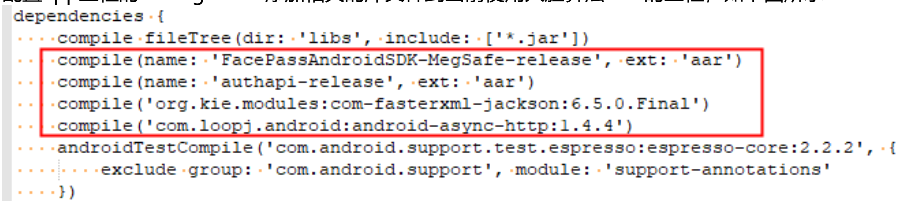
FacePass Android版MegSafe软授权操作指南 V3.4.2

1. App在启动进入主界面前，必须先进行授权激活操作；没有授权的设备，在加载人脸算法SDK时，会导致应用直接闪退。
2. 相关文件列表
    a. SDK/libs/authapi-release.aar MegSafe软授权接口
    b. Doc/get_authed_file.py 联网获取授权文件的脚本 （新的授权aar包里，有接口实现这个脚本的功能，这个脚本可以不用）
    c. Demo/TestFacePass_MegSafeMegSafe 软授权使用demo，demo中的lib目录包含了人脸算法sdk包（FacePassAndroidSDK-MegSafe-release.aar）
    d. Demo/MegSafe_AuthServer 如果用户授权方式使用代理服务器，则需要用到这里面的程序
3. authapi-release.aar接口使用说明：

    a. 配置app工程的build.gradle 添加相关的库文件到当前使用人脸算法SDK的工程，如下图所示：



```
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile(name: 'FacePassAndroidSDK-MegSafe-release', ext: 'aar')
    compile(name: 'authapi-release', ext: 'aar')
    compile('org.kie.modules:com-fasterxml-jackson:6.5.0.Final')
    compile('com.loopj.android:android-async-http:1.4.4')
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
```

    b. 在进入用户程序主界面前，先进入授权页面，这步由用户自行实现。也可参考demo中的流程：SplashActivity->AuthActivity→MainActivity.
    c. 获取设备的指纹文件，接口如下：
        i. AuthApi obj = new AuthApi();
        ii. ReturnInfo text = new ReturnInfo();
        iii. obj.GenC2vFile(text);
        - 上面三步调用，将在/sdcard/根目录生成request.c2v文件，该文件为设备的指纹文件。参考AuthActivity模块中的generate_fingerprint函数。

generateFingerprint参考代码

```
private  void generateFingerprint() {
        AuthApi obj = new AuthApi();
        ReturnInfo text = new ReturnInfo();
        obj.GenC2vFile(text);
}
```

    d. 联网激活设备

方式一：端上联网获取授权文件并激活设备

1）端上主要接口

端上联网授权接口

```
a)  String GetAuthFileByKey (String KeyFileName)
/sdcard/Download/v2c
          httpGetNetResult

ok, http

 MyHttpHandler.GetErrorInfo()


b)  String GetNetResult()
ActiveAuth
”ok” v2c
            “false” v2c
            “org”

MyHttpHandler.GetErrorInfo()


c) /sdcard/Download
String path = "update.v2c";
AuthApi obj = new AuthApi();
final ReturnInfo retunInfo = new ReturnInfo();
obj.AuthDevice(path, retunInfo);


ReturnInfo text"Apply update: OK""Apply update: Update already added"
```

2）端上参考代码

端上联网获取授权

```
private boolean getActivateFile() {
        NetApi api = new NetApi();
        return api.GetAuthFileByKey("CBG_Android_Face_Reco---89-Trial-one-stage.cert");
}


private class authThread extends Thread {

        @Override
        public void run() {
            try {
                    boolean res = getActivateFile();  //
                    if (!res) {
                        Log.d(TAG, "after getActivateFile result: " + res);
                        runOnUiThread(new Runnable() {
                            @Override
                            public void run() {
                                showAuthResult("getActivateFile fail ... ...");
                            }
                        });

                        return;
                    }

                    //getActivateFileFromHost();   //

                    Thread.sleep(1000);
                    while (true) {
                        String netRes = MyHttpHandler.GetNetResult();
                        Log.d(TAG, "GetNetResult: " + netRes);
                        if (netRes.equals("org")) {
                            Thread.sleep(1000);
                            String errInfo = MyHttpHandler.GetErrorInfo();
                            Log.d(TAG, errInfo);
                            continue;
                        } else if (netRes.equals("ok")) {
                            break;
                        } else {
                            String errInfo = MyHttpHandler.GetErrorInfo();
                            Log.d(TAG, errInfo);
                            return;
                        }
                    }

                    Log.d(TAG, "before ActivateDevice");
                    String path = "update.v2c";
                    AuthApi obj = new AuthApi();
                    final ReturnInfo retunInfo = new ReturnInfo();
                    obj.AuthDevice(path, retunInfo);

                    runOnUiThread(new Runnable() {
                        @Override
                        public void run() {
                            showAuthResult(retunInfo.retInfo);
                        }
                    });

                    Log.d(TAG, retunInfo.retInfo);
            } catch (InterruptedException e) {
                    e.printStackTrace();
            }
        }
    }
```

方式二：通过代理服务器联网获取授权文件并激活设备

1）代理服务器配置及运行

当前仅支持window 平台， 具体请参考代理服务器的使用说明（Demo/MegSafe_AuthServer）。

2）端上主要接口

端上联网授权接口

```
a)  boolean GetAuthFileByProxy (String AuthSvrIp,
    String AuthSvrPort, String KeyFile,
    String ActiveCode)
v2cGetNetResult
AuthSvrIp: ip
      AuthSvrPort:
      KeyFile:
      ActiveCode:


b)  String GetNetResult()
ActiveAuth
”ok” v2c
            “false” v2c
            “org”

MyHttpHandler.GetErrorInfo()


c) /sdcard/Download
String path = "update.v2c";
AuthApi obj = new AuthApi();
final ReturnInfo retunInfo = new ReturnInfo();
obj.AuthDevice(path, retunInfo);


ReturnInfo text"Apply update: OK""Apply update: Update already added"
```

3）端上参考代码

端上联网获取授权

```
private void getActivateFileFromHost() {
        NetApi api = new NetApi();
        api.GetAuthFileByProxy("10.156.xxx.xxx", "6666", "CBG_Android_Face_Reco---89-Trial-one-stage.cert", "");
    }
private class authThread extends Thread {

        @Override
        public void run() {
            try {
//                    boolean res = getActivateFile();  //
//                    if (!res) {
//                        Log.d(TAG, "after getActivateFile result: " + res);
//                        runOnUiThread(new Runnable() {
//                            @Override
//                            public void run() {
//                                showAuthResult("getActivateFile fail ... ...");
//                            }
//                        });
//
//                        return;
//                    }

                    getActivateFileFromHost();   //

                    Thread.sleep(1000);
                    while (true) {
                        String netRes = MyHttpHandler.GetNetResult();
                        Log.d(TAG, "GetNetResult: " + netRes);
                        if (netRes.equals("org")) {
                            Thread.sleep(1000);
                            String errInfo = MyHttpHandler.GetErrorInfo();
                            Log.d(TAG, errInfo);
                            continue;
                        } else if (netRes.equals("ok")) {
                            break;
                        } else {
                            String errInfo = MyHttpHandler.GetErrorInfo();
                            Log.d(TAG, errInfo);
                            return;
                        }
                    }

                    Log.d(TAG, "before ActivateDevice");
                    String path = "update.v2c";
                    AuthApi obj = new AuthApi();
                    final ReturnInfo retunInfo = new ReturnInfo();
                    obj.AuthDevice(path, retunInfo);

                    runOnUiThread(new Runnable() {
                        @Override
                        public void run() {
                            showAuthResult(retunInfo.retInfo);
                        }
                    });

                    Log.d(TAG, retunInfo.retInfo);
            } catch (InterruptedException e) {
                    e.printStackTrace();
            }
        }
    }
```

4. 激活设备后，才可以进入主界面。

需要由用户自己去记录设备的激活状态，只有激活成功才能进入主界面，才能使用人脸sdk算法