移动安全

快速入门

快速入门

登录 移动安全管理控制台。

单击 开通移动安全，即可开通。

以漏洞扫描升级专业版为例，步骤如下：

第一步 购买实例

（若已购买该服务专业版实例，跳过这一步）

单击查看教程。

第二步 应用所有权认证

（若该应用已认证过，则跳过这一步）

为了保障应用开发者的隐私，在实例配置前需进行应用所有权认证。

登录 移动安全管理控制台。

选择要认证应用，单击该应用的 **升级** 下拉框。

选择欲升级的服务，进入所有权认证页面。

参照验证原理，下载空白应用，使用页面所示命令进行签名后上传。

注意： 认证已发布的应用，待认证应用、空白应用和官方应用的签名必须保持一致；如果你的应用还在开发中，则需保持待认证应用和空白应用的签名一致。

认证成功，总览页面会显示“认证成功”，同时您将会收到收到短信，邮件和站内信通知。

若认证失败，则会在页面中提醒错误原因与解决办法，请解决后重新上传。

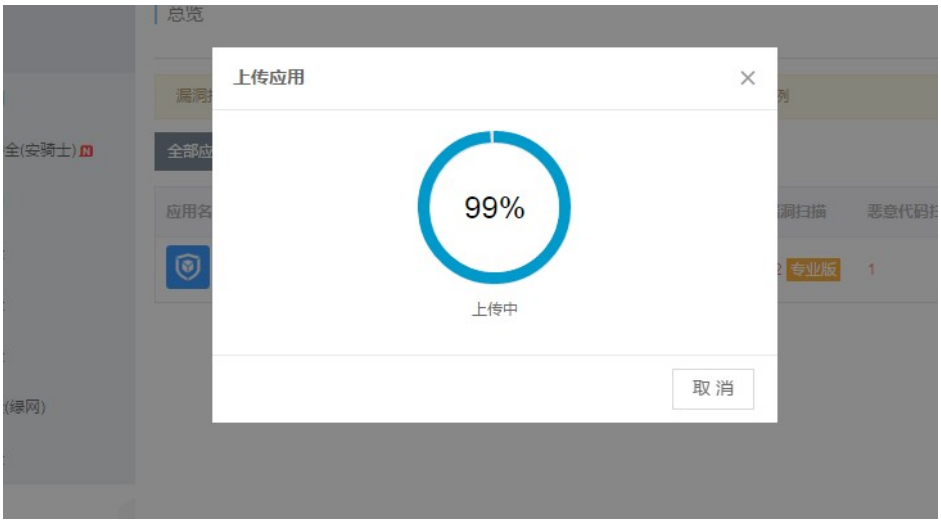
若出现以下页面，说明您的应用进入人工审核流程，请填写官网下载地址或大型应用市场下载页面地址，配合工作人员进行审核，预计需要一个工作日，请耐心等待。

第三步 实例配置

应用所有权通过后，则会自动跳转到实例配置页面。若是人工审核通过，则需单击总览页的 **升级**。在实例列表中选择您要使用的实例，单击 **确定**，将其配置到应用上。

单击左侧栏目【总览】，在页面右上角上传APK包。

等待上传应用，上传成功后勾选漏洞扫描功能（可以全部勾选），等待完成APK扫描。



单击漏洞扫描的数字，进入漏洞扫描页面查看漏洞具体详情。

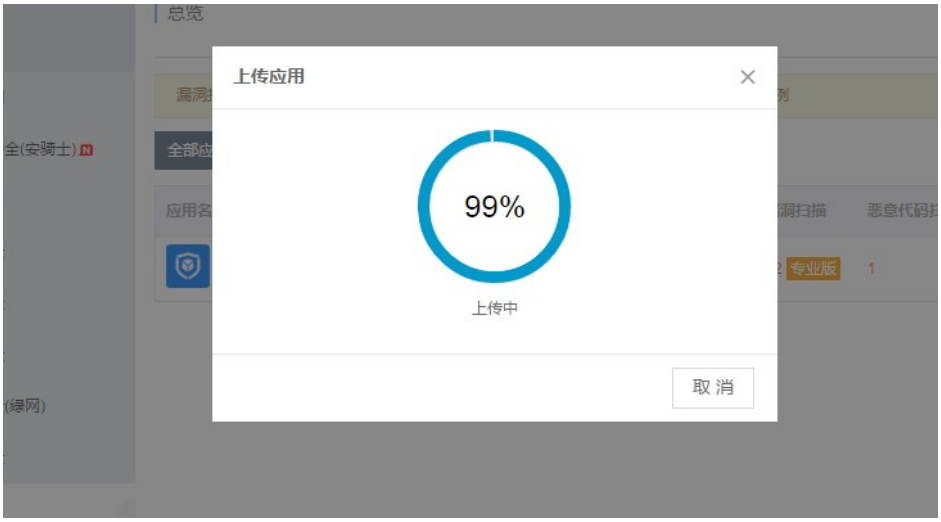
页面上方显示为漏洞总数和漏洞分类，下方为漏洞名称、详情、修复建议等。

展开漏洞详情，可查看具体漏洞位置。

注意： 如果您需要查看具体的漏洞位置，请 升级到专业版。

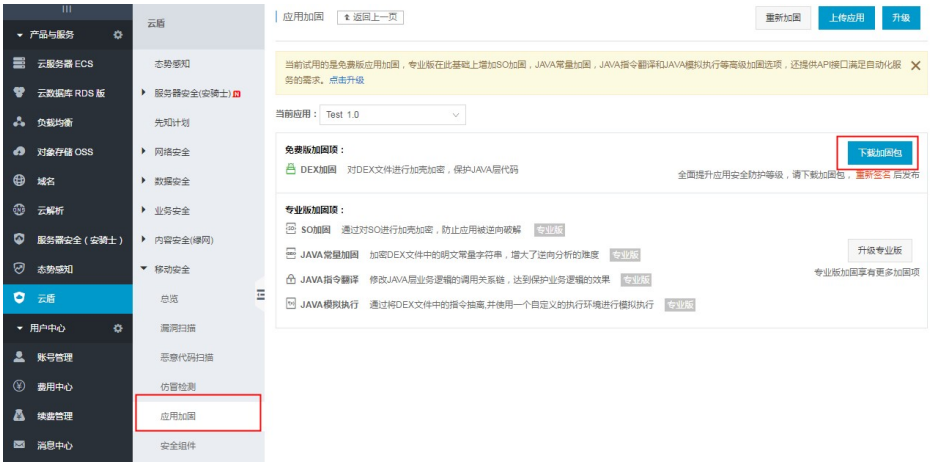
单击左侧栏目 **总览**，在页面右上角上传APK包。

等待上传应用，上传成功后勾选应用加固功能（可以全部勾选）。



等待应用加固完成，下载加固包，并给 应用签名。

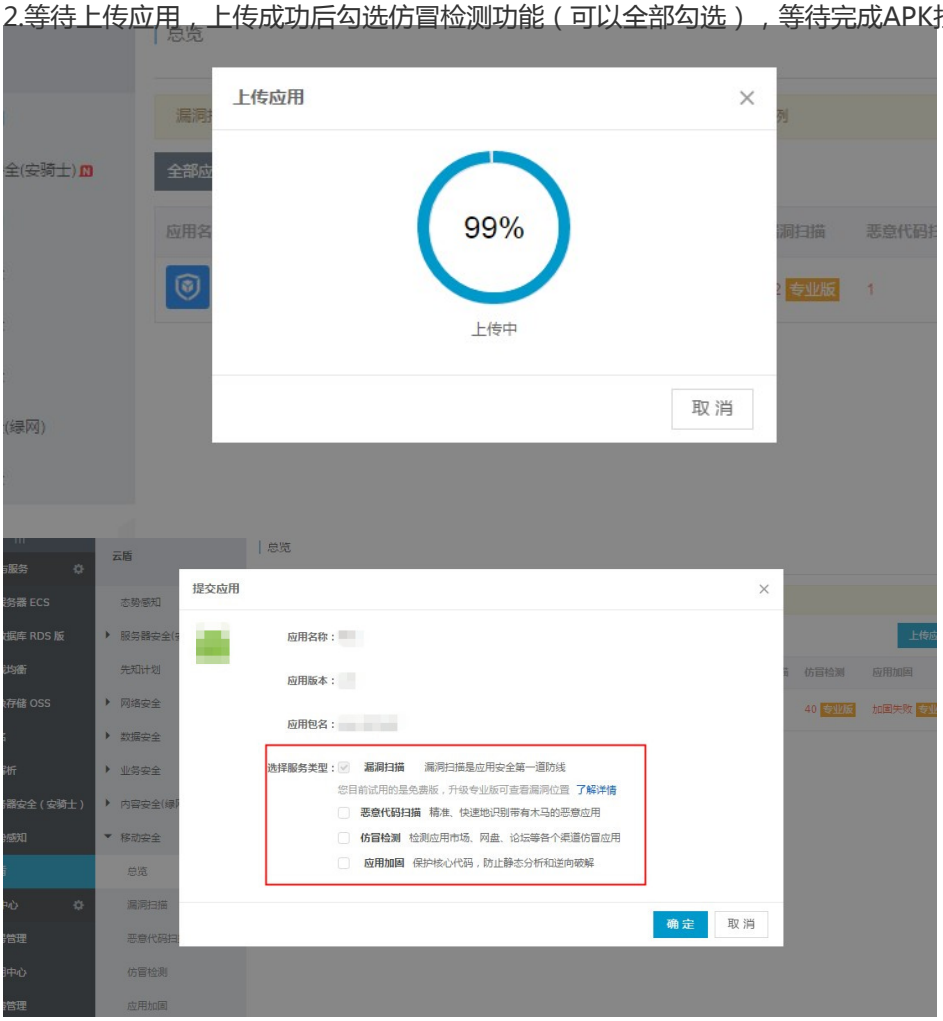
或者单击左边栏 **应用加固**，下载加固包并给应用签名。



当前试用的是基础版应用加固，专业版在此基础上增加SO加固，JAVA常量加固，JAVA指令翻译和JAVA模拟执行等高级加固选项，还提供API接口满足自动化服务的需求。如有需求，请查看 **升级专业版**。

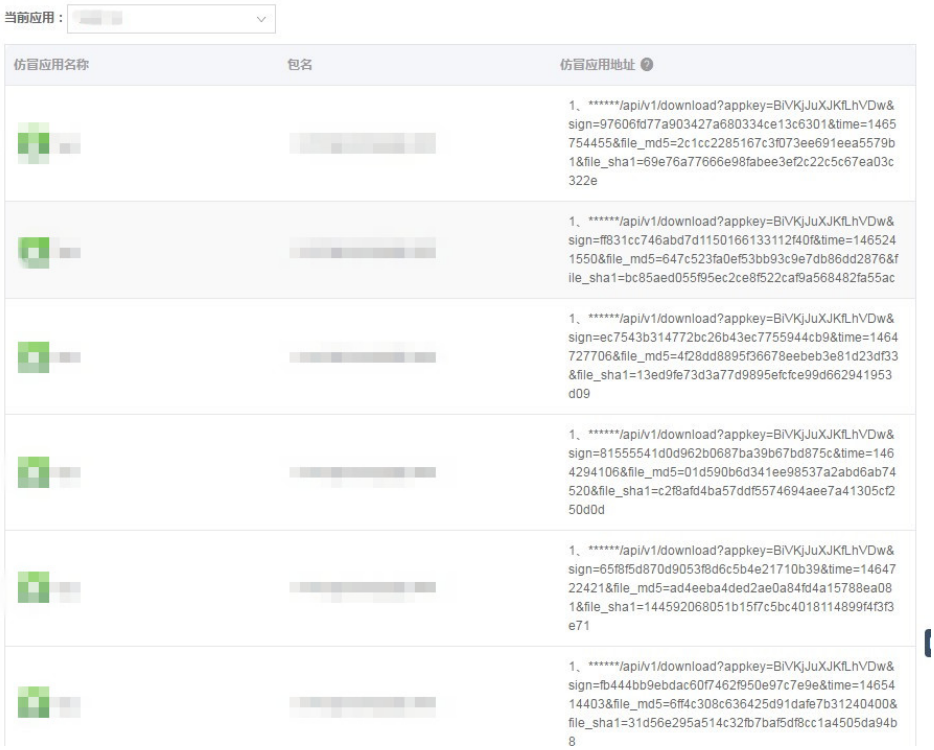
1.点击左侧栏目【总览】，在页面右上角上传APK包

2.等待上传应用，上传成功后勾选仿冒检测功能（可以全部勾选），等待完成APK扫描



3.点击仿冒检测下的数字，进入扫描页面查看仿冒详情

4.页面显示为仿冒应用的名称、包名及仿冒应用地址



5.需要看到具体的仿冒应用地址需要升级到专业版

1.点击左侧栏目【总览】，在页面右上角上传APK包

2.等待上传应用，上传成功后勾选恶意代码扫描功能（可以全部勾选），等待完成APK扫描

3.点击恶意代码扫描下的数字，进入扫描页面查看恶意代码详情

4.页面上方显示为恶意代码的风险等级，下方为恶意代码详情

1.点击左侧栏目【总览】，在页面右上角上传APP

2.等待上传应用，上传成功后勾选安全组件功能（可以全部勾选）

3.下载免费版安全组件，按照接入文档接入集成

4.Android平台提供基础版和企业版，基础版为免费版本，用户可免费下载使用；企业版为收费版本，用户需在线付费购买后方可下载。

iOS平台仅提供收费的企业版，用户需在线付费购买后方可下载。

安全组件接入请查看接入教程

应用开发者在对加固包进行重签名时，建议使用之前对应用签名的keystore，以避免同一应用使用不同签名导致无法正常上传到应用市场的情况。

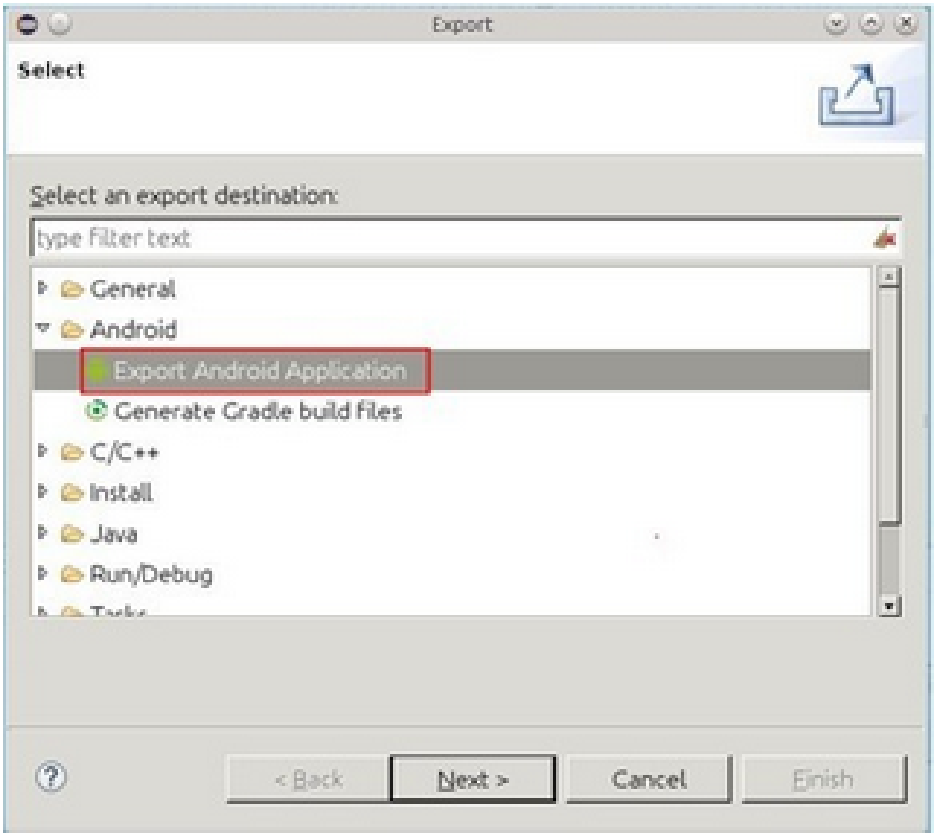
- 使用方法：jarsigner -digestalg SHA1 -sigalg MD5withRSA -verbose -keystore

[your_keystore_path] -signedjar [signed_apk_name] [unsigned_apk_name] [your_keystore_alias]

- 其中，your_keystore_path代表密钥所在位置的绝对路径；
- signed_apk_name代表签名后的安装包名称；
- unsigned_apk_name代表未签名的安装包名称；
- your_keystore_alias代表密钥的别名。

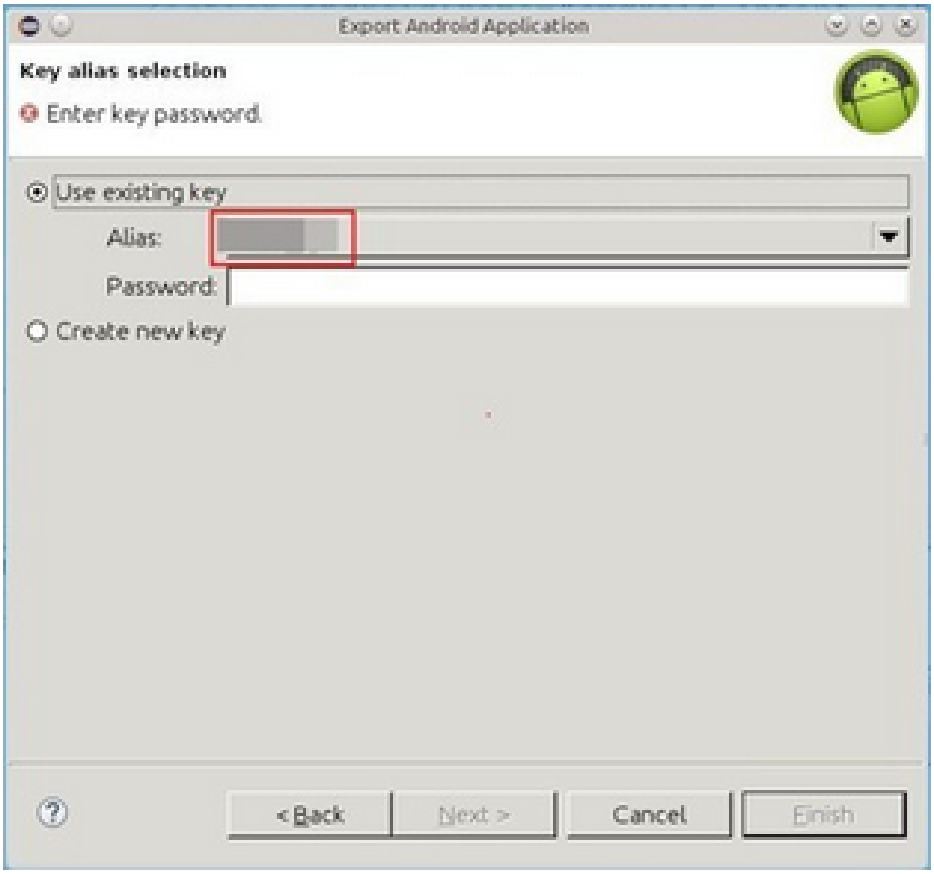
2.获取 keystore

在eclipse项目上点击右键，选择Export，选择Export Android Application并点击Next，Location项中显示的绝对路径即是之前使用过的keystore位置，将此作为步骤1“使用方法“中的your_keystore_path。



3.获取 key alias

在keystore界面点击Next，进入key选择界面。在此界面中可以看到keystore中存储的所有可用key，选择之前对应用签名时使用的key alias，将此作为步骤1”使用方法“中的your_keystore_alias。


Export Android Application
Key alias selection
Enter key password.
Use existing key
Alias:
Password:
Create new key
< Back
Next >
Cancel
Finish