# ThinkLink TLS 配置说明书

tkl-tls 2.2.0 中文版

2026 年 10 月 26 日

本手册供已有 ThinkLink Docker 部署的管理员使用。2.2 新增 MQTTS 8883 的监听、证书和 Docker 映射管理，统一支持换证、校验和回退。18083 仍为 HTTP，管理 HTTPS 使用 18084。

| 端口 | 协议 | 工具行为 |
| --- | --- | --- |
| 443 | HTTPS | ThinkLink 页面与 API |
| 18083 | HTTP | 保留原监听并补齐缺失映射 |
| 18084 | HTTPS | EMQX 管理页面及证书 |
| 8084 | WSS | 管理证书及 /mqtt 路径 |
| 8883 | MQTTS | 管理监听和证书并补齐映射 |
| 1883 / 8083 | MQTT / WS | 保留原配置及映射并只读检查 |

已有 18083、8084、8883 映射保留原宿主机端口和绑定地址，包括 loopback。缺失时默认补齐同号端口；如需限制访问，应在维护计划中核对新增映射和防火墙。

## 运行条件

单机 Linux x86_64、rootful Docker、Compose V2、EMQX 5.8.x；原 ThinkLink 与 EMQX HTTP 服务正常。工具为静态二进制，不安装网关，不下载或重建镜像，不安装主机依赖。

域名、证书签发、私有 CA 客户端信任和防火墙由部署方准备。证书 SAN 必须覆盖访问域名，PEM 链完整，私钥匹配且未加密，剩余有效期超过 24 小时。

# 1 安装与升级

将已审核的工具包与证书安全复制到服务器，解压后进入工具目录。先检查版本，再运行向导或以下预检命令。已有 2.0 或 2.1 部署可直接使用新工具 apply，继续使用原状态目录。

```
sudo ./tkl-tls --version
sudo bash tkl-tls.sh
```

向导询问域名、证书链和私钥路径。默认共用域名和证书，并管理 443、18084、8084、8883。核对实际计划后输入 YES；安装、换证和回退会短暂重建 ThinkLink 和 EMQX，请安排维护时间。

```
sudo ./tkl-tls check \
  --thinklink-domain iot.example.com \
  --cert /root/tls-input/fullchain.pem \
  --key /root/tls-input/privkey.pem
```

预检通过后将 check 改为 apply。私有 CA 加 --ca-file /root/tls-input/root-ca.pem。check 会生成受保护候选文件并启动短生命周期校验容器，但不重建现有服务或修改原 Compose。

## 域名与端口参数

EMQX 使用其他域名时传 --emqx-domain；独立证书同时传 --emqx-cert 和 --emqx-key。MQTTS、WSS 和管理 HTTPS 共用本次 EMQX 证书。

--mqtts-port 指定 8883 的宿主机端口；--wss-port 和 --emqx-http-port 对应 8084 和 18083。已有不同映射时拒绝覆盖。所有受管理端口必须不同且不得与其他服务冲突。

默认 Compose 为 /opt/thinklink/docker-compose.yml，服务名 tkl 和 emqx。自定义部署使用 --compose、--root、--thinklink-service、--emqx-service，更新时保持一致。

# 2 换证与恢复

```
sudo ./tkl-tls status
sudo ./tkl-tls apply \
  --cert /root/tls-input/new-fullchain.pem \
  --key /root/tls-input/new-privkey.pem
sudo ./tkl-tls rollback
```

换证继承已保存的域名、端口和管理模式。使用私有 CA 时继续提供 --ca-file；独立 EMQX 证书仍须成对提供。默认配置模式会一起更新 Dashboard HTTPS、WSS 和 MQTTS 证书。

应用失败自动尝试恢复上一部署并验证。强制结束或断电留下 pending.json 时，先运行 rollback，再继续 apply。Compose 被其他操作修改时拒绝覆盖，应先核对外部变更。

升级回退恢复升级前映射和配置：如果 2.1 原来没有 8883 映射，回退将移除本次新增映射。旧状态仍可用旧版工具 status 读取。回退不降级数据库，也不删除已持久化数据。

## 保护部署数据

工具固定当前镜像 ID，保留原数据卷，只重建两个目标服务。实际 EMQX 数据目录可能为 /opt/emqx/data2；未挂载时先停服复制并设置服务用户权限，再持久化挂载。

| 路径 | 用途 |
| --- | --- |
| /opt/tkl-tls | 私钥、状态和恢复记录 |
| /opt/tkl-tls/releases | 证书和版本配置 |
| /opt/tkl-tls/persistent | 迁移后的实际数据 |
| /opt/tkl-tls/original-compose.yml | 原部署备份 |

以上目录和备份可能包含私钥或原配置凭据，不得公开或随日志发送。不得删除状态目录，不要用 docker compose down -v 或 volume prune 排查 TLS 问题。

# 3 验证与管理模式

```
sudo ./tkl-tls ports
sudo ./tkl-tls status
```

ports 只读检查七个端口的容器监听、实际映射和本机 TCP。status 校验受管理入口的证书、域名、HTTP、WSS 101 和 MQTT CONNACK。本机通过后，应从另一台机器再次验证公网访问。

在 MQTTX 或设备中使用 mqtts://iot.example.com:8883，开启服务端证书验证，提供项目原认证信息，私有 CA 添加根证书。实际执行订阅、发布和重连；WSS 使用 wss://iot.example.com:8084/mqtt。

MQTTS 探测不保存账号，使用临时 clean session，不设置遗嘱、不发布消息。认证拒绝码 4 或 5 可证明 MQTT 协议正常，但不代表用户登录成功；实际认证收发必须单独验收。

## 配置模式与保留模式

默认 --mqtts-mode configure 和 --wss-mode configure 管理证书与监听。保留原认证、授权和客户端证书策略；遇到更高优先级环境变量拒绝覆盖。配置写入 emqx.conf，换证使用本工具，不在 Web 中重复上传。

首次纳管时，可用 --mqtts-mode preserve 保留已有 MQTTS 监听和证书；原 8883 必须监听且证书可信、域名匹配。WSS 同理使用 --wss-mode preserve，并要求 /mqtt 路径。已管理 MQTTS 从 configure 改为 preserve 前须回退。

要求客户端证书的双向 TLS 不受当前探测支持；工具不会关闭双向认证来通过检查。18083 保留 HTTP；浏览器管理入口为 https://iot.example.com:18084/。工具不签发证书、不自动续期、不改 DNS 或防火墙。

EMQX 5.8 TLS reference
https://docs.emqx.com/en/emqx/v5.8/network/emqx-mqtt-tls.html