptools wheel
```

7 安装 HA 核心依赖

```
pip install -r requirements.txt
```

8 编译 PyAV 17.0.1（修补源码）

8.1 获取源码与修补

```
cd ~
git clone --branch v17.0.1 --depth 1 https://github.com/PyAV-Org/PyAV.git av-17.0.1-full
cd av-17.0.1-full
sed -i '7d' av/option.py
sed -i 's/sws_free_context/sws_freeContext/g' $(grep -rl "sws_free_context" av/)
nano av/video/reformatter.py   # 注释掉 self.ptr.threads 和 self.ptr.flags 赋值
nano av/video/reformatter.pxd  # 修改 sws_free_context 声明为 sws_freeContext(SwsContext *ctx)
```

8.2 编译安装

```
export PKG_CONFIG_PATH=/usr/local/lib/pkgconfig:$PKG_CONFIG_PATH
rm -rf build
python setup.py build_ext --inplace --ffmpeg-dir=/usr/local
pip install . --no-build-isolation
python -c "import av; print(av.__version__, av.library_versions['libavcodec'])"
```

9 安装 HA 本体

```
cd ~/core
source venv/bin/activate
pip install -e . --no-deps
hass --version
```

10 首次启动与翻译修复

```
hass -c ~/ha-config
# 若提示翻译缺失，运行：
python -m script.translations develop --all
```

11 配置 systemd 开机自启

```
sudo nano /etc/systemd/system/home-assistant@user.service
```

服务文件内容：

```
[Unit]
Description=Home Assistant
After=network-online.target

[Service]
Type=simple
User=%i
```

```
ExecStart=/home/user/core/venv/bin/hass -c "/home/user/ha-config"
Restart=on-failure
RestartSec=10
KillMode=process
TimeoutStopSec=60

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

```
sudo systemctl daemon-reload
sudo systemctl enable --now home-assistant@user.service
sudo systemctl status home-assistant@user.service
```

12 安装 HACS 极速版

```
cd ~/ha-config
wget -O - https://get.hacs.vip | bash -
sudo systemctl restart home-assistant@user.service
```

13 禁用旧 Docker 容器自启

```
sudo docker update --restart=no homeassistant
sudo docker stop homeassistant
```

> ✅ **完成**

HA 2026.8.3 已成功运行在 StarFive 2 上，并配置为 systemd 开机自启。
