

Nº 180056

Fazendo uma distro Linux mínima: compilação cruzada do Kernel

Douglas Bellomo Cavalcante
Rodrigo Dias Garcia
Bruno Gabriel da Fonseca

*Palestra apresentada no curso
no IPT/SITEC. 4 slides.*

A série “Comunicação Técnica” compreende trabalhos elaborados por técnicos do IPT, apresentados em eventos, publicados em revistas especializadas ou quando seu conteúdo apresentar relevância pública.
PROIBIDO A REPRODUÇÃO, APENAS PARA CONSULTA.

TREINAMENTO: CONSTRUINDO UM LINUX EMBARCADO DO ZERO (5 DIAS)

Dia 3: Compilação Cruzada do Kernel (Mainline)

Público-alvo: Estudantes/Profissionais com conhecimentos básicos de terminal Linux.

Hardware necessário: Computador Host (Linux/WSL) e Raspberry Pi 4 (Target).

Ministrado por: Douglas Bellomo Cavalcante, Rodrigo Dias Garcia e Bruno Gabriel da Fonseca

Objetivo

Compilar o Kernel Linux oficial (Mainline/Upstream) adaptando a configuração padrão do fabricante (Vendor).

1. Teoria: Mainline vs. Vendor Kernel

- **Vendor Kernel:** Versão mantida pela Raspberry Pi Foundation. Possui suporte total ao hardware específico, mas contém código customizado.
- **Mainline Kernel:** Versão oficial do Kernel.org. Código mais limpo e padronizado, mas exige adaptação das configurações para hardwares específicos.
- **Device Tree:** Arquivo binário (.dtb) que descreve a topologia do hardware para o Kernel.

2. Prática: Download e Preparação

2.1. Download do Kernel

Baixe uma versão LTS (Long Term Support) do Kernel oficial:

```
Bash
```

```
cd $WORK_DIR/sources  
wget https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.6.63.tar.xz  
tar -xvf linux-6.6.63.tar.xz  
cd linux-6.6.63
```

2.2. Obtenção da Configuração Base

Baixe o arquivo de configuração padrão (defconfig) do repositório da Raspberry Pi para usar como base:

```
Bash  
wget https://raw.githubusercontent.com/raspberrypi/linux/rpi-6.6.y/arch/arm64/configs/bcm2711\_defconfig -O .config
```

3. Prática: Compilação

3.1. Adaptação da Configuração

Execute o comando para adaptar o .config do Vendor para a árvore do Mainline. Isso removerá opções obsoletas e definirá padrões para novas opções:

```
Bash  
make olddefconfig
```

3.2. Compilação do Kernel e Binários

Inicie o processo de compilação (utilizando todos os núcleos do processador):

Bash

```
make -j$(nproc) Image modules dtbs
```

3.3. Organização dos Artefatos

Copie os arquivos resultantes para a pasta de saída:

Bash

```
# Copia o Kernel descomprimido
```

```
cp arch/arm64/boot/Image $WORK_DIR/output/kernel8.img
```

```
# Copia o Device Tree (No Mainline, localizado em broadcom)
```

```
cp arch/arm64/boot/dts/broadcom/bcm2711-rpi-4-b.dtb $WORK_DIR/output/
```

