内网穿透技术详解 - FRP 实战指南

> 一站式 FRP 内网穿透部署指南，从零开始搭建属于你的内网穿透服务

目录

什么是内网穿透?

基本概念

内网穿透（Intranet Penetration）是一种网络技术，它允许外网用户访问位于内网（局域网）中的服务或设备。

为什么需要内网穿透?

在日常开发和运维中，我们经常遇到以下场景：

- **家庭场景**：在公司想访问家里的 NAS 存储、树莓派等设备
- **开发场景**：需要让客户或团队成员访问本地开发环境
- **运维场景**：需要远程管理没有公网 IP 的服务器
- **游戏场景**：搭建私人游戏服务器供朋友联机

问题的本质

传统困境：
外网用户 → [公网] → [NAT路由器] → 无法直接访问内网设备

内网穿透解决方案：
外网用户 → [公网服务器(中转)] → [内网设备主动连接] → 成功访问

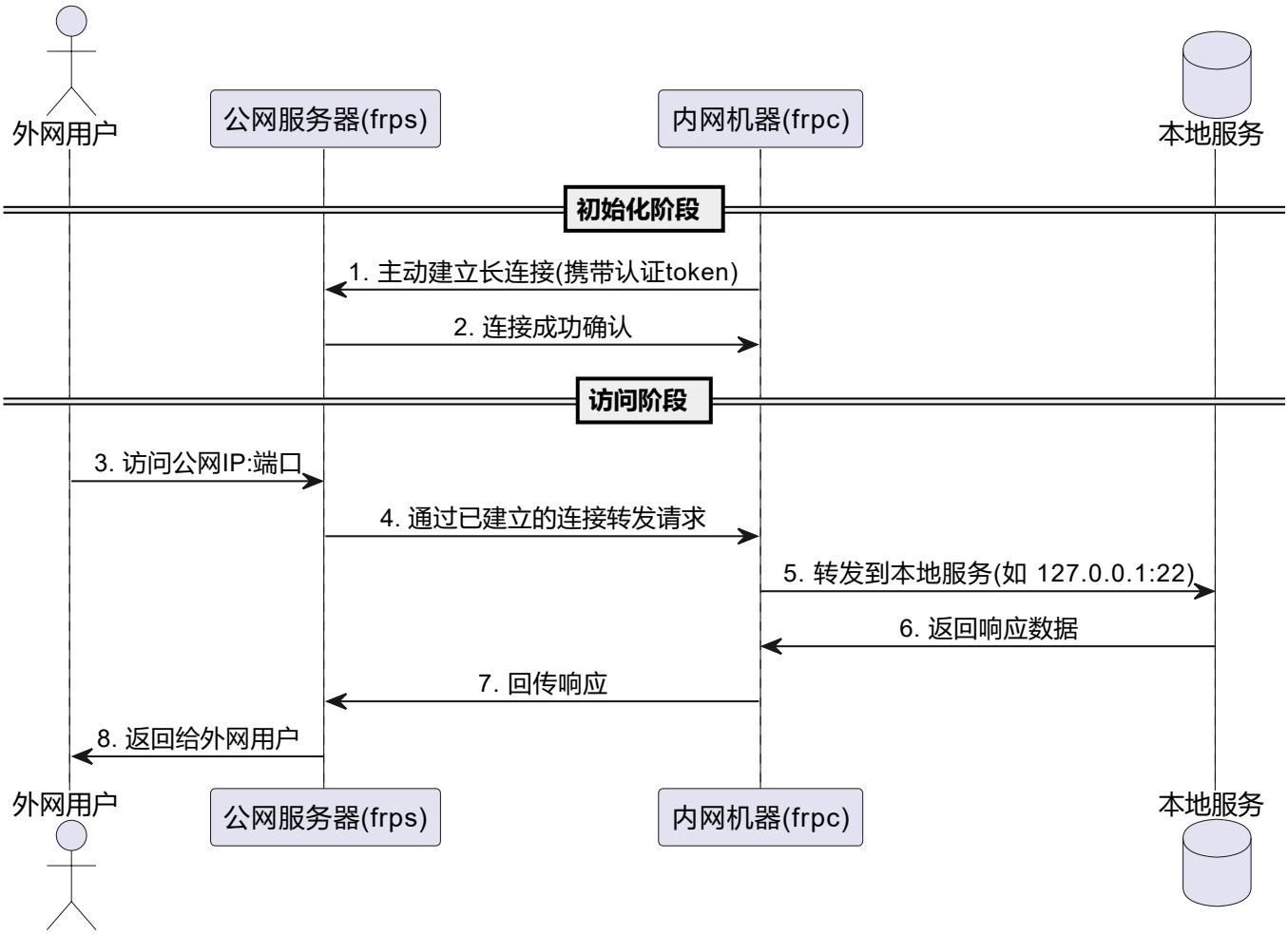
FRP 工作原理

FRP（Fast Reverse Proxy）是一个高性能的反向代理应用，专注于内网穿透。

核心组件

1. **frps（服务端）**：部署在有公网 IP 的服务器上，负责接收和转发流量
2. **frpc（客户端）**：部署在内网机器上，主动连接 frps 并暴露本地服务

工作流程图



通俗理解

可以把 FRP 想象成一个"快递中转站"：

- **公网服务器（frps）**：就像快递中转站，有固定地址，谁都能找到
- **内网机器（frpc）**：就像你家，主动去中转站"租了个柜子"
- **外网用户**：想给你寄东西的人，只需要把快递寄到中转站

- **数据传输**：中转站会自动把快递放到你租的柜子里，你随时可以取

详细配置说明

服务端配置（frps）

关键配置：

部署在公网服务器（如阿里云）上，配置文件 `frps.toml`：

```
# 服务端监听配置
bindAddr = "0.0.0.0"       # 监听所有网卡
bindPort = 7000            # frpc 客户端连接的端口

# 认证配置 - 安全第一!
auth.method = "token"
auth.token = "your_secure_token_123456"  # ⚠ 务必修改为复杂密码

# Web 管理面板配置
webServer.addr = "0.0.0.0"
webServer.port = 7500                    # 访问 http://服务器IP:7500 查看状态
webServer.user = "admin"
webServer.password = "your_dashboard_password"  # ⚠ 务必修改

# 端口白名单（可选，建议配置）
allowPorts = [
  { start = 6000, end = 60000 }  # 只允许客户端使用 6000-60000 端口
]

# 日志配置
log.to = "./frps.log"
log.level = "info"
log.maxDays = 3
```

配置项说明

配置项	说明	建议
`bindPort`	frpc 连接端口	默认 7000，需在防火墙开放
`auth.token`	认证密钥	必须设置强密码，客户端需一致
`webServer.port`	管理面板端口	可通过浏览器查看连接状态
`allowPorts`	允许的端口范围	限制端口范围，提高安全性

启动服务端

方式1：前台运行（测试用）

```
./frps -c frps.toml
```

方式2：后台运行

```
nohup ./frps -c frps.toml > frps.log 2>&1 &
```

方式3：systemd 管理（推荐生产环境）

创建服务文件：

```
sudo vim /etc/systemd/system/frps.service
```

配置内容：

```
[Unit]
Description=Frp Server Service - Fast Reverse Proxy for Intranet Penetration
Documentation=https://gofrp.org/
After=network-online.target syslog.target
Wants=network-online.target

[Service]
Type=simple
# 修改为你的 frps 实际路径
ExecStart=/root/frp_0.65.0_linux_amd64/frps -c
/root/frp_0.65.0_linux_amd64/frps.toml
# 工作目录（与 ExecStart 路径保持一致）
WorkingDirectory=/root/frp_0.65.0_linux_amd64
# 以 root 用户运行（如需其他用户，修改此项）
User=root
Group=root
# 自动重启配置
Restart=on-failure
RestartSec=10s
# 启动超时时间
TimeoutStartSec=30s
# 标准输出和错误输出到日志
StandardOutput=journal
StandardError=journal
# 进程限制（可选）
LimitNOFILE=65536

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

启动并设置开机自启：

```
# 重新加载 systemd 配置
sudo systemctl daemon-reload

# 启动服务
sudo systemctl start frps

# 设置开机自启动
sudo systemctl enable frps

# 查看服务状态
sudo systemctl status frps

# 查看实时日志
sudo journalctl -u frps -f

# 其他常用命令
sudo systemctl stop frps      # 停止服务
sudo systemctl restart frps   # 重启服务
sudo systemctl disable frps   # 取消开机自启
```

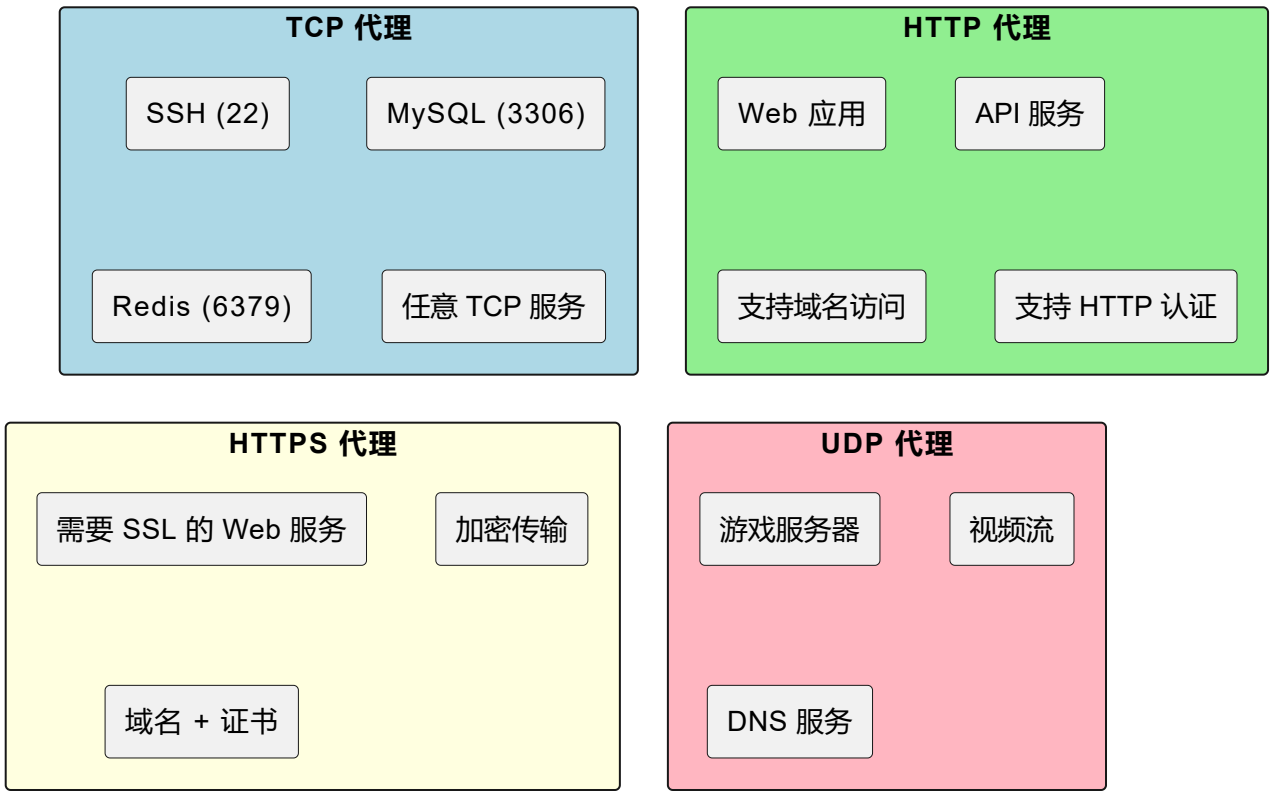
客户端配置（frpc）

部署在内网机器上，配置文件 `frpc.toml`：

```
# 服务端连接配置
serverAddr = "8.146.198.118"  # 公网服务器 IP
serverPort = 7000             # 与 frps 的 bindPort 一致

# 认证配置（必须与服务端一致）
auth.method = "token"
auth.token = "your_secure_token_123456"

# ============ 代理配置示例 ============

# 示例1：暴露 SSH 服务（远程登录内网机器）
[[proxies]]
name = "ssh"                  # 代理名称，自定义
type = "tcp"                  # 协议类型：tcp/udp/http/https
localIP = "127.0.0.1"         # 本地服务地址
localPort = 22                # 本地 SSH 端口
remotePort = 6000             # 公网服务器映射端口
# 可选：启用加密和压缩
transport.useEncryption = true
transport.useCompression = true

# 示例2：暴露 Web 服务（HTTP）
[[proxies]]
name = "web"
type = "http"
localIP = "127.0.0.1"
```

```
localPort = 8080                # 本地 Web 服务端口
customDomains = ["dev.example.com"]  # 自定义域名（需 DNS 解析到公网服务器）
# 可选：HTTP 基础认证
httpUser = "admin"
httpPassword = "admin123"

# 示例3：暴露数据库服务（如 MySQL）
[[proxies]]
name = "mysql"
type = "tcp"
localIP = "127.0.0.1"
localPort = 3306
remotePort = 6001
```

代理类型说明



访问方式对比

代理类型	访问方式	示例
TCP	公网IP:远程端口	`ssh root@8.146.198.118 -p 6000`
HTTP	http://域名	http://dev.example.com
HTTPS	https://域名	https://dev.example.com

启动客户端

方式1：前台运行（测试用）

```
./frpc -c frpc.toml
```

方式2：后台运行

```
nohup ./frpc -c frpc.toml > frpc.log 2>&1 &
```

方式3：systemd 管理（推荐）

创建服务文件：

```
sudo vim /etc/systemd/system/frpc.service
```

配置内容：

```
[Unit]
Description=Frp Client Service - Fast Reverse Proxy Client
Documentation=https://gofrp.org/
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
# 修改为你的 frpc 实际路径
ExecStart=/root/frp_0.65.0_linux_amd64/frpc -c
/root/frp_0.65.0_linux_amd64/frpc.toml
# 工作目录
WorkingDirectory=/root/frp_0.65.0_linux_amd64
# 运行用户
User=root
Group=root
# 自动重启配置（重要：确保内网穿透持续可用）
Restart=always
RestartSec=10s
# 启动超时
TimeoutStartSec=30s
# 日志输出
StandardOutput=journal
StandardError=journal
# 进程限制
LimitNOFILE=65536

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

启动并设置开机自启：

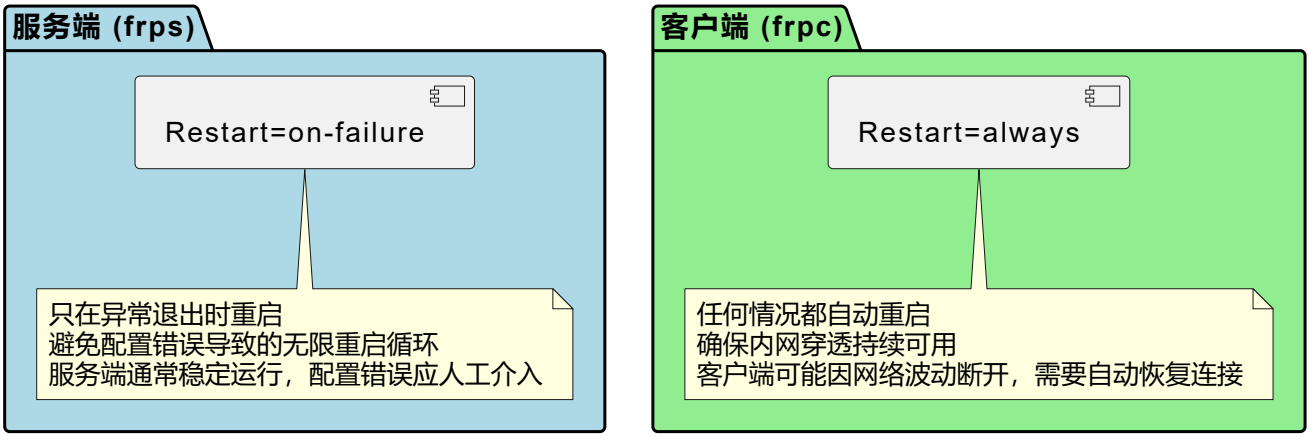
```
# 重新加载配置
sudo systemctl daemon-reload

# 启动客户端
sudo systemctl start frpc

# 设置开机自启动（重要！）
sudo systemctl enable frpc

# 查看运行状态
sudo systemctl status frpc

# 查看实时日志
sudo journalctl -u frpc -f

# 其他管理命令
sudo systemctl stop frpc      # 停止
sudo systemctl restart frpc   # 重启
sudo systemctl disable frpc   # 取消自启
```

Systemd 服务配置详解

配置项说明

配置项	说明	推荐值
`After=network-online.target`	等待网络完全就绪后启动	必须配置
`Wants=network-online.target`	依赖网络服务	必须配置
`Restart=always/on-failure`	自动重启策略	客户端用 `always`，服务端用 `on-failure`
`RestartSec=10s`	重启间隔时间	10-30秒
`TimeoutStartSec=30s`	启动超时时间	30秒
`StandardOutput=journal`	日志输出到 systemd journal	推荐
`LimitNOFILE=65536`	最大文件描述符数	高并发场景必须设置

服务端与客户端配置差异



快速部署脚本

服务端一键部署脚本

创建部署脚本 `frps-install.sh`：

```
#!/bin/bash
# frps-install.sh - FRP 服务端一键部署脚本
# 使用方法: sudo bash frps-install.sh

set -e  # 遇到错误立即退出

# ============ 配置区域（可修改） ============
FRP_VERSION="0.65.0"
INSTALL_DIR="/usr/local/frp"
AUTH_TOKEN=""  # 留空则自动生成
WEB_USER="admin"
WEB_PASSWORD=""  # 留空则自动生成
BIND_PORT=7000
WEB_PORT=7500
ALLOW_PORT_START=6000
ALLOW_PORT_END=60000

# ============ 颜色输出 ============
RED='\033[0;31m'
GREEN='\033[0;32m'
YELLOW='\033[1;33m'
NC='\033[0m' # No Color

echo_info() { echo -e "${GREEN}[INFO]${NC} $1"; }
echo_warn() { echo -e "${YELLOW}[WARN]${NC} $1"; }
echo_error() { echo -e "${RED}[ERROR]${NC} $1"; }

# ============ 检查权限 ============
if [[ $EUID -ne 0 ]]; then
   echo_error "此脚本需要 root 权限运行"
   echo "请使用: sudo bash $0"
   exit 1
fi
```

```
# ============ 生成随机密码 ============
generate_password() {
    openssl rand -base64 32 | tr -d "=+/" | cut -c1-25
}

# 如果未设置密码，自动生成
if [ -z "$AUTH_TOKEN" ]; then
    AUTH_TOKEN=$(generate_password)
    echo_info "已自动生成认证 Token: $AUTH_TOKEN"
fi

if [ -z "$WEB_PASSWORD" ]; then
    WEB_PASSWORD=$(generate_password)
    echo_info "已自动生成 Web 密码: $WEB_PASSWORD"
fi

# ============ 检查并安装依赖 ============
echo_info "检查系统依赖..."
if ! command -v wget &> /dev/null; then
    echo_warn "wget 未安装，正在安装..."
    apt-get update && apt-get install -y wget || yum install -y wget
fi

# ============ 下载 FRP ============
echo_info "开始下载 FRP v${FRP_VERSION}..."
cd /tmp
DOWNLOAD_URL="https://github.com/fatedier/frp/releases/download/v${FRP_VERSION}/fr
p_${FRP_VERSION}_linux_amd64.tar.gz"

if [ -f "frp_${FRP_VERSION}_linux_amd64.tar.gz" ]; then
    echo_warn "安装包已存在，跳过下载"
else
    wget -O frp_${FRP_VERSION}_linux_amd64.tar.gz "$DOWNLOAD_URL" || {
        echo_error "下载失败，请检查网络或版本号"
        exit 1
    }
fi

# ============ 解压并安装 ============
echo_info "解压安装包..."
tar -xzf frp_${FRP_VERSION}_linux_amd64.tar.gz

# 备份旧版本
if [ -d "$INSTALL_DIR" ]; then
    echo_warn "检测到旧版本，备份到 ${INSTALL_DIR}.bak"
    mv "$INSTALL_DIR" "${INSTALL_DIR}.bak.$(date +%Y%m%d_%H%M%S)"
fi

mv frp_${FRP_VERSION}_linux_amd64 "$INSTALL_DIR"
echo_info "安装到: $INSTALL_DIR"

# ============ 创建配置文件 ============
echo_info "生成配置文件..."
```

```
cat > ${INSTALL_DIR}/frps.toml <<EOF
# FRP 服务端配置文件
# 生成时间: $(date '+%Y-%m-%d %H:%M:%S')

# 服务端监听配置
bindAddr = "0.0.0.0"
bindPort = ${BIND_PORT}

# 认证配置
auth.method = "token"
auth.token = "${AUTH_TOKEN}"

# Web 管理面板
webServer.addr = "0.0.0.0"
webServer.port = ${WEB_PORT}
webServer.user = "${WEB_USER}"
webServer.password = "${WEB_PASSWORD}"

# 端口白名单
allowPorts = [
  { start = ${ALLOW_PORT_START}, end = ${ALLOW_PORT_END} }
]

# 日志配置
log.to = "${INSTALL_DIR}/frps.log"
log.level = "info"
log.maxDays = 7
EOF

# ============ 创建 systemd 服务 ============
echo_info "创建 systemd 服务..."
cat > /etc/systemd/system/frps.service <<EOF
[Unit]
Description=Frp Server Service - Fast Reverse Proxy for Intranet Penetration
Documentation=https://gofrp.org/
After=network-online.target syslog.target
Wants=network-online.target

[Service]
Type=simple
ExecStart=${INSTALL_DIR}/frps -c ${INSTALL_DIR}/frps.toml
WorkingDirectory=${INSTALL_DIR}
User=root
Group=root
Restart=on-failure
RestartSec=10s
TimeoutStartSec=30s
StandardOutput=journal
StandardError=journal
LimitNOFILE=65536

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

```
# ============ 配置防火墙 ============
echo_info "配置防火墙规则..."
if command -v ufw &> /dev/null; then
    ufw allow ${BIND_PORT}/tcp comment 'FRP Server'
    ufw allow ${WEB_PORT}/tcp comment 'FRP Web Panel'
    ufw allow ${ALLOW_PORT_START}:${ALLOW_PORT_END}/tcp comment 'FRP Proxy Ports'
    echo_info "UFW 防火墙规则已添加"
elif command -v firewall-cmd &> /dev/null; then
    firewall-cmd --permanent --add-port=${BIND_PORT}/tcp
    firewall-cmd --permanent --add-port=${WEB_PORT}/tcp
    firewall-cmd --permanent --add-port=${ALLOW_PORT_START}-${ALLOW_PORT_END}/tcp
    firewall-cmd --reload
    echo_info "Firewalld 防火墙规则已添加"
else
    echo_warn "未检测到防火墙，请手动开放端口: ${BIND_PORT}, ${WEB_PORT},
${ALLOW_PORT_START}-${ALLOW_PORT_END}"
fi

# ============ 启动服务 ============
echo_info "启动 FRP 服务..."
systemctl daemon-reload
systemctl enable frps
systemctl start frps

# ============ 检查服务状态 ============
sleep 2
if systemctl is-active --quiet frps; then
    echo_info "☑ FRP 服务端安装成功并已启动！"
else
    echo_error "❌ 服务启动失败，请查看日志: journalctl -u frps -n 50"
    exit 1
fi

# ============ 输出配置信息 ============
SERVER_IP=$(curl -s ifconfig.me || echo "获取失败")

echo ""
echo "========================================="
echo -e "${GREEN}🎉 FRP 服务端部署完成！${NC}"
echo "========================================="
echo ""
echo "📁 安装目录: $INSTALL_DIR"
echo "📝 配置文件: ${INSTALL_DIR}/frps.toml"
echo "📊 日志文件: ${INSTALL_DIR}/frps.log"
echo ""
echo "🔐 认证信息（请妥善保管）:"
echo "   Token: $AUTH_TOKEN"
echo ""
echo "🌐 Web 管理面板:"
echo "   地址: http://${SERVER_IP}:${WEB_PORT}"
echo "   用户: $WEB_USER"
echo "   密码: $WEB_PASSWORD"
echo ""
```

```
echo "🔧 常用命令:"
echo "   查看状态: systemctl status frps"
echo "   查看日志: journalctl -u frps -f"
echo "   重启服务: systemctl restart frps"
echo "   停止服务: systemctl stop frps"
echo "   修改配置: vim ${INSTALL_DIR}/frps.toml"
echo ""
echo "⚠️  重要提示:"
echo "   1. 请将 Token 配置到客户端"
echo "   2. 建议修改 Web 管理面板密码"
echo "   3. 确保云服务器安全组已开放相关端口"
echo "========================================="

# 保存配置信息到文件
cat > ${INSTALL_DIR}/install_info.txt <<EOF
FRP 服务端安装信息
安装时间: $(date '+%Y-%m-%d %H:%M:%S')
服务器IP: ${SERVER_IP}
认证Token: ${AUTH_TOKEN}
Web用户: ${WEB_USER}
Web密码: ${WEB_PASSWORD}
Web地址: http://${SERVER_IP}:${WEB_PORT}
EOF

chmod 600 ${INSTALL_DIR}/install_info.txt
echo_info "配置信息已保存到: ${INSTALL_DIR}/install_info.txt"
```

客户端一键部署脚本

创建部署脚本 `frpc-install.sh`:

```
#!/bin/bash
# frpc-install.sh - FRP 客户端一键部署脚本
# 使用方法: sudo bash frpc-install.sh

set -e  # 遇到错误立即退出

# ============ 配置区域（必须修改） ============
SERVER_IP=""           # 公网服务器 IP（必填）
AUTH_TOKEN=""          # 服务端的认证 Token（必填）
FRP_VERSION="0.65.0"
INSTALL_DIR="/usr/local/frp"
SERVER_PORT=7000

# ============ 颜色输出 ============
RED='\033[0;31m'
GREEN='\033[0;32m'
YELLOW='\033[1;33m'
BLUE='\033[0;34m'
NC='\033[0m'
```

```
echo_info() { echo -e "${GREEN}[INFO]${NC} $1"; }
echo_warn() { echo -e "${YELLOW}[WARN]${NC} $1"; }
echo_error() { echo -e "${RED}[ERROR]${NC} $1"; }
echo_step() { echo -e "${BLUE}[STEP]${NC} $1"; }

# ============ 检查权限 ============
if [[ $EUID -ne 0 ]]; then
   echo_error "此脚本需要 root 权限运行"
   echo "请使用: sudo bash $0"
   exit 1
fi

# ============ 交互式配置 ============
echo "=========================================="
echo "FRP 客户端一键部署脚本"
echo "=========================================="
echo ""

if [ -z "$SERVER_IP" ]; then
    read -p "请输入服务端 IP 地址: " SERVER_IP
    if [ -z "$SERVER_IP" ]; then
        echo_error "服务端 IP 不能为空"
        exit 1
    fi
fi

if [ -z "$AUTH_TOKEN" ]; then
    read -p "请输入服务端的认证 Token: " AUTH_TOKEN
    if [ -z "$AUTH_TOKEN" ]; then
        echo_error "认证 Token 不能为空"
        exit 1
    fi
fi

echo ""
echo_info "配置信息确认:"
echo "  服务端 IP: $SERVER_IP"
echo "  服务端端口: $SERVER_PORT"
echo "  认证 Token: ${AUTH_TOKEN:0:10}..."
echo ""
read -p "确认继续安装? (y/N): " confirm
if [[ ! $confirm =~ ^[Yy]$ ]]; then
    echo "安装已取消"
    exit 0
fi

# ============ 检查依赖 ============
echo_step "检查系统依赖..."
if ! command -v wget &> /dev/null; then
    echo_warn "wget 未安装，正在安装..."
    apt-get update && apt-get install -y wget || yum install -y wget
fi

# ============ 下载 FRP ============
```

```
echo_step "下载 FRP v${FRP_VERSION}..."
cd /tmp
DOWNLOAD_URL="https://github.com/fatedier/frp/releases/download/v${FRP_VERSION}/fr
p_${FRP_VERSION}_linux_amd64.tar.gz"

if [ -f "frp_${FRP_VERSION}_linux_amd64.tar.gz" ]; then
    echo_warn "安装包已存在，跳过下载"
else
    wget -O frp_${FRP_VERSION}_linux_amd64.tar.gz "$DOWNLOAD_URL" || {
        echo_error "下载失败，尝试使用国内镜像..."
        # 可以添加国内镜像地址
        exit 1
    }
fi

# ============ 解压安装 ============
echo_step "解压安装包..."
tar -xzf frp_${FRP_VERSION}_linux_amd64.tar.gz

# 备份旧版本
if [ -d "$INSTALL_DIR" ]; then
    echo_warn "检测到旧版本，备份到 ${INSTALL_DIR}.bak"
    systemctl stop frpc 2>/dev/null || true
    mv "$INSTALL_DIR" "${INSTALL_DIR}.bak.$(date +%Y%m%d_%H%M%S)"
fi

mv frp_${FRP_VERSION}_linux_amd64 "$INSTALL_DIR"
echo_info "安装到: $INSTALL_DIR"

# ============ 配置代理服务 ============
echo_step "配置代理服务..."
echo ""
echo "请选择要配置的代理类型:"
echo "  1) SSH (端口 22)"
echo "  2) Web 服务 (HTTP)"
echo "  3) 自定义配置"
read -p "请选择 (1-3): " proxy_type

case $proxy_type in
    1)
        read -p "请输入远程端口 (默认 6000): " remote_port
        remote_port=${remote_port:-6000}

        cat > ${INSTALL_DIR}/frpc.toml <<EOF
# FRP 客户端配置文件
# 生成时间: $(date '+%Y-%m-%d %H:%M:%S')

serverAddr = "${SERVER_IP}"
serverPort = ${SERVER_PORT}

auth.method = "token"
auth.token = "${AUTH_TOKEN}"

# SSH 代理
```

```
[[proxies]]
name = "ssh"
type = "tcp"
localIP = "127.0.0.1"
localPort = 22
remotePort = ${remote_port}
transport.useEncryption = true
transport.useCompression = true
EOF
        echo_info "已配置 SSH 代理，远程端口: ${remote_port}"
        ;;
    2)
        read -p "请输入本地 Web 端口 (默认 8080): " local_port
        local_port=${local_port:-8080}
        read -p "请输入自定义域名 (如 dev.example.com): " custom_domain

        cat > ${INSTALL_DIR}/frpc.toml <<EOF
# FRP 客户端配置文件
serverAddr = "${SERVER_IP}"
serverPort = ${SERVER_PORT}

auth.method = "token"
auth.token = "${AUTH_TOKEN}"

# Web 服务代理
[[proxies]]
name = "web"
type = "http"
localIP = "127.0.0.1"
localPort = ${local_port}
customDomains = ["${custom_domain}"]
EOF
        echo_info "已配置 Web 代理，本地端口: ${local_port}，域名: ${custom_domain}"
        ;;
    3)
        cat > ${INSTALL_DIR}/frpc.toml <<EOF
# FRP 客户端配置文件
serverAddr = "${SERVER_IP}"
serverPort = ${SERVER_PORT}

auth.method = "token"
auth.token = "${AUTH_TOKEN}"

# 请手动添加代理配置
# [[proxies]]
# name = "your-service"
# type = "tcp"
# localIP = "127.0.0.1"
# localPort = 22
# remotePort = 6000
EOF
        echo_warn "已创建基础配置，请手动编辑: ${INSTALL_DIR}/frpc.toml"
        ;;
    *)
```

```
        echo_error "无效选择"
        exit 1
        ;;
esac

# ============ 创建 systemd 服务 ============
echo_step "创建 systemd 服务..."
cat > /etc/systemd/system/frpc.service <<EOF
[Unit]
Description=Frp Client Service - Fast Reverse Proxy Client
Documentation=https://gofrp.org/
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
ExecStart=${INSTALL_DIR}/frpc -c ${INSTALL_DIR}/frpc.toml
WorkingDirectory=${INSTALL_DIR}
User=root
Group=root
Restart=always
RestartSec=10s
TimeoutStartSec=30s
StandardOutput=journal
StandardError=journal
LimitNOFILE=65536

[Install]
WantedBy=multi-user.target
EOF

# ============ 启动服务 ============
echo_step "启动 FRP 客户端..."
systemctl daemon-reload
systemctl enable frpc
systemctl start frpc

# ============ 检查服务状态 ============
sleep 2
if systemctl is-active --quiet frpc; then
    echo_info "✅ FRP 客户端安装成功并已启动！"
else
    echo_error "❌ 服务启动失败，请查看日志：journalctl -u frpc -n 50"
    exit 1
fi

# ============ 输出配置信息 ============
LOCAL_IP=$(hostname -I | awk '{print $1}')

echo ""
echo "========================================="
echo -e "${GREEN}🎉 FRP 客户端部署完成！${NC}"
echo "========================================="
echo ""
```

```
    echo "📁 安装目录: $INSTALL_DIR"
    echo "📝 配置文件: ${INSTALL_DIR}/frpc.toml"
    echo "🖥  本机 IP: $LOCAL_IP"
    echo "🌐 服务端: ${SERVER_IP}:${SERVER_PORT}"
    echo ""
    echo "🔧 常用命令:"
    echo "   查看状态: systemctl status frpc"
    echo "   查看日志: journalctl -u frpc -f"
    echo "   重启服务: systemctl restart frpc"
    echo "   停止服务: systemctl stop frpc"
    echo "   修改配置: vim ${INSTALL_DIR}/frpc.toml"
    echo ""
    echo "⚠️  重要提示:"
    echo "   1. 修改配置后需重启服务: systemctl restart frpc"
    echo "   2. 查看连接状态: journalctl -u frpc -f"
    echo "   3. 确保服务端已正确配置并运行"
    echo "========================================="

    # 保存配置信息
    cat > ${INSTALL_DIR}/install_info.txt <<EOF
    FRP 客户端安装信息
    安装时间: $(date '+%Y-%m-%d %H:%M:%S')
    本机IP: ${LOCAL_IP}
    服务端: ${SERVER_IP}:${SERVER_PORT}
    配置文件: ${INSTALL_DIR}/frpc.toml
    EOF

    chmod 600 ${INSTALL_DIR}/install_info.txt
    echo_info "配置信息已保存到: ${INSTALL_DIR}/install_info.txt"
```

快速部署使用指南

方式一：直接执行（推荐）

服务端部署:

```
# 下载并执行脚本
curl -fsSL https://raw.githubusercontent.com/your-repo/frp-scripts/main/frps-install.sh | sudo bash

# 或者分步执行
wget https://raw.githubusercontent.com/your-repo/frp-scripts/main/frps-install.sh
chmod +x frps-install.sh
sudo bash frps-install.sh
```

客户端部署:

```
# 下载并执行（会提示输入服务端信息）
wget https://raw.githubusercontent.com/your-repo/frp-scripts/main/frpc-install.sh
```

```
chmod +x frpc-install.sh
sudo bash frpc-install.sh
```

方式二：本地创建脚本

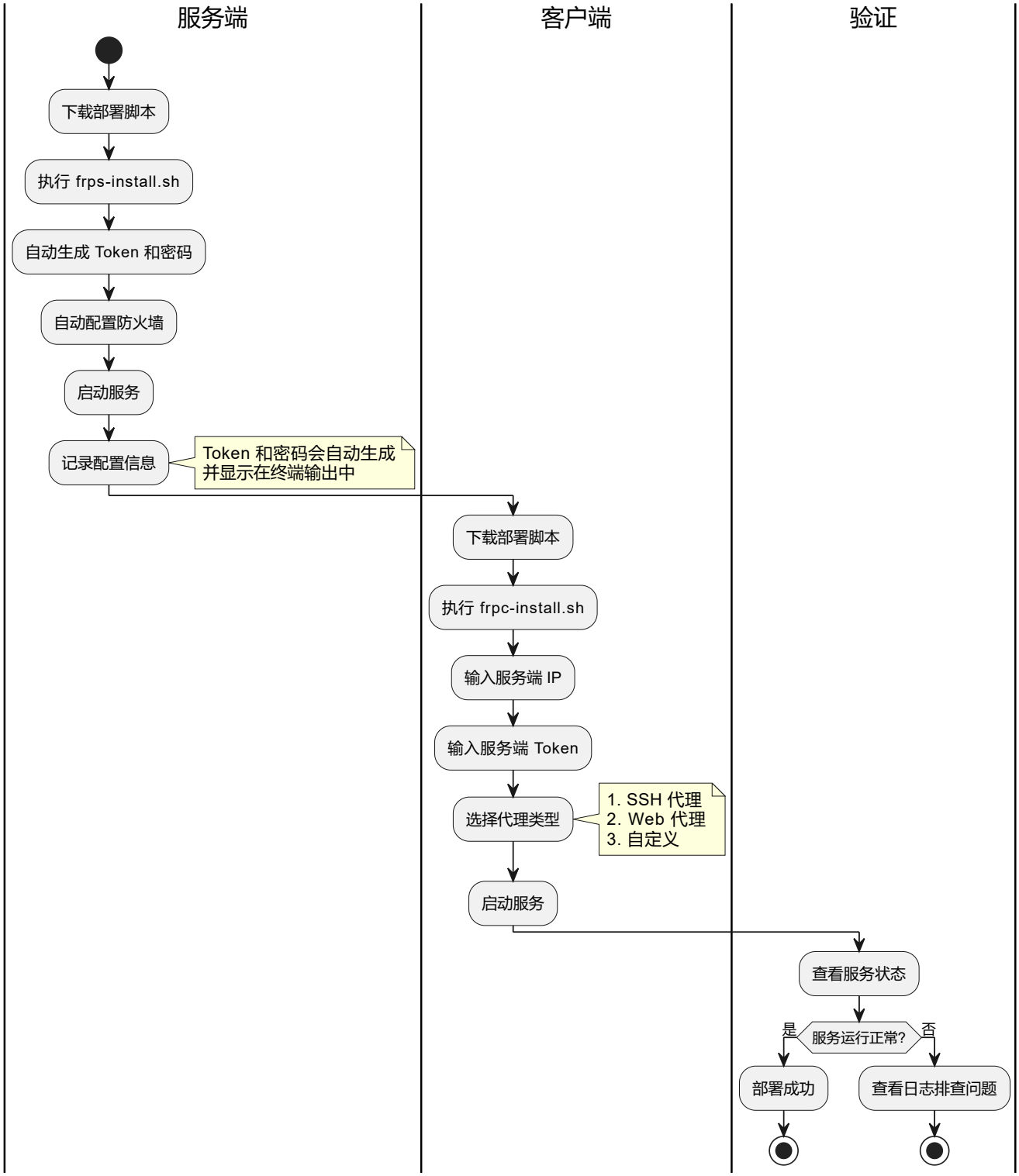
1. 创建服务端脚本

```
# 将上面的服务端脚本内容保存为 frps-install.sh
vim frps-install.sh
# 粘贴脚本内容，保存退出

# 添加执行权限
chmod +x frps-install.sh

# 执行安装
sudo bash frps-install.sh
```

2. 创建客户端脚本

```
# 将客户端脚本内容保存为 frpc-install.sh
vim frpc-install.sh
# 粘贴脚本内容，保存退出

# 执行安装（交互式配置）
chmod +x frpc-install.sh
sudo bash frpc-install.sh
```

部署流程图



部署后操作

服务端:

```
# 查看自动生成的配置信息
cat /usr/local/frp/install_info.txt

# 输出示例:
# FRP 服务端安装信息
# 安装时间: 2025-12-07 13:30:00
```

```
# 服务器IP: 8.146.198.118
# 认证Token: Xy9mK2pL8nQ5rT3vW6yZ1
# Web用户: admin
# Web密码: aB3dE5fG7hJ9kL2mN4pQ6
# Web地址: http://8.146.198.118:7500

# 访问 Web 管理面板
# 浏览器打开: http://8.146.198.118:7500
# 使用上面的用户名和密码登录

# 查看服务状态
systemctl status frps

# 实时查看日志
journalctl -u frps -f
```

客户端:

```
# 查看配置信息
cat /usr/local/frp/install_info.txt

# 查看服务状态
systemctl status frpc

# 查看连接日志
journalctl -u frpc -f

# 如果需要修改配置
vim /usr/local/frp/frpc.toml
systemctl restart frpc
```

常见部署场景

场景1：快速搭建 SSH 远程访问

```
# 服务端（公网服务器）
sudo bash frps-install.sh
# 记录输出的 Token

# 客户端（内网机器）
sudo bash frpc-install.sh
# 输入服务端 IP: 8.146.198.118
# 输入 Token: [刚才记录的 Token]
# 选择: 1 (SSH)
# 输入远程端口: 6000

# 测试连接
ssh root@8.146.198.118 -p 6000
```

场景2：暴露本地 Web 服务

```
# 客户端执行
sudo bash frpc-install.sh
# 输入服务端信息
# 选择：2（Web 服务）
# 输入本地端口：3000
# 输入域名：dev.example.com

# 配置 DNS
# 将 dev.example.com 的 A 记录指向服务端 IP

# 访问测试
curl http://dev.example.com
```

脚本特性说明

特性	服务端脚本	客户端脚本
自动生成密码	☑ Token 和 Web 密码	❌ 需手动输入
交互式配置	❌ 全自动	☑ 引导式配置
防火墙配置	☑ 自动配置	❌ 无需配置
备份旧版本	☑ 自动备份	☑ 自动备份
服务状态检查	☑ 自动检查	☑ 自动检查
配置信息保存	☑ install_info.txt	☑ install_info.txt
错误处理	☑ 完善	☑ 完善

故障排查

如果部署失败，按以下步骤检查：

```
# 1. 检查脚本执行日志
# 脚本会输出详细的错误信息

# 2. 检查服务状态
systemctl status frps  # 或 frpc
journalctl -u frps -n 50  # 查看最近 50 行日志

# 3. 检查网络连通性
# 服务端
netstat -tlnp | grep 7000  # 检查端口监听

# 客户端
telnet 8.146.198.118 7000  # 测试连接
```

```
# 4. 检查防火墙
# 服务端
ufw status  # 或 firewall-cmd --list-all

# 5. 手动启动测试
cd /usr/local/frp
./frps -c frps.toml  # 或 ./frpc -c frpc.toml
# 查看错误输出
```

🚀 实战案例

案例1：远程 SSH 登录内网服务器

场景：在公司需要登录家里的 Linux 服务器

配置：

```
# frpc.toml
[[proxies]]
name = "home-ssh"
type = "tcp"
localIP = "127.0.0.1"
localPort = 22
remotePort = 6000
```

使用：

```
# 在任何地方通过公网服务器访问
ssh root@8.146.198.118 -p 6000
```

案例2：展示本地开发的 Web 项目

场景：需要让客户预览本地开发的网站

配置：

```
# frpc.toml
[[proxies]]
name = "dev-web"
type = "http"
localIP = "127.0.0.1"
localPort = 3000  # 本地开发服务器端口
customDomains = ["demo.yourdomain.com"]
```

使用：

- 客户访问：`http://demo.yourdomain.com`
- 自动转发到你的本地 `localhost:3000`

案例3：远程访问内网数据库

场景：在外网需要连接公司内网的 MySQL 数据库

配置：

```
# frpc.toml
[[proxies]]
name = "company-mysql"
type = "tcp"
localIP = "192.168.1.100"  # 内网数据库服务器 IP
localPort = 3306
remotePort = 6001
transport.useEncryption = true  # 启用加密
```

使用：

```
# 使用 MySQL 客户端连接
mysql -h 8.146.198.118 -P 6001 -u root -p
```

安全建议

1. 认证安全

```
# ☑ 推荐：使用强密码
auth.token = "Xy9$mK2#pL8@nQ5&rT3!"

# ✖ 避免：简单密码
auth.token = "123456"
```

2. 端口限制

```
# 限制可用端口范围
allowPorts = [
  { start = 6000, end = 6100 }  # 只开放 100 个端口
]
```

3. 传输加密

```
# 对敏感服务启用加密
transport.useEncryption = true
transport.useCompression = true
```

4. HTTP 认证

```
# Web 服务添加基础认证
httpUser = "admin"
httpPassword = "complex_password_here"
```

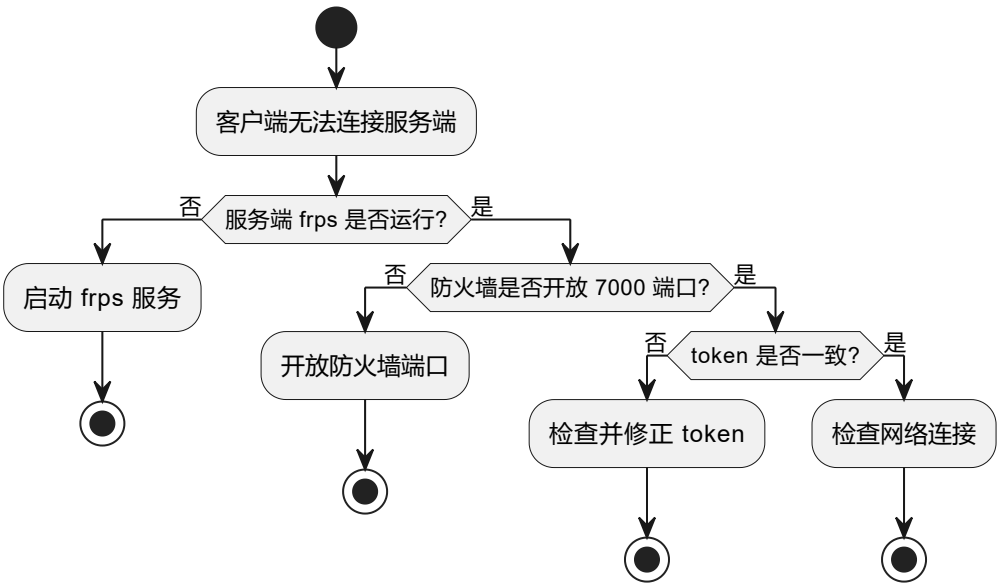
5. 防火墙配置

```
# 服务端只开放必要端口
sudo ufw allow 7000/tcp   # frp 服务端口
sudo ufw allow 7500/tcp   # Web 管理面板
sudo ufw allow 6000:6100/tcp  # 代理端口范围
```

故障排查

常见问题

1. 客户端连接失败



检查步骤:

```
# 1. 检查服务端是否运行
ps aux | grep frps
```

```
# 2. 检查端口监听
netstat -tlnp | grep 7000

# 3. 查看日志
tail -f frps.log
tail -f frpc.log

# 4. 测试网络连通性
telnet 8.146.198.118 7000
```

2. 无法访问内网服务

问题	可能原因	解决方案
连接超时	本地服务未启动	检查本地服务状态
连接被拒绝	端口配置错误	核对 localPort 和 remotePort
域名无法访问	DNS 未解析	检查域名 A 记录指向

3. 性能优化

```
# 启用连接池
transport.poolCount = 5

# 启用压缩（适合文本传输）
transport.useCompression = true

# 心跳设置
transport.heartbeatInterval = 30
transport.heartbeatTimeout = 90
```

监控与管理

Web 管理面板

访问 `http://8.146.198.118:7500`，可以查看：

- ☑ 在线客户端列表
- ☑ 代理连接状态
- ☑ 流量统计
- ☑ 实时日志

命令行监控

```
# 查看连接数
netstat -an | grep 7000 | wc -l
```

```
# 实时日志
tail -f frps.log | grep -E "error|warn"

# 查看进程状态
systemctl status frps
```

🎓 总结

FRP 的优势

- ☑ **简单易用**：配置简洁，上手快
- ☑ **性能优秀**：基于 Go 语言，高并发支持
- ☑ **功能丰富**：支持 TCP/UDP/HTTP/HTTPS 多种协议
- ☑ **安全可靠**：支持加密传输和认证
- ☑ **开源免费**：活跃的社区支持

适用场景

场景	推荐指数	说明
个人开发测试	★★★★★	完美适配
小团队协作	★★★★	性价比高
远程运维	★★★★★	安全便捷
生产环境	★★★	需配合其他安全措施

下一步

1. 📥 下载 FRP：https://github.com/fatedier/frp/releases
2. 📖 官方文档：https://gofrp.org/
3. 🔧 根据实际需求调整配置
4. 🌓 做好安全防护措施

提示：本文档基于 FRP v0.65.0 版本编写，不同版本配置可能略有差异。