使用 wpa_supplicant 步骤

1、 配置 wpa_supplicant.conf 文件：添加、修改、删除 network
2、 wpa_supplicant 命令创建进程：可使用 wpa_tools start/stop 创建删除
3、 wpa_cli 命令与 wpa_supplicant 进程交互：可使用 wpa_tools cli 创建
4、 漫游：在 wpa_cli 交互模式下
    输入：roam addr
    即可漫游到指定 AP

wpa_supplicant 设备

THINKPAD（X230）笔记本：
用户名：han
密码：123456
root 密码：123456

一 命令（shell）

文件位置：
    wpa.sh : /etc/wpa.sh
    wpa_tools（软链接）: /usr/bin
    wpa_supplicant log : /var/log/wpa.log
命令：
    wpa_tools start： 开始 wpa_supplicant 进程
    wpa_tools stop： 停止 wpa_supplicant 进程
    wpa_tools dhcp： 重新获取 IP 地址
    wpa_tools cli： 开启与 wpa_supplicant 交互的 wpa_cli
    default：帮助信息
    注意：wpa_tools 需要 root 权限运行

二 命令（wpa_supplicant）

2.1 wpa_supplicant 命令

用法
```
  wpa_supplicant [-BddfhKLqqtuvW] [-P<pid file>] [-g<global ctrl>] \
        [-G<group>] \
        -i<ifname> -c<config file> [-C<ctrl>] [-D<driver>] [-p<driver_param>] \
        [-b<br_ifname> [-MN -i<ifname> -c<conf> [-C<ctrl>] [-D<driver>] \
```

[-p<driver_param>] [-b<br_ifname>] [-m<P2P Device config file>] ...

选项

-b = 可选的桥接接口名称
-B = 后台运行
-c = 配置文件路径
-C = ctrl_interface 参数（仅在-c 不使用的时候使用）
-i = 接口名称
-d = 增加调试信息详细程度（-dd 显示更多）
-D = 驱动名称
-f = 将日志输出到默认日志位置（通常为/ tmp）
-g = 全局 ctrl_interface
-G = 全局 ctrl_interface group
-K = 包括密钥信息在调试中输出
-t = 调试信息添加时间戳
-h = 显示帮助文档
-L = 显示许可证
-p = 驱动程序参数
-P = PID 文件
-q = 减少调试信息详细程度（-qq 更少）
-u = 驱动 DBus control interface
-v = 显示版本
-W = 在启动之前等待 control interface monitor
-M = 开始描述匹配接口
-N = 开始描述新接口
-m = P2P Device 的配置文件

2.2 wpa_cli 命令

wpa_cli 是一个基于文本的前端程序，用于与 wpa_supplicant 进行交互。它用于查询当前状态，更改配置，触发事件并请求交互式用户输入。

wpa_cli 支持两种模式：交互式和命令行。

wpa_cli 命令：

status = 获取当前 WPA/EAPOL/EAP 状态
mib = 获取 MIB 变量
help = 显示帮助文档
interface [ifname] = 连接接口
level <debug level> = 改变 debug 等级
license = 显示 wpa_cli 证书
logoff = IEEE 802.1X EAPOL state machine logoff
logon = IEEE 802.1X EAPOL state machine logon
set = 设置变量
pmksa = 显示 PMKSA 缓存

reassociate = 强制重新连接
reconfigure =强制 wpa_supplicant 重新读取配置文件
preauthenticate <BSSID> = 强制预先认证
identity <network id> <identity> = 为 SSID 配置身份
password <network id> <password> =为 SSID 配置密码
otp <network id> <password> = 为 SSID 配置临时密钥
passphrase <network id> <passphrase> =为 SSID 配置私钥
bssid <network id> <BSSID> = 为 SSID 设置首选 BSSID
list_networks = 列举已经配置的网络
select_network <network id> = 选择网络（关闭其他）
enable_network <network id> = 启用网络
disable_network <network id> = 禁用网络
add_network = 添加网络
remove_network <network id> = 删除网络
set_network <network id> <variable> <value> =设置网络变量
get_network <network id> <variable> = 获取网络变量
save_config = 保存当前配置
disconnect = 断开连接，等待重新连接命令
scan = 扫描
scan_results = 扫描结果
get_capability <eap/pairwise/group/key_mgmt/proto/auth_alg> = 获取能力
terminate = 终止 wpa_supplicant
quit = 退出 wpa_cli
roam bssid = 漫游到指定 BSS

问题：在 wpa_supplicant 的官方文档 readme 中，包含的 wpa_cli 命令是不全的。
可以在 wpa_cli 交互中，help 获取全部命令

三 network 配置（wpa_supplicant.conf）

wpa_supplicant.conf.c

所有的配置信息全部在 wpa_supplicant 中有说明，下面介绍一些常用的参数。
修改配置文件：

vi /etc/wpa_supplicant.conf
修改保存即可

3.1 全局参数

1、ctrl_interface=/var/run/wpa_supplicant
//与 wpa_cli 通信的接口套接字所在路径
2、eapol_version=1
//IEEE802.1X/EAPOL 版本，默认版本为 1
3、ap_scan=1
//wpa_Supplicant 请求驱动扫描 AP，然后选择一个合适的 AP
1：wpa_supplicant 初始化扫描，选择合适的已经启用的 AP。如果没有符合的 AP，则会根据配置文件创建一个连接
0：不会试图连接 AP。如果使用有线以太网驱动时，必须为 0
2：与 0 类似，但是会使用安全的连接方法和 SSID 连接 AP
4、fast_reauth=1
//EAP 快速重新认证
5、load_dynamic_eap=/usr/lib/wpa_supplicant/eap_md5.so
//动态 EAP 方法，默认为静态，所以不是必须的
6、scan_cur_freq
//扫描当前频率
0：扫描当前有效的工作频率（默认）
1：Scan current operating frequency if another VIF on the same radio is already associated.
7、preassoc_mac_addr=0
//预先连接时的 MAC 地址政策
0：使用永久 MAC 地址
1：使用随机 MAC 地址
2：随机 MAC 地址，但是 OUI 相同
8、update_config=1
//允许 wpa_cli 中保存配置文件。无此项命令，wpa_cli 不能使用 save_config 命令
9、okc=1
//okc 模式是否打开，配合单个 network 中的 proactive_key_caching 使用
……

3.2 network block

用于连接 AP
disable：
0：默认。该网络可以被使用
1：该网络被禁用。可以通过 wpa_cli 启用
id_str：
网络标识字符串
***ssid**：网络名
scan_ssid：
0：默认。不发送探测帧进行 ssid 扫描

1：发送探测帧扫描

bssid：可选。如果配置，则只能连接配置的 BSSID

***priority**：优先级，默认为 0

mode：802.11 工作模式

0：基础模式 默认

1：IBSS

2：AP

注意：IBSS 只能在 key_mgmt NONE 和 WPA-PSK 情况下使用

frequency：IBSS 的通道频率

pbss：是否是个人基础服务集（802.11ad 独有）

0：不是

1：是

2：不关心

proto：能够接受的协议列表（默认是 WPA RSN）

WPA = WPA/IEEE 802.11i/D3.0

RSN = WPA2/IEEE 802.11i

***key_mgmt**：可接受的认证密钥管理协议**列表**

WPA-PSK = WPA pre-shared key (this requires 'psk' field)

WPA-EAP = WPA using EAP authentication

IEEE8021X = IEEE 802.1X using EAP authentication and (optionally) dynamically generated WEP keys

NONE = WPA is not used; plaintext or static WEP could be used

WPA-NONE = WPA-None for IBSS (deprecated; use proto=RSN key_mgmt=WPA-PSK instead)

FT-PSK = Fast BSS Transition (IEEE 802.11r) with pre-shared key

FT-EAP = Fast BSS Transition (IEEE 802.11r) with EAP authentication

WPA-PSK-SHA256 = Like WPA-PSK but using stronger SHA256-based algorithms

WPA-EAP-SHA256 = Like WPA-EAP but using stronger SHA256-based algorithms

SAE = Simultaneous authentication of equals; pre-shared key/password -based authentication with stronger security than WPA-PSK especially when using not that strong password

FT-SAE = SAE with FT

WPA-EAP-SUITE-B = Suite B 128-bit level

WPA-EAP-SUITE-B-192 = Suite B 192-bit level

OSEN = Hotspot 2.0 Rel 2 online signup connection

默认为 WPA-PSK WPA-EAP

ieee80211w：管理帧保护是否开启

0：关闭（默认）

1：可选

2：必需的

auth_alg：802.11 认证方式

OPEN：open 认证

SHARED：共享密钥
LEAP

pairwise：可接受的 WPA 的成对算法列表。（默认 CCMP 和 TKIP）
CCMP
TKIP
NONE

group: 可接受的 WPA 的组成对算法列表。（默认 CCMP、TKIP、WEP104、WEP40）
CCMP
TKIP
WEP104
WEP40

***psk**：WPA 预共用密钥 256bit

mem_only_psk：是否保存 PSK 到内存
0：允许保存到配置文件
1：不允许

***proactive_key_caching=1**：（okc）使能，与 okc 配合使用

***eap**：eap 方法列表
***identity**：eap 身份
anonymous_identity：匿名 eap 身份
***password**：eap 密码
ca_cert：ca 证书路径（网络获取）
ca_path：ca 证书目录路径
client_cert：客户证书路径
private_key：私有密钥路径
private_key_passwd：

……

3.3 示例

PSK 认证：

```
network={
    ssid="example"
    proto=WPA
    key_mgmt=WPA-PSK    FT-PSK（802.11r）
    pairwise=CCMP TKIP
    group=CCMP TKIP WEP104 WEP40
    psk=”123456”
    priority=2
}
```

802.1X 认证：

```
network={
```

```
    ssid="example"
    eap=PEAP
    key_mgmt=WPA-EAP FT-EAP（802.11r）
    identity="test"
    password="123456"
    priority=3
#   proactive_key_caching=1
}
```