

通用 VPS 三协议部署指南

VLESS Reality · Hysteria2 · Trojan · Mihomo

在一台具备公网入口的 Linux VPS 上，使用一个 sing-box 服务接收三种协议，再由 Mihomo 客户端按需选择连接。适用于任意地区与服务商；地区影响线路质量，不改变这里的基本部署结构。

默认路线：无需自有域名。Reality 使用独立密钥；Hysteria2 与 Trojan 使用自签名 TLS 证书，并在 Mihomo 中固定证书指纹。有自有域名时，可按第 12 至 13 页切换为公有 CA 证书。

默认端口安排

连接方式	公网监听	身份与信任
VLESS + Reality	TCP/443	UUID + Reality 公私钥 + Short ID
Hysteria2	UDP/443	独立密码 + TLS 证书指纹
Trojan over TLS	TCP/8443	独立密码 + TLS 证书指纹

Reality 与 Hysteria2 可以使用同一个数字端口，因为 TCP 与 UDP 分属不同监听空间。Trojan 也使用 TCP，因此默认另用 8443。三种协议共享一台服务器，提供的是连接方式选择，不是服务器容灾。

阅读与执行顺序

页码	内容
2 至 3	理解链路与适用条件，准备参数
4 至 7	系统准备、安装、认证材料与 TLS 证书
8 至 10	完整服务端 JSON、启动检查、完整客户端 YAML
11	逐一测试三种协议，再启用 AUTO
12 至 13	可选的自有域名证书与自动续期
14 至 17	排错、优化、维护、验收与官方资料

版本说明：按 2026 年 10 月 14 日的官方文档核对；实际版本以命令输出为准。本版已检查配置结构与字段对应关系，尚未在真实 VPS 上实测。

1 理解链路 & 部署边界

先分清代理协议 服务器和配置分发

应用先把流量交给 Mihomo；Mihomo 根据 PROXY 的选择连接 VPS；sing-box 完成认证与解封装后，通过 direct 出站访问目标服务。回包沿已建立的连接返回。

层级	作用	排错重点
应用与流量接管	系统代理或 TUN 决定哪些流量进入 Mihomo	浏览器通但其他应用不通，先检查接管方式
Mihomo 配置	保存节点参数、规则和策略组	能导入不等于参数正确，也不等于流量已接管
三种入站协议	承载客户端到 VPS 的连接	Reality 与 Trojan 走 TCP；HY2 走 QUIC/UDP
sing-box 与出站	接收连接并访问目标服务	所有节点同时失败时先看服务和 VPS 出网
云防火墙与系统防火墙	控制公网流量能否到达对应端口	两层都要放行；本机监听不代表公网可达
本地 YAML 或订阅	把相同连接参数交给设备	订阅只是配置分发，不承载代理流量

哪些 VPS 可以直接照做

- 主流程使用 Ubuntu 22.04 / 24.04 或 Debian 12 / 13，要求 APT、systemd 和管理员权限。其他发行版可沿用架构，但要替换安装、服务和防火墙命令。
- 默认具备可从客户端访问的公网 IPv4；NAT VPS 需要提供商映射相应 TCP 与 UDP 端口。若公网端口与内网端口不同，客户端填写公网端口，服务端填写实际监听端口。
- VPS 必须能正常访问外网、解析域名，并允许相关端口入站。只有 IPv6 的 VPS、已有网站占用 443、面板接管网络或容器部署，都需要额外调整。

如何选择协议

先用 Reality 验证 TCP 路径，再测 HY2，最后测 Trojan。HY2 的吞吐和丢包表现需要在实际网络测量，不能预设一定更快。Trojan 提供另一种 TLS 连接方式；如果当前网络只允许 TCP/443，TCP/8443 也可能被拦截。

三个节点都指向同一台 VPS：IP 不可达、VPS 停机、sing-box 故障或 VPS 出网故障时，切换协议可能没有帮助。

2 准备参数与命名

占位符必须替换 命令按执行位置区分

本文的服务端命令在 VPS 的 root shell 中执行；客户端测试在自己的电脑执行。可以先用普通管理账号 SSH 登录，再运行 `sudo -i`。所有尖括号占位符都要替换，不包含尖括号本身。

参数	用途与取值
<VPS_IP>	VPS 公网 IPv4；203.0.113.10 只是保留示例地址，不能用于连接。
<UUID>	VLESS 身份标识，由服务端生成；两端保持一致。
<REALITY_PRIVATE_KEY>	Reality 私钥；仅放 VPS，绝不放入客户端或订阅。
<REALITY_PUBLIC_KEY>	对应公钥；填入客户端 reality-opts.public-key。
<SHORT_ID>	本文生成 8 字节，即 16 个十六进制字符；保留字符串引号。
<HY2_PASSWORD>	Hysteria2 独立随机密码。
<TROJAN_PASSWORD>	Trojan 独立随机密码，与 HY2 密码分开生成。
<TLS_CERT_SHA256>	自签名叶子证书的 SHA-256 指纹；本例供 HY2 和 Trojan 共用。
www.cloudflare.com	Reality 握手目标示例；先从 VPS 验证可达，需要时两端一起修改。
proxy.invalid	自签名证书 SAN 与客户端 SNI；不需要 DNS 指向 VPS。
proxy.example.com	仅用于可选 CA 方案；替换为自己可控制的真实域名。

三个容易混淆的字段

server 是真正要连接的 VPS 地址。Reality 的 servername 是握手目标域名。HY2 与 Trojan 的 sni 是 TLS 证书对应名称；它们与 Reality 的握手目标并不是一回事。

fingerprint 是服务器证书指纹，用于固定证书身份。client-fingerprint: chrome 是 TLS ClientHello 的外观选项，不是证书指纹；不能互换。参见官方资料 [5][6]。

配置中的 `www.cloudflare.com` 和 `proxy.invalid` 是本例默认值；如果改名，需要同步更新相关服务端和客户端字段。客户端 YAML 使用空格缩进，不使用 Tab。

3 准备系统与防火墙

以下命令在 VPS 的 root shell 执行

登录 更新与时间同步

```
ssh <ADMIN_USER>@<VPS_IP>
sudo -i
apt update
apt upgrade -y
apt install -y curl ca-certificates openssl ufw nano tcpdump
timedatectl set-ntp true
timedatectl status
date -u
```

若升级提示需要重启，安排重启后再继续。检查时间同步状态；本例 Reality 限制两端时间差为 1 分钟，客户端也应开启自动校时。已有 chrony 等时间服务时，使用该服务确认同步即可。

先检查端口 再放行防火墙

```
ss -ltnup
ufw status verbose
```

在云控制台的安全组或防火墙放行实际 SSH 端口、TCP/443、UDP/443、TCP/8443。若有出站限制，还需允许 DNS、校时以及代理所需目标访问。先检查是否已有网站或其他代理占用这些端口。

```
ufw allow <SSH_PORT>/tcp
ufw allow 443/tcp
ufw allow 443/udp
ufw allow 8443/tcp
ufw enable
ufw status verbose
```

把 换成真实端口，例如 22。开启 UFW 前先放行 SSH，并保留当前 SSH 会话，用第二个窗口确认可重新登录。已有 nftables、firewalld、Docker 或面板规则时，按现有体系修改，不要叠加一套不明规则。

如何改用其他端口

例如 TCP/443 已被网站占用，可以把 Reality 改到另一个可用 TCP 端口；必须同时修改服务端 listen_port、客户端 port 和云端/本机防火墙。不要让 Reality 与 Trojan 同时直接监听同一 IP 的 TCP/443。本指南不引入额外分流前端。

仅 Trojan 客户端的 udp: true，并不表示需要开放 UDP/8443；本例 Trojan 外层仍通过 TCP/TLS 连接。

4 安装 sing-box 与生成认证信息

使用官方稳定软件包 记录实际版本

以下使用官方 APT 仓库安装 sing-box，不安装 beta 包。安装方式参见 [1]。

```
install -d -m 755 /etc/apt/keyrings
curl -fsSL https://sing-box.app/gpg.key \
  -o /etc/apt/keyrings/sagernet.asc
chmod 644 /etc/apt/keyrings/sagernet.asc
cat > /etc/apt/sources.list.d/sagernet.sources <<'EOF'
Types: deb
URIs: https://deb.sagernet.org/
Suites: *
Components: *
Enabled: yes
Signed-By: /etc/apt/keyrings/sagernet.asc
EOF
apt update
apt install -y sing-box
sing-box version
systemctl cat sing-box
id sing-box
```

本指南的权限命令按官方包的 sing-box 服务用户编写。若 id sing-box 失败，先检查软件包安装与系统用户创建；若 unit 使用其他用户，按实际用户和组调整权限。检查 ExecStart 实际读取的配置目录，并移走不再使用的旧 JSON 配置，防止重复载入。

生成身份标识 密钥与独立密码

```
sing-box generate uuid
sing-box generate reality-keypair
openssl rand -hex 8
openssl rand -hex 32
openssl rand -hex 32
```

按顺序记录 UUID、Reality 的 PrivateKey/PublicKey、Short ID、HY2 密码、Trojan 密码。两个密码命令各运行一次；不要重复使用同一输出。

```
umask 077
nano /root/sing-box-secrets.txt
chmod 600 /root/sing-box-secrets.txt
```

在秘密记录中按第 3 页参数名称保存各值。这里是记录文件，不是需要 source 的脚本。Reality 私钥、TLS 私钥、UUID 与密码均不要公开；客户端文件也包含可用凭据。

5 验证 Reality 并准备 TLS

默认方案不需要自有域名

先验证 Reality 握手目标

```
openssl s_client \  
-connect www.cloudflare.com:443 \  
-servername www.cloudflare.com \  
-tls1_3 -brief </dev/null
```

应能从 VPS 完成 TLS 1.3 握手。这是目标可用性的预检，不代替最终 Reality 连接测试。若目标在当前线路不可达，选择另一个可稳定访问、支持 TLS 1.3 的 HTTPS 目标，同时修改服务端 handshake.server 与客户端 servername。[2][5]

生成 HY2 与 Trojan 共用的自签名证书

本例让这两个入站共用一套 TLS 证书，以减少维护点；协议密码仍独立。证书 SAN 使用 proxy.invalid，客户端直接连接 VPS IP，无需为该名称配置 DNS。

```
install -d -o root -g sing-box -m 750 /etc/sing-box  
install -d -o root -g sing-box -m 750 /etc/sing-box/certs  
umask 077  
openssl req -x509 -newkey rsa:2048 -nodes -sha256 -days 365 \  
-keyout /etc/sing-box/certs/proxy.key \  
-out /etc/sing-box/certs/proxy.crt \  
-subj '/CN=proxy.invalid' \  
-addext 'subjectAltName=DNS:proxy.invalid' \  
-addext 'basicConstraints=critical,CA:FALSE'  
chown root:sing-box /etc/sing-box/certs/proxy.*  
chmod 640 /etc/sing-box/certs/proxy.key  
chmod 644 /etc/sing-box/certs/proxy.crt
```

已存在 proxy.key / proxy.crt 时不要直接重跑生成命令；会替换证书，使已有客户端的固定指纹失效。需要换证时先备份并安排所有客户端同步更新。证书私钥共用意味着换证时两个协议都要一起核对。

提取叶子证书指纹

```
openssl x509 -noout -fingerprint -sha256 \  
-in /etc/sing-box/certs/proxy.crt  
openssl x509 -noout -dates \  
-in /etc/sing-box/certs/proxy.crt
```

复制 SHA256 Fingerprint= 后面的完整值，作为 <TLS_CERT_SHA256>；保留冒号并放在 YAML 引号内。有效期 365 天只是本例的换证计划，不代表固定指纹会替你管理证书有效期。

6 证书信任与部署前检查

固定指纹与公有 CA 是两种不同信任路径

项目	默认自签名方案	可选公有 CA 方案
自有域名	不需要	本指南采用自己控制的域名
服务器证书	本地生成 proxy.crt 与 proxy.key	ACME 客户端申请并自动续期
客户端地址	server 填 VPS IP	server 仍可填 VPS IP
客户端名称	sni 填 proxy.invalid	sni 填证书覆盖的真实域名
验证方式	fingerprint 固定可信渠道获取的叶子证书	删去旧 fingerprint，使用 CA 证书链和名称验证
换证后	新叶子证书产生新指纹，两个节点都要更新	正常续期无需固定新叶子指纹
跳过验证	保持 skip-cert-verify: false	保持 skip-cert-verify: false

Mihomo 官方文档说明：固定叶子证书指纹时，只比较服务器发送的证书指纹，不再进行其他常规证书检查。因此，不能把这种配置描述为同时依赖公有 CA、主机名与有效期验证。指纹必须从自己的 VPS 或其他可信渠道获取。[6]

部署前确认这四件事

- 客户端确实使用支持 VLESS Reality、Hysteria2、Trojan 及证书指纹字段的 Mihomo 内核。名称含 Clash 的旧客户端不一定满足要求；以实际内核和版本为准。
- 安装目录中已有的配置已备份。不要在已有 sing-box 配置目录中保留会被继续加载的旧 .json 文件。
- Reality 公钥只进入客户端，私钥只进入服务端；Short ID 两端按相同字符串填写。
- 默认监听为 0.0.0.0，即 IPv4。不要只发布 AAAA 记录却没有 IPv6 监听和防火墙规则。

写入服务端配置前先备份

```
if [ -d /etc/sing-box ]; then
    tar -czf "/root/sing-box-$(date +%F-%H%M%S).tgz" \
        -C /etc sing-box
fi
nano /etc/sing-box/config.json
```

下一页是完整 JSON。替换占位符后保存；本例的 Trojan 没有额外 WebSocket、gRPC 或 fallback 设置。无需先部署网站。

7 完整 sing-box 服务端配置

文件 /etc/sing-box/config.json

```
{
  "log": { "level": "info" },
  "inbounds": [
    {
      "type": "vless", "tag": "reality-in",
      "listen": "0.0.0.0", "listen_port": 443,
      "users": [{
        "name": "primary", "uuid": "<UUID>",
        "flow": "xtls-rprx-vision"
      }],
      "tls": {
        "enabled": true,
        "reality": {
          "enabled": true,
          "handshake": {
            "server": "www.cloudflare.com", "server_port": 443
          },
          "private_key": "<REALITY_PRIVATE_KEY>",
          "short_id": ["<SHORT_ID>"],
          "max_time_difference": "1m"
        }
      }
    },
    {
      "type": "hysteria2", "tag": "hy2-in",
      "listen": "0.0.0.0", "listen_port": 443,
      "users": [{ "name": "primary", "password": "<HY2_PASSWORD>" }],
      "tls": {
        "enabled": true, "alpn": ["h3"],
        "certificate_path": "/etc/sing-box/certs/proxy.crt",
        "key_path": "/etc/sing-box/certs/proxy.key"
      }
    },
    {
      "type": "trojan", "tag": "trojan-in",
      "listen": "0.0.0.0", "listen_port": 8443,
      "users": [{
        "name": "primary", "password": "<TROJAN_PASSWORD>"
      }],
      "tls": {
        "enabled": true,
        "certificate_path": "/etc/sing-box/certs/proxy.crt",
        "key_path": "/etc/sing-box/certs/proxy.key"
      }
    }
  ],
  "outbounds": [{ "type": "direct", "tag": "direct" }],
  "route": { "final": "direct" }
}
```

三种 inbound 共用 direct 出站。字段依据 [2][3][4][5]；所有尖括号占位符均需替换。

8 校验 启动与监听检查

先确认服务端工作 再导入客户端

收紧权限并检查配置

```
chown root:sing-box /etc/sing-box/config.json
chmod 640 /etc/sing-box/config.json
runuser -u sing-box -- test -r /etc/sing-box/config.json
runuser -u sing-box -- test -r /etc/sing-box/certs/proxy.key
sing-box check -c /etc/sing-box/config.json
sing-box check -C /etc/sing-box
```

前两项读取测试退出状态应为 0；配置检查必须成功。-c 检查这一份文件，-C 检查服务实际使用的整个目录，可发现旧配置合并造成的重复标签等问题。若 systemd unit 使用其他参数，按实际参数检查。

启动服务并检查三个监听

```
systemctl enable sing-box
systemctl restart sing-box
systemctl status sing-box --no-pager -l
ss -lntp '( sport = :443 or sport = :8443 )'
ss -lnup '( sport = :443 )'
journalctl -u sing-box -n 80 --no-pager
```

检查项	预期结果
systemctl status	active (running)，没有持续重启
TCP 监听	sing-box 占用 0.0.0.0:443 与 0.0.0.0:8443
UDP 监听	sing-box 占用 0.0.0.0:443
日志	没有解析失败、权限不足、端口占用或证书读取错误

为客户端测试留一个日志窗口

```
journalctl -u sing-box -f --output cat
```

日志会帮助定位认证和握手错误，但日志级别可能不会显示每个入站包。不能仅凭“没有连接日志”认定网络被拦；必要时用第 14 页的 tcpdump 检查数据包是否到达。

单次 HTTPS 请求成功只说明所测链路可用，不能证明所有应用、UDP、DNS 和 IPv6 都已正确代理。下一页 YAML 是基础连通配置，不包含平台相关的完整 TUN 与 DNS 接管配置。

9 完整 Mihomo 客户端配置

在自己的设备保存为 config.yaml 并导入

```
mixed-port: 7890
allow-lan: false
mode: rule
log-level: info
ipv6: false

proxies:
- name: VPS-Reality
  type: vless
  server: "<VPS_IP>"
  port: 443
  uuid: "<UUID>"
  udp: true
  flow: xtls-rprx-vision
  packet-encoding: xudp
  network: tcp
  tls: true
  servername: www.cloudflare.com
  client-fingerprint: chrome
  reality-opts:
    public-key: "<REALITY_PUBLIC_KEY>"
    short-id: "<SHORT_ID>"

- name: VPS-HY2
  type: hysteria2
  server: "<VPS_IP>"
  port: 443
  password: "<HY2_PASSWORD>"
  sni: proxy.invalid
  fingerprint: "<TLS_CERT_SHA256>"
  skip-cert-verify: false
  alpn: [h3]

- name: VPS-Trojan
  type: trojan
  server: "<VPS_IP>"
  port: 8443
  password: "<TROJAN_PASSWORD>"
  udp: true
  network: tcp
  sni: proxy.invalid
  fingerprint: "<TLS_CERT_SHA256>"
  skip-cert-verify: false

proxy-groups:
- name: PROXY
  type: select
  proxies: [VPS-Reality, VPS-HY2, VPS-Trojan, AUTO]
- name: AUTO
  type: url-test
  proxies: [VPS-Reality, VPS-HY2, VPS-Trojan]
  url: https://www.gstatic.com/generate_204
  interval: 300
  tolerance: 80

rules:
- MATCH,PROXY
```

先手动选择节点，测试全部通过后再选 AUTO。手机与电脑的系统代理或 TUN 开关需另行启用。

10 逐一测试三种协议

手动验证后再启用自动选择

使用桌面命令行内核时，可先运行 `mihomo -v` 记录版本，再运行下方检查命令；图形客户端则使用导入检查与内核日志。确认本地 `mixed-port` 实际为 7890。

```
mihomo -t -f config.yaml
```

顺序	在 PROXY 中选择	重点核对
1	VPS-Reality	TCP/443 ; UUID、Reality 公钥、Short ID、servername、两端时间
2	VPS-HY2	UDP/443 ; HY2 密码、TLS 指纹、SNI 与 ALPN h3
3	VPS-Trojan	TCP/8443 ; Trojan 密码、TLS 指纹与 SNI
4	AUTO	三节点均经过独立测试后，再观察自动选择行为

每换一个节点 都重新发起一次请求

```
curl -4 --proxy http://127.0.0.1:7890 \  
  --connect-timeout 10 --max-time 30 https://api.ipify.org  
printf '\n'  
curl -I --proxy http://127.0.0.1:7890 \  
  --connect-timeout 10 --max-time 30 https://example.com
```

在自己的电脑执行。出口应是 VPS 实际出网 IP；某些云厂商通过 NAT 出网，出网地址可能不同于客户端连接的入站 IP，可在 VPS 上用 `curl -4 https://api.ipify.org` 对照。关闭旧连接或重启测试应用，避免策略切换后仍复用原连接。

如何解读测试结果

- 显式 `--proxy` 能排除系统代理未启用的干扰。如果 `curl` 通过而浏览器或某个应用失败，先查该应用是否使用系统代理，或是否需要 TUN。
- 通过 HY2 发出的 HTTPS 请求会验证客户端到 VPS 的 QUIC/UDP 链路，但不会证明应用原生 UDP 也已被 TUN 接管。游戏、语音等需要另外实测。
- AUTO 的 `url-test` 测量指定 URL 的连接表现，不能代表真实下载速度或全部网站可达。保留手动切换；测试 URL 不可用时换为自己能稳定访问的地址。[10]

默认 `MATCH,PROXY` 让进入 Mihomo 规则引擎的连接走 PROXY；它不是“整台设备所有流量已被代理”的保证。DNS、TUN 与 IPv6 需要按操作系统和客户端另行配置。

11 可选的自有域名证书

先完成默认方案 再切换证书信任方式

希望使用系统认可的证书链时，可把 HY2 和 Trojan 一起换为自有域名的公有 CA 证书。下面用 proxy.example.com 作占位示例，必须替换为自己的真实域名。Reality 继续使用原握手目标与密钥。

DNS 与验证方式

- 把真实域名的 A 记录指向 VPS。若有 AAAA 记录，必须确认对应 IPv6 服务同样可用，否则修正记录。此直连方案不经过普通 HTTP CDN 代理。
- 以下采用 Certbot standalone 的 HTTP-01 验证：公网 TCP/80 必须可达且本机空闲。云安全组和 UFW 均需放行。HTTP-01 不能简单改成 8443 验证。[11][12]
- 已有 Web 服务占用 80 时，使用相应 webroot/服务插件，或 DNS-01 插件。DNS-01 不要求 VPS 开放 80，但要保护 DNS API 凭据并配置自动续期。

申请证书

```
apt install -y certbot
ufw allow 80/tcp
ss -lntp '( sport = :80 )'
certbot certonly --standalone --preferred-challenges http \
  --cert-name proxy.example.com -d proxy.example.com
certbot certificates
```

完成交互式邮箱与条款设置。证书目录以 certbot certificates 的实际输出为准；本文通过 --cert-name 固定目录名，后续示例仍需替换。自动续期也依赖相同验证条件，不要签发后就关闭 TCP/80。

把证书放到 sing-box 可读取的位置

采用专用副本目录，避免把整个 /etc/letsencrypt 的权限放宽。下一页提供续期部署脚本，它会在续期成功后复制完整证书链与私钥，再检查并重启服务。首次也用同一个脚本完成复制。

来源文件	sing-box 使用的副本
fullchain.pem	/etc/sing-box/certs/ca-fullchain.pem
privkey.pem	/etc/sing-box/certs/ca.key

不要使用仅供 CDN 回源的 Origin CA 证书替代公网受信任证书。这里配置的是客户端直连 VPS 的 TLS；客户端必须能验证其信任链。

12 安装续期脚本并切换证书

本页的 proxy.example.com 均须替换

在 VPS 创建脚本。代码只处理指定证书目录；服务用户仍按 sing-box 设置。首次执行时原配置可继续使用自签名文件，复制成功后再改配置。

```
install -d -m 755 /etc/letsencrypt/renewal-hooks/deploy
nano /etc/letsencrypt/renewal-hooks/deploy/50-sing-box
```

```
#!/bin/sh
set -eu
expected=/etc/letsencrypt/live/proxy.example.com
[ "${RENEWED_LINEAGE:-}" = "$expected" ] || exit 0
dest=/etc/sing-box/certs
install -d -o root -g sing-box -m 750 "$dest"
install -o root -g sing-box -m 644 \
    "$RENEWED_LINEAGE/fullchain.pem" "$dest/ca-fullchain.pem"
install -o root -g sing-box -m 640 \
    "$RENEWED_LINEAGE/privkey.pem" "$dest/ca.key"
sing-box check -C /etc/sing-box
systemctl restart sing-box
systemctl is-active --quiet sing-box
```

```
chmod 750 /etc/letsencrypt/renewal-hooks/deploy/50-sing-box
RENEWED_LINEAGE=/etc/letsencrypt/live/proxy.example.com \
    /etc/letsencrypt/renewal-hooks/deploy/50-sing-box
```

修改服务端 HY2 与 Trojan 的 tls : certificate_path 改为 /etc/sing-box/certs/ca-fullchain.pem，key_path 改为 /etc/sing-box/certs/ca.key；保留 HY2 的 alpn: ["h3"]。

修改客户端这两个节点：sni 改为真实域名，删除旧 fingerprint 字段，保持 skip-cert-verify: false。server 可继续填 VPS IP。然后重新检查、重启并逐节点验证。

```
sing-box check -C /etc/sing-box
systemctl restart sing-box
certbot renew --dry-run
systemctl list-timers --all | grep -i certbot
```

确认已存在自动续期 timer 或 cron 任务。若系统提供 certbot.timer 但未启用，可运行 systemctl enable --now certbot.timer。dry-run 验证续期流程，默认不执行部署钩子；上面的首次手动调用用于验证复制与重启。[11]

脚本中的重启会短暂中断三种协议。续期、复制、校验或重启异常要查看日志并修复；不能仅根据证书签发成功判断服务已加载新证书。

13 按症状排查故障

从服务 监听 网络 认证顺序定位

现象	先检查什么
三个节点都失败	sing-box 是否运行；VPS 入站 IP 是否正确；云安全组、UFW、VPS DNS 与出网是否正常。
Reality 失败 其他成功	TCP/443、公钥与私钥配对、UUID、Short ID、servername、握手目标可达性和时钟。
仅 HY2 失败	UDP/443 是否可达；密码与证书指纹；h3 是否匹配；换网络测试 UDP。
仅 Trojan 失败	是否开放 TCP/8443；密码是否误用了 HY2 密码；证书路径、SNI、指纹或 CA 信任链。
address already in use	对应 TCP 或 UDP 端口已被其他进程占用；不要直接停掉未知业务。
permission denied	服务实际用户、配置/私钥读取权限、上级目录执行权限，以及 systemd 沙箱限制。
证书验证错误	是否换过证书但保留旧指纹；CA 方案是否遗漏完整链、名称不匹配或证书过期。
curl 通但应用不通	应用流量未进入 Mihomo；系统代理、TUN、DNS 或 IPv6 路径不同。
AUTO 全部显示失败	先手动逐节点访问；检查健康检查 URL、DNS 和测试站点本身。

最常用的诊断命令

```
systemctl status sing-box --no-pager -l
journalctl -u sing-box --since '10 min ago' --no-pager
ss -lntup
ufw status verbose
tcpdump -ni any 'tcp port 443 or tcp port 8443 or udp port 443'
curl -4 --connect-timeout 10 --max-time 30 https://api.ipify.org
getent ahostsv4 www.cloudflare.com
```

在测试窗口内，目标端口一个包也没有：先确认客户端真的发起连接、IP/端口/协议及抓包过滤器正确，再检查云防火墙、NAT 和中间网络。有包到达但握手失败，重点检查认证与 TLS。只有单向流量也可能是回程或丢包问题。

openssl s_client 可以检查 Trojan 的 TLS 证书，但它不是 Trojan 客户端，不能验证密码或代理转发成功。普通 TCP 工具也不能证明 HY2 的 UDP 链路可用。

14 连通之后的优化与配置分发

把性能 接管和分发问题分别验证

AUTO 与真实性能

先保留默认配置，在常用 Wi-Fi、移动网络等环境分别测量访问、下载和断线情况。url-test 只反映测试 URL 的连接表现，单次延迟低不等于吞吐更大；可随时手动指定节点。

BBR 只作用于相应的 TCP 路径

```
sysctl net.ipv4.tcp_available_congestion_control
sysctl net.ipv4.tcp_congestion_control
```

若内核已提供 bbr，可选择启用下面的 TCP 设置。若未提供，先查该发行版和内核支持，不要把启用 BBR 当作连通前提。

```
cat > /etc/sysctl.d/99-proxy-bbr.conf <<'EOF'
net.core.default_qdisc=fq
net.ipv4.tcp_congestion_control=bbr
EOF
sysctl --system
sysctl net.ipv4.tcp_congestion_control
```

这些设置面向内核 TCP，影响 Reality 和 Trojan 等 TCP 连接。HY2 在 QUIC 实现中使用自己的拥塞控制，不能直接套用这个开关。

HY2 带宽 混淆与端口跳跃

基础配置省略 up_mbps/down_mbps 与客户端 up/down；先确认连通，再按真实带宽与内核文档调参。不要照抄高带宽数值。obfs、端口跳跃会增加两端参数和 UDP 防火墙规则；确有需求时再启用并重新测试。[3][8]

TUN DNS 与 IPv6

系统代理通常只接管愿意使用它的应用。需要更多应用或原生 UDP 时，按客户端平台配置 TUN 与 DNS 接管，再检查实际出口。ipv6: false 是本例内核设置，不能保证操作系统不存在其他 IPv6 路径；开启双栈前要同步验证监听、防火墙、路由与 DNS。

先用本地 YAML 再考虑订阅

少量设备可以分别导入本地 YAML。订阅只是下载这份配置，仍需 HTTPS、访问控制与可撤销的凭据；不要放入公开仓库。订阅 URL 本身也应当作秘密。

本例 TCP/8443 已给 Trojan。若再搭订阅服务，请使用独立主机或另一个空闲 HTTPS 端口；不要沿用双协议版本把订阅也直接放在 8443 的安排。同机订阅不会提高 VPS 故障时的可用性。

15 维护 升级与回退

先备份并记录版本 再进行变更

升级前备份与记录

```
umask 077
tar -czf "/root/sing-box-backup-$(date +%F-%H%M%S).tgz" \
  -C /etc sing-box
dpkg-query -W -f='${Version}\n' sing-box \
  > /root/sing-box-version-before.txt
sing-box version
apt-cache policy sing-box
```

同时备份客户端 YAML 和实际 Mihomo 版本。若使用 CA 方案，再保护好 ACME 账号、续期配置和部署脚本。需要可靠回退时，应在升级前保存当前版本的软件包；只有版本号不保证仓库仍保留旧包。

阅读迁移说明后再升级

```
apt update
apt install --only-upgrade sing-box
sing-box version
sing-box check -C /etc/sing-box
systemctl restart sing-box
systemctl status sing-box --no-pager -l
```

不要在配置检查失败后继续重启。部分软件包升级过程本身可能重启服务，应安排维护窗口。升级后把三种协议逐一复测，再恢复 AUTO。跨大版本时先阅读官方迁移与兼容性说明；换成其他服务端内核也需要重新核对 Mihomo 兼容性。[6][13]

回退时区分配置与程序版本

- 只改错配置：恢复此前备份中的配置或证书，检查成功后再重启。恢复旧自签名证书时，客户端必须使用对应旧指纹。
- 升级导致不兼容：使用事先保存的软件包回退程序，同时恢复匹配配置。仅把 JSON 换回旧文件未必能解决问题。
- 秘密泄漏：更换相应 UUID、密码或密钥；若 TLS 私钥泄漏，替换证书和私钥并更新信任配置。不要只改节点显示名称。

日常维护

确认 SSH 密钥登录可用后再调整密码登录策略。定期检查日志、资源、证书与续期任务；仅开放必要端口，不向公网暴露 Mihomo 控制接口。

自签名证书更新会改变固定指纹；HY2 与 Trojan 两处都要更新。公有 CA 方案使用正常证书链验证时，避免重新加上固定旧叶子证书的 fingerprint。

16 验收清单与官方资料

满足清单后 再把配置分发到其他设备

验收项	完成条件
系统与网络	时间同步；真实入站 IP 已确认；SSH、TCP/443、UDP/443、TCP/8443 的规则正确。
服务与权限	目录检查成功；配置与私钥仅授权用户可读；sing-box 正常运行且三个监听齐全。
Reality	单独选择后 HTTPS 访问成功，出口为 VPS 实际出网地址。
Hysteria2	在目标网络单独测试成功，客户端到 VPS 的 UDP 链路可用。
Trojan	单独测试 TCP/8443 成功；使用独立密码且证书验证开启。
AUTO 与应用	三种协议独立验证完成；策略组可工作；实际应用接管方式已确认。
证书与维护	自签名指纹已同步，或 CA 续期与部署脚本已验证；配置、版本与回退材料已备份。

资料核对日期：2026-10-14。以下为官方配置与维护文档；升级后以当前安装版本对应文档和本机检查结果为准。链接可直接点击。

- [1] sing-box 安装与服务管理
- [2] sing-box VLESS 入站
- [3] sing-box Hysteria2 入站
- [4] sing-box Trojan 入站
- [5] sing-box TLS 与 Reality
- [6] Mihomo TLS 指纹与兼容性
- [7] Mihomo VLESS
- [8] Mihomo Hysteria2
- [9] Mihomo Trojan
- [10] Mihomo url-test
- [11] Certbot 用户指南与续期钩子
- [12] Let's Encrypt 验证方式
- [13] sing-box 配置迁移说明