|  |  |  |
| --- | --- | --- |
| 华为技术有限公司 | 产品版本 | 密级 |
|  | 外部公开 |
| 产品名称 | 页数：共21页 |

【华为SSL VPN集成指导文档-Android】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **拟制:** |  |  | **日期：** | **2020-3-10** |
| **审核:** |  |  | **日期：** | **年-月-日** |
| **审核:** |  |  | **日期：** | **年-月-日** |
| **批准:** |  |  | **日期：** | **年-月-日** |



**华为技术有限公司**

**版权所有 侵权必究**

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **修订版本** | **描述** | **作者** |
| 2019-06-18 | 1.0 | 初稿 |  |
| 2020-03-20 | 2.0 | 第一版 |  |
|  |  |  |  |

目录

[1 SDK介绍 6](#_Toc43906836)

[1.1 兼容性 6](#_Toc43906837)

[1.2 支持功能 6](#_Toc43906838)

[1.3 网络扩展场景描述 6](#_Toc43906839)

[2 SDK使用说明 7](#_Toc43906840)

[2.1 Android SDK包组成 7](#_Toc43906841)

[2.2 集成说明 8](#_Toc43906842)

[2.2.1 导入jar包和so库 8](#_Toc43906843)

[2.2.2 代码修改 9](#_Toc43906844)

[2.2.3代码修改初始化日志 10](#_Toc43906845)

[2.2.4初始化vpn回调 11](#_Toc43906846)

[注：所有失败之后都需要调用vpn退出函数 11](#_Toc43906847)

[2.2.5建立sslvpn连接 11](#_Toc43906848)

[3 安卓平台接口介绍 12](#_Toc43906849)

[3.1 初始化日志 12](#_Toc43906850)

[3.2 初始化callback回调 13](#_Toc43906851)

[3.3 14](#_Toc43906852)

[3.4 初始化vpn参数 14](#_Toc43906853)

[3.5 开始登录 15](#_Toc43906854)

[3.6 退出登录接口 15](#_Toc43906855)

[3.6初始化修改密码callback回调 16](#_Toc43906856)

[3.7 设置修改密码回调结果 17](#_Toc43906857)

[**设置修改密码回调结果** 17](#_Toc43906858)

[3.8 修改密码接口 17](#_Toc43906859)

# SDK介绍

## 兼容性

android平台仅支持android5.0及以上版本。

## 支持功能

支持SSL VPN连接

支持网络扩展功能

支持用户名密码认证方式

## 网络扩展场景描述

移动端SDK目前仅支持网络扩展，此功能需要在客户端生成一个虚拟网卡，通过虚拟网卡与企业 内网进行通信。网络扩展功能将终端用户连接到企业网上，对终端用户而言，就像接入到企业内网的局域网一样，可以访问内网的任何资源。

网络扩展功能支持所有基于IP层之上的应用。因此被称为基于IP层的隧道技术，称作L3VPN功能。相对于其他业务功能而言，用户启用网络扩展功能后，如同接入到企业内网一样，网络扩展功能主要是为了提供用户远程接入能力，提供IP层的接入功能，实现对内网的全网访问。使用网络扩展功能之后，用户的PC就可以如同在企业内网一样，可以快速、安全的访问企业内网的所有资源。

网络扩展的典型应用场景如下：

网络扩展是基于SSL协议进行的功能扩展。

网络扩展是基于IP的内网业务的全面访问。

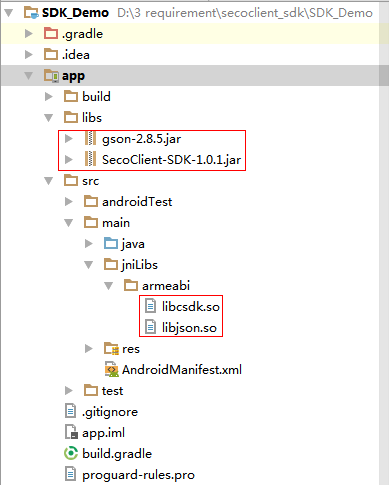
路由模式决定了客户端发送报文的路由。网络扩展功能支持三种路由模式：

分离模式（Split Tunnel），全路由模式（Full Tunnel），手动模式（Manual Tunnel）。详细介绍如下：  
• 分离模式下，客户端发送给内网的数据，经系统路由表识别以后，交由虚拟网卡转发，其源IP赋值为虚拟IP。而访问本地子网的数据则交由真实网卡转发，源IP赋值为真实的IP。由此，网络扩展只转发前往内网的数据。同时，分离模式也把不是访问本地子网资源的其他数据经过虚拟网卡转发。  
• 全路由模式下，无论是访问什么资源，数据一概被虚拟网卡截获，转发给虚拟网关处理。  
• 手动模式下，在FW端，管理员必须手动配置内网网段静态路由，然后在客户端识别前往该网段的数据，交由虚拟网卡转发。

# SDK使用说明

## Android SDK包组成

SecoClient-SDK-1.0.0.jar、及libcsdk.so、libjson.so两部分组成。



**SecoClient-SDK-1.0.0.jar包：封装的用户接口包**

**.so后缀文件：为编译的SDK库文件，需要加载到工程当中**

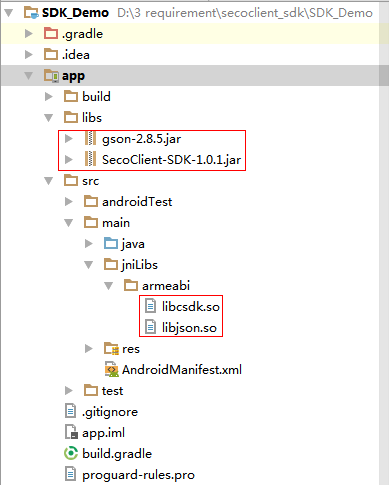
**注意：gson-2.8.5.jar 也需要和 SecoClient-SDK-1.0.0.jar一起放到lib下**

## 集成说明

### 2.2.1 导入jar包和so库

1.将jar导入工程当中，在android工程中找到对应的libs目录，拷贝SecoClient-SDK-1.0.0.jar、gson-2.8.5.jar到libs目录下。注意：app代码混淆时，不要混淆这两个jar包，需要排除com.huawei.secoclient包名下的所有文件。

2. so导入到工程中，Android studio放入在jniLibs目录下,如下图：



2.2.2 代码修改

(1)修改AndroidManifest.xml文件

添加访问网络及wifi状态、sd卡读写等权限和服务。

添加vpn服务

<action android:name="android.net.VpnService" />

(2)添加类继承于Application类

在此类中创建一个ServiceConnection的实例,ServiceConnection包含一个回调方法，系统调用这个方法来传递要返回的IBinder．

在Application类中加载so库

System.loadLibrary("json");  
System.loadLibrary("csdk");

在 Application类的onCreat方法中添加绑定vpn服务的方法。

### 2.2.3代码修改初始化日志

初始化SslVpnUtils类，调用其中的initlog函数进行初始化日志

SslVpnUtils mUtils =new SslVpnUtils()

void initLog(Activity activity, String path, int loglevel)

activity:当前activity

path:日志保存路径

loglevel：日志等级，日志等级分为两类，只可传入3或者4，当日志等级为3时，打印info、warning、error日志，当日志等级为4时，将打印info、warning、error及debug日志。

mUtils.initLog(MainActivity.this, path, 4)

2.2.4初始化vpn回调

1. 初始化回调函数mUtils.initcallback(Context);
2. 实现回调监听函数

mUtils.setOnCallBackListener(newSslVpnUtils.OnCallBackListen()

Const.CONNECT\_FAIL：无法建立VPN连接，VPN服务器可能无法到达

Const.UDP\_TUNNEL\_FAIL：网络扩展启动失败

Const.VPN\_FORCE\_EXIT：被强制下线

Const.VPN\_RECONNECT\_FAIL：断线重连失败

Const.VPN\_CONTINUE\_FAIL：DHCP续租失败

Const.LOGIN\_FAIL：认证失败

Const.LOGIN\_FIRST\_LOGIN：首次登录需要修改密码

Const.CNEM\_FAIL：网络扩展启动失败

Const.CNEM\_SUCCCESS：网络扩展启动成功

注：所有失败之后都需要调用vpn退出函数mUtils.loginOut()

2.2.5建立sslvpn连接

1、准备vpn连接

Intent intent = VpnService.prepare(MainActivity.this);

startActivityForResult启动服务

2、在onActivityResult方法中接收startactivity回调

如允许建立vpn连接，则开始初始化vpnSettingModel对象

vpnSettingModel：VpnSettingModel对象

需要在vpnSettingModel对象中设置：

虚拟网关地址：vpnSettingModel.setConnectIp(String)

端口号: vpnSettingModel.setConnectPort(String)

是否是国密: vpnSettingModel. setGmAlgorithm(int)

用户名: vpnSettingModel.setConn setUserName (String)

密码: vpnSettingModel. setPassWord(String)

隧道模式信息vpnSettingModel. setTunnelMold(int)

隧道传输模式，取值为：

VPN\_TUNNEL\_SAFE = 0;//可靠模式  
 VPN\_TUNNEL\_FAST = 1;//快速模式  
 VPN\_TUNNEL\_INTRODUCTIONS = 2;//自适应模式

选择自适应模式，如果快速模式不通，则切换可靠模式

1. 设置vpn连接配置

SslVpnUtils. setVpnData(Context，VpnSettingModel，SslVpnServiceProvider)

1. 开始登陆(登陆前要初始化vpn回调函数)

mUtils.vpnLogin();

5、退出登录：mUtils.loginOut()

# 安卓平台接口介绍

## 初始化日志

|  |  |  |
| --- | --- | --- |
| **名称** | **boolean initLog(Activity activity, String path, int loglevel)** | |
| **功能** | **初始化日志函数** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **activity** | **当前activity的实例** |
| **path** | **日志初始化路径** |
| **loglevel** | **日志初始化等级** |
| **返回值** | **boolean 成功调用返回true，失败调用返回false** | |
| **说明** | **sdk日志初始化，记录sdk运行中的日志**  **loglevel取值范围为[3,4],当取值为3时，打印info、warning、error日志，当日志等级为4时，将打印info、warning、error及debug日志** | |
| **备注** | **无** | |

## 初始化callback回调

|  |  |  |
| --- | --- | --- |
| **名称** | **void initcallback(Context context)** | |
| **功能** | **初始化callback回调** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **context** | **当前activity上下文** |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **无** | |

## 初始化回调结果函数

|  |  |  |
| --- | --- | --- |
| **名称** | **void setOnCallBackListener (new SslVpnUtils.OnCallBackListener())** | |
| **功能** | **初始化回调结果函数** | |
| **前置条件** | **初始化callback回调** | |
| **参数** | **参数名** | **描述** |
| **OnCallBackListener** | **回调接口** |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **无** | |

## 初始化vpn参数

|  |  |  |
| --- | --- | --- |
| **名称** | **void setVpnData(Context context, VpnSettingModel mode, SslVpnServiceProvider provider)** | |
| **功能** | **设置vpn参数** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **context** | **当前activity上下文** |
| **currentVpn** | **VpnSettingModel对象** |
| **provider** | **在application中返回的SslVpnServiceProvider实例** |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **vpnSettingModel对象中必须添加虚拟网关ip地址、端口号、是否支持国密、用户名、密码、隧道模式字段。** | |

## 开始登录

|  |  |  |
| --- | --- | --- |
| **名称** | **boolean vpnLogin()** | |
| **功能** | **Sslvpn开始登录** | |
| **前置条件** | **已进行初始化回调函数及设置vpn参数函数** | |
| **参数** | **参数名** | **描述** |
| **无** |  |
| **返回值** | **boolean 成功调用返回true，失败调用返回false** | |
| **说明** | **最终结果由回调接口返回setOnCallBackListener** | |
| **备注** |  | |

## 退出登录接口

|  |  |  |
| --- | --- | --- |
| **名称** | **void loginOut()** | |
| **功能** | **Sslvpn退出登录接口** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **无** |  |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **无** | |

## 3.6初始化修改密码callback回调

|  |  |  |
| --- | --- | --- |
| **名称** | **void initPwdCallback();** | |
| **功能** | **初始化initPwdCallback回调** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **无** |  |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **无** | |

## 设置修改密码回调结果

|  |  |  |
| --- | --- | --- |
| **名称** | **mUtils.setOnPwdCallBackListener(new SslVpnUtils.OnPwdCallBackListener()** | |
| **功能** | **设置修改密码回调结果** | |
| **前置条件** | **无** | |
| **参数** | **参数名** | **描述** |
| **OnPwdCallBackListener** | **修改密码回调接口** |
| **返回值** | **无** | |
| **说明** | **无** | |
| **备注** | **无** | |

## 修改密码接口

|  |  |  |
| --- | --- | --- |
| **名称** | **boolean ModifyPassword(final SslVpnServiceProvider provider, String OriginalPass, String newPass, String confirmPass)** | |
| **功能** | **修改密码接口** | |
| **前置条件** | **sslvpn连接成功后调用，且已初始化修改密码回调initPwdCallback();** | |
| **参数** | **参数名** | **描述** |
| **provider** | **在application中返回的SslVpnServiceProvider实例** |
| **OriginalPass** | **原始密码** |
| **newPass** | **新密码** |
| **confirmPass** | **确认新密码** |
| **返回值** | **boolean 成功调用返回true，失败调用返回false** | |
| **说明** | **1、sslvpn已连接成功后调用，且已初始化修改密码回调initPwdCallback()**  **2、最终结果由回调接口返回setOnPwdCallBackListener** | |
| **备注** | **回调结果：**  **Const.MODIIFY\_PASS\_SUCCESS：修改密码成功**  **Const.MODIIFY\_OLD\_PASS\_ERROR：输入的旧密码错误**  **Const.MODIIFY\_PASS\_IS\_LOW\_ERROR：新密码长度为1~16个字符，且不能仅包含‘\*’**  **Const.MODIIFY\_PASS\_IS\_MIDDLE\_ERROR：新密码不能与用户名相同，长度为6-16，至少包含大小写字母、数字、特殊字符中的二种**  **Const.MODIIFY\_PASS\_IS\_HIGH\_ERROR：新密码不能与用户名相同，长度为6-16，至少包含大小写字母、数字、特殊字符中的三种** | |