

Guia de Resolução de Problemas - Orange Pi Provisioning

Este guia fornece soluções para problemas comuns encontrados durante o uso do sistema de provisionamento.

Índice

- [Problemas de Ambiente](#)
- [Problemas de Docker](#)
- [Problemas de Hardware](#)
- [Problemas de Rede](#)
- [Problemas de Deploy](#)
- [Problemas de Validação](#)
- [Logs e Diagnóstico](#)
- [FAQ](#)

Problemas de Ambiente

Docker não instalado ou não funcionando

Sintomas:

```
bash: docker: command not found
# ou
permission denied while trying to connect to the Docker daemon socket
```

Soluções:

```
# Instalar Docker (Ubuntu/Debian)
sudo apt-get update
sudo apt-get install -y docker.io docker-compose

# Adicionar usuário ao grupo docker
sudo usermod -aG docker $USER
newgrp docker

# Verificar instalação
docker --version
docker compose version
```

Permissões insuficientes

Sintomas:

```
Permission denied (publickey).
# ou
sudo: required for this operation
```

Soluções:

```
# Verificar permissões de arquivos
ls -la scripts/
chmod +x scripts/*.sh

# Executar com privilégios quando necessário
docker compose run --rm --privileged provisioner scripts/deploy-ender3.sh

# Verificar chave SSH
ls -la ~/.ssh/
ssh-keygen -t rsa -b 4096 -C "your.email@example.com"
```

Dependências em falta**Sintomas:**

```
jq: command not found
# ou
shellcheck: command not found
```

Soluções:

```
# Instalar dependências essenciais
sudo apt-get update
sudo apt-get install -y jq shellcheck curl wget git

# Verificar instalação
jq --version
shellcheck --version
```

**Problemas de Docker****Build falha****Sintomas:**

```
ERROR [internal] load metadata for docker.io/library/ubuntu:22.04
# ou
failed to solve with frontend dockerfile.v0
```

Soluções:

```
# Limpar cache do Docker
docker system prune -a -f

# Rebuild sem cache
docker compose build --no-cache

# Verificar conectividade
ping docker.io

# Verificar espaço em disco
df -h
docker system df
```

Container não inicia

Sintomas:

```
docker: Error response from daemon: container exited with code 125
# ou
OCI runtime create failed
```

Soluções:

```
# Verificar logs do container
docker compose logs

# Executar em modo interativo para debug
docker compose run --rm provisioner bash

# Verificar recursos do sistema
free -h
df -h

# Reiniciar serviço Docker
sudo systemctl restart docker
```

Problemas de volume/mount

Sintomas:

```
no such file or directory
# ou
bind source path does not exist
```

Soluções:

```
# Verificar caminhos no docker-compose.yml
cat docker-compose.yml

# Criar diretórios necessários
mkdir -p state logs images reports

# Verificar permissões
ls -la state/ logs/ images/ reports/

# Corrigir propriedade se necessário
sudo chown -R $USER:$USER state/ logs/ images/ reports/
```

Problemas de Hardware

MicroSD não detectado

Sintomas:

```
No USB storage devices found
# ou
/dev/sdb: No such file or directory
```

Soluções:

```
# Verificar dispositivos USB
lsblk -d -o NAME,SIZE,TRAN | grep usb
lsusb

# Verificar se MicroSD está montado
mount | grep sd

# Desmontar se necessário
sudo umount /dev/sdb*

# Verificar integridade do cartão
sudo fsck /dev/sdb

# Testar com outro cartão/leitor
```

Problemas de escrita no MicroSD

Sintomas:

```
dd: error writing '/dev/sdb': No space left on device
# ou
Input/output error
```

Soluções:

```
# Verificar espaço disponível
df -h /dev/sdb

# Verificar integridade do cartão
sudo badblocks -v /dev/sdb

# Reformatar cartão (CUIDADO: apaga dados)
sudo fdisk /dev/sdb
# d (delete all partitions)
# n (new partition)
# w (write)

# Formatar como FAT32
sudo mkfs.vfat -F 32 /dev/sdb1

# Testar velocidade de escrita
sudo dd if=/dev/zero of=/dev/sdb bs=1M count=100 status=progress
```

Orange Pi não inicializa

Sintomas:

- LED não acende
- Não aparece na rede
- Não responde a ping

Soluções:

```
# Verificar fonte de alimentação
# Orange Pi Zero 3: 5V/2A mínimo
# Orange Pi Zero 2W: 5V/1.5A mínimo

# Verificar integridade da imagem
sha256sum images/Armbian_*.img

# Regravar MicroSD
docker compose run --rm --privileged provisioner scripts/deploy-ender3.sh

# Testar com monitor HDMI conectado
# Verificar mensagens de boot

# Verificar se MicroSD está bem inserido
# Testar com outro MicroSD
```

Problemas de Rede

WiFi não conecta

Sintomas:

```
ping: 192.168.1.100: Name or service not known
# ou
ssh: connect to host 192.168.1.100 port 22: No route to host
```

Soluções:

```
# Verificar configuração WiFi
cat state/local-info.json | jq '.wifi'

# Reconfigurar WiFi
docker compose run --rm provisioner scripts/collect-local-info.sh

# Verificar se SSID está correto
iwlist scan | grep ESSID

# Testar conectividade da rede host
ping 192.168.1.1 # Gateway
ping 8.8.8.8     # Internet

# Conectar monitor HDMI ao Orange Pi
# Verificar logs de rede: dmesg | grep wlan
```

IP estático não funciona

Sintomas:

```
# Orange Pi recebe IP diferente do configurado
ping 192.168.1.100 # Falha
nmap -sn 192.168.1.0/24 # Orange Pi aparece com IP diferente
```

Soluções:

```
# Verificar configuração de rede
jq '.projects.ender3.network' configs/projects-config.json

# Verificar conflito de IP
nmap -sn 192.168.1.100

# Alterar IP se necessário
jq '.projects.ender3.network.static_ip = "192.168.1.150"' \
  configs/projects-config.json > temp.json
mv temp.json configs/projects-config.json

# Verificar configuração do roteador
# Reservar IP no DHCP do roteador

# Regravar MicroSD com nova configuração
docker compose run --rm --privileged provisioner scripts/deploy-ender3.sh
```

SSH não funciona

Sintomas:

```
ssh: connect to host 192.168.1.100 port 22: Connection refused
# ou
Permission denied (publickey).
```

Soluções:

```
# Verificar se Orange Pi está acessível
ping 192.168.1.100

# Verificar porta SSH
nmap -p 22 192.168.1.100

# Verificar chave SSH local
ls -la ~/.ssh/id_rsa.pub
cat ~/.ssh/id_rsa.pub

# Regenerar chave SSH se necessário
ssh-keygen -t rsa -b 4096 -C "your.email@example.com"

# Tentar SSH com senha (se habilitado)
ssh -o PreferredAuthentications=password ender3@192.168.1.100

# Verificar logs SSH no Orange Pi (via monitor HDMI)
sudo journalctl -u ssh
```

Problemas de Deploy

Download da imagem Armbian falha

Sintomas:

```
curl: (6) Could not resolve host: redirect.armbian.com
# ou
curl: (28) Operation timed out
```

Soluções:

```
# Verificar conectividade
ping redirect.armbian.com
curl -I https://redirect.armbian.com

# Tentar download manual
wget https://redirect.armbian.com/orangepizero3/archive/Armbian_23.8.1_Orangepizero3_bookworm_current_6.1.47.img.xz

# Usar mirror alternativo
# Editar configs/projects-config.json
jq '.projects.ender3.software.armbian_image_url = "https://mirror.armbian.com/..."' \
  configs/projects-config.json > temp.json

# Verificar espaço em disco
df -h images/

# Limpar downloads antigos
rm -f images/Armbian_*.img*
```

Falha na gravação da imagem

Sintomas:

```
dd: error writing '/dev/sdb': Input/output error
# ou
xz: (stdin): File format not recognized
```

Soluções:

```
# Verificar integridade do arquivo baixado
xz -t images/Armbian_*.img.xz

# Verificar checksum se disponível
sha256sum images/Armbian_*.img.xz

# Tentar descompressão manual
xz -d images/Armbian_*.img.xz

# Verificar MicroSD
sudo badblocks -v /dev/sdb

# Usar dd com sync
sudo dd if=images/Armbian_*.img of=/dev/sdb bs=1M status=progress conv=fsync

# Verificar se gravação foi bem-sucedida
sudo dd if=/dev/sdb bs=512 count=1 | hexdump -C
```

Configuração não aplicada

Sintomas:

- Orange Pi inicializa mas configurações não estão aplicadas
- Usuário padrão não foi criado
- Serviços não estão rodando

Soluções:

```
# Verificar se first_run foi executado
# Conectar monitor HDMI e verificar logs

# Verificar arquivos de configuração na partição boot
sudo mkdir -p /mnt/sdcard
sudo mount /dev/sdb1 /mnt/sdcard
ls -la /mnt/sdcard/
cat /mnt/sdcard/armbian_first_run.txt

# Verificar se configuração foi aplicada
sudo mount /dev/sdb2 /mnt/sdcard
ls -la /mnt/sdcard/home/
cat /mnt/sdcard/etc/hostname

# Regravar com configuração corrigida
sudo umount /mnt/sdcard
docker compose run --rm --privileged provisioner scripts/deploy-ender3.sh
```

Problemas de Validação

Validação SSH falha

Sintomas:

✗ SSH connection failed **for** ender3@192.168.1.100

Soluções:

```
# Aguardar mais tempo para inicialização
sleep 300 # 5 minutos

# Verificar se Orange Pi terminou primeira inicialização
ping 192.168.1.100

# Tentar SSH manual
ssh -v ender3@192.168.1.100

# Verificar chave SSH
ssh-keygen -R 192.168.1.100 # Remove entrada antiga
ssh-keyscan 192.168.1.100 >> ~/.ssh/known_hosts

# Verificar logs de validação
cat logs/validate-*.log
```

Serviços não estão rodando

Sintomas:

✗ Service klipper is not running
✗ Service nginx is not running

Soluções:

```
# Conectar via SSH e verificar serviços
ssh ender3@192.168.1.100
sudo systemctl status klipper
sudo systemctl status nginx

# Verificar logs dos serviços
sudo journalctl -u klipper -f
sudo journalctl -u nginx -f

# Reiniciar serviços se necessário
sudo systemctl restart klipper
sudo systemctl restart nginx

# Verificar dependências
sudo systemctl list-dependencies klipper

# Verificar configuração
sudo nano /etc/systemd/system/klipper.service
```

Portas não estão acessíveis

Sintomas:

✗ Port 80 is not accessible on 192.168.1.100

Soluções:

```
# Verificar se serviço está rodando
ssh ender3@192.168.1.100 'sudo netstat -tlnp | grep :80'

# Verificar firewall
ssh ender3@192.168.1.100 'sudo ufw status'

# Desabilitar firewall temporariamente
ssh ender3@192.168.1.100 'sudo ufw disable'

# Verificar configuração do nginx
ssh ender3@192.168.1.100 'sudo nginx -t'

# Testar conectividade local
ssh ender3@192.168.1.100 'curl -I http://localhost'

# Verificar roteamento
traceroute 192.168.1.100
```



Logs e Diagnóstico

Localização dos Logs

```
# Logs do sistema
logs/
├─ deploy-ender3-YYYYMMDD_HHMMSS.log
├─ deploy-laser-YYYYMMDD_HHMMSS.log
├─ validate-YYYYMMDD_HHMMSS.log
└─ provision-manager-YYYYMMDD_HHMMSS.log

# Estados dos deployments
state/
├─ local-info.json
├─ ender3-deployment.json
├─ laser-deployment.json
└─ ender3-config.json

# Relatórios de validação
reports/
├─ validation_report_YYYYMMDD_HHMMSS.md
└─ test_report_YYYYMMDD_HHMMSS.md
```

Comandos de Diagnóstico

```
# Verificar logs em tempo real
tail -f logs/deploy-ender3-*.log

# Buscar erros nos logs
grep -i error logs/*.log
grep -i fail logs/*.log

# Verificar estado do sistema
docker compose ps
docker system df
docker images

# Verificar conectividade
ping -c 4 192.168.1.100
nmap -sn 192.168.1.0/24
nmap -p 22,80,443 192.168.1.100

# Verificar recursos do sistema
free -h
df -h
lsblk
lsusb
```

Modo Debug

```
# Habilitar debug em scripts
export DEBUG=true
export VERBOSE=true

# Executar com debug
docker compose run --rm provisioner bash -x scripts/deploy-ender3.sh

# Habilitar debug no Docker
export DOCKER_BUILDKIT=0
docker compose build --progress=plain
```

? FAQ

P: O deploy demora muito tempo, é normal?

R: Sim, o tempo varia conforme:

- Velocidade da internet (download da imagem): 2-10 minutos
- Velocidade do MicroSD (gravação): 3-8 minutos
- Primeira inicialização do Orange Pi: 5-10 minutos
- **Total esperado:** 10-30 minutos

P: Posso usar outros modelos de Orange Pi?

R: Atualmente suportamos:

- ☒ Orange Pi Zero 3 (2GB)
- ☒ Orange Pi Zero 2W (1GB)
- ☒ Outros modelos requerem adaptação das configurações

P: Como adicionar um novo projeto?

R:

1. Editar `configs/projects-config.json`
2. Criar script `scripts/deploy-novo-projeto.sh`
3. Atualizar `scripts/provision-manager.sh`
4. Adicionar validações específicas
5. Documentar no `USE_CASES.md`

P: Posso usar com Raspberry Pi?

R: Não diretamente. O sistema foi otimizado para Orange Pi com Armbian. Para Raspberry Pi seria necessário:

- Adaptar para Raspberry Pi OS
- Modificar configurações de hardware
- Ajustar scripts de deploy

P: Como fazer backup das configurações?




R:

```
# Backup completo
tar -czf backup-$(date +%Y%m%d).tar.gz configs/ state/ scripts/

# Backup apenas configurações
cp -r configs/ configs-backup-$(date +%Y%m%d)/
```

P: O sistema funciona offline?

R: Parcialmente:

-  Scripts e validações funcionam offline
-  Download de imagens Armbian requer internet
-  Instalação de pacotes requer internet
- **Solução:** Fazer cache das imagens localmente

P: Como contribuir com o projeto?

R:

1. Fork do repositório
2. Criar branch para sua feature
3. Seguir guidelines do `DEVELOPMENT.md`
4. Executar testes completos
5. Criar Pull Request
6. Aguardar review

P: Onde reportar bugs?

R:

1. Verificar se não é problema conhecido neste guia
2. Coletar logs relevantes
3. Criar issue no GitHub com:
 - Descrição detalhada
 - Passos para reproduzir
 - Logs e screenshots
 - Informações do ambiente

Nota: Este guia é atualizado regularmente. Se você encontrou um problema não documentado aqui, por favor contribua adicionando a solução após resolvê-lo.