# 2015 年全国大学生信息安全竞赛

# 作品测试报告

作品名称：面向 Android 系统内部数据传递过程敏感信息保护系统

电子邮箱：1242750990@qq.com

提交日期：2015 年 7 月 27 日

# 测试报告

# 1 系统概述

本作品创新性地提出并实现了针对Android系统中敏感数据从产生到使用整个生命周期的自适应性透明加密的保护方案，防止敏感数据被恶意第三方窃取与篡改，保障了用户敏感数据在系统内部传输过程中的私密性及完整性。

用户通过使用本程序软件，可自主根据当前的应用场景进行勾选，配置需要保护的服务以及需要保护的 App 应用。在此之上，本软件能够为用户提供相应的防护功能，防止用户的敏感信息（如短信，GPS 信息）被不法分子窃取。实际测试与使用中，报告如下。

# 2 测试环境

本作品的测试环境如表 1，表 2 所示

表 1 作品测试必要代码与程序

| 名称 | 主要功能 |
|---|---|
| 面向 Android 系统内部数据传递过程的敏感信息保护软件 | 为用户提供敏感数据的保护 |
| 涉及短信敏感信息的 App 应用 | 用于测试短信保护功能方案 |
| 涉及 GPS 位置敏感信息的第三方 App 应用 | 用于测试 GPS 保护功能方案 |
| 基于 SO 库注入的恶意代码 | 用于检验本软件的保护方式是否有效 |

表 2 作品的测试环境

| 名称 | 主要配置 |
|---|---|
| 支持 Android 操作系统的手机 | 型号：TCL S838M<br>系统：Android 4.3<br>内核版本：3.4.0<br>CPU: 1.2GHz<br>内存：2048M |

# 3 测试方案

为了测试本课题组所设计的保护方案是否能真正保护用户的敏感数据，设计了如下测试方案。

## 3.1 保护功能测试

### 3.1.1 测试 1-1 短信保护方案功能测试

#### 3.1.1.1 测试概览

模型建立：使用手机 A 的第三方短信 App 应用未添加保护的前提下，使用该 App 向 10086 发送短信，因为 10086 可以自动回复短信，可以看作是用户双方之间发送的短信的模型。

测试概括：短信保护测试较为复杂，分发送短信敏感信息的保护测试，与接收短信的敏感信息的保护测试。

#### 3.1.1.2 测试方案

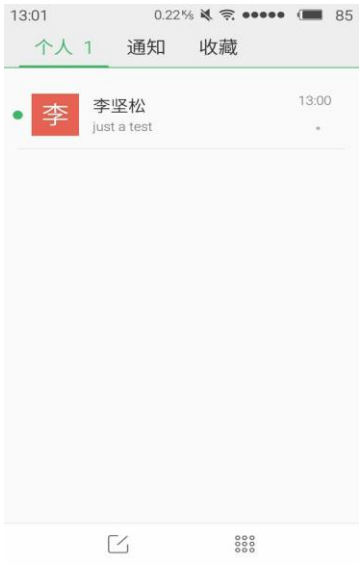
1. 发送短信保护

   测试说明：

   因为添加保护的过程是透明化的，也就是的说用户在 UI 界面写下内容，点击发送的时候，短信的敏感信息产生，并通过 Binder 机制进行数据传输，本课题组的透明化加密是在数据产生时加密，数据离开系统时解密。在系统的传输通道中都以密文传输。为了测试我们的保护方案，我们在 Binder 通信流程中对敏感数据进行拦截，用来验证我们是否保护。

   测试如下:

   保护状态下：打开我们保护的短信软件发送短信，系统处于保护状态，整个保护过程实现对用户的透明化，不会影响用户的正常使用，实际在通道中全程密文传输。


发送情况

接受情况

去除保护下：打开任意未进行保护的短信软件发送短信，程序属于不安全状态，调用短信功能呈现密文状态


关闭保护


同样的内容，关闭保护后，发送呈现密文

2. 接收短信保护

在手机 A 上运行本作品，开启短信保护模块，并对手机 A 上的第三方短信 App 应用添加保护，我们从信息进入操作系统的一刻就进行加密，在传输通道中是密文，然后对用户添加保护的应用，当应用获取这些短信时候帮其解密，未受保护的应用获取的是密文，完成度用户数据的保护

测试如下：

打开本软件的应用管理，选择涉及短信敏感权限的应用，如果对其保护，则该应用获得的短信信息是明文，如果未保护，该应用获得的短信是密文。

以系统自带的短信应用为例：

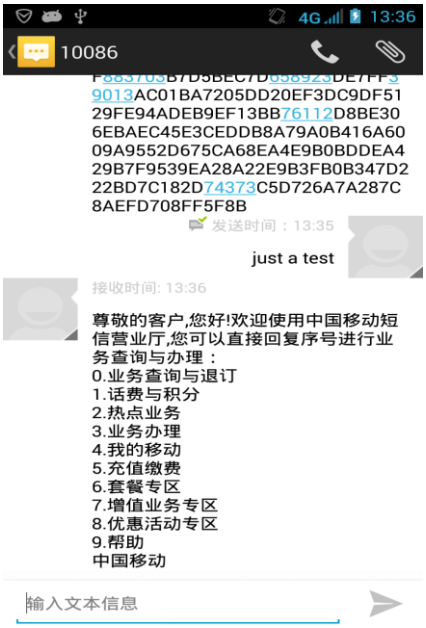
在未保护状态下，所有基于 Binder IPC 通信机制的敏感信息被本课题设计的软件保护，同时提供用户自主选择保护的功能，即若用户未勾选保护，则获取的是乱码，若勾选保护，则被保护软件可正常获得所需通信信息。

未保护状态：

发送短信给 10086：


未保护，不能获得正确信息

勾选保护：


保护，获得正确信息

### 3.1.1.3 测试目的

测试本课题组设计的应用软件是否能实现预期的短信保护方案，保证短信敏感数据在传输过程中不会被恶意用户窃取。

### 3.1.1.4 预期结果

第三方的应用在未开启保护的情况下，能够被恶意代码截获用户的敏感数据，而当开启保护后，恶意代码无法获取用户的敏感数据。另外，在开启短信保护后，用户无需进行额外的操作即可正常使用短信服务，即透明化保护。

## 3.1.2 测试 1-2 GPS 保护方案功能测试

**一、 测试方法**

在手机 A 上运行本软件，开启 GPS 保护模块，考虑到可视化本软件的 GPS 保护效果，这里以腾讯的微信为测试例子，在对微信 App 添加保护的情况下，使用“附近的人”功能获取当前地理位置，查看该位置信息是否正确。

关闭对微信 App 的保护，再次使用“附近的人”功能获取当前地理位置，再次查看位置信息是否正确。

应用未添加保护：

微信：

添加保护


**二、 测试目的**

测试本课题组开发的应用软件是否能实现预期的 GPS 保护方案，保证只有被防护程序保护的 APP 才能正常调用 GPS 功能，获取当前真实的位置信息。

**三、 预期结果**

保证只有当微信 App 被我们的软件保护之后，才能获取真实的位置信息，另外，在开启 GPS 保护后，用户无需进行额外的操作即可正常使用 GPS 服务。

## 3.1.3 测试 1-3 Binder 数据拦截测试(以短信为例)

**一、 测试方法**

在第三方应用程序关闭和开启保护之后，分别使用基于 SO 库注入的恶意代码攻击，对与 Binder 进行通信的 ioctl 函数进行 hook，截获其中传递的短信信息，写到文件中，比较防护开启前后的保护效果。

**二、 测试目的**

验证我们防护方案能否有效地抵御攻击者基于 Android 系统 Binder 数据通信的攻击，保障用户敏感数据在系统内部传输过程中的私密性和完整性。

**三、 预期结果**

在本课题组开发的应用程序对第三方手机应用进行保护之前，恶意应用可以截获 Binder 中所承载的短信信息，但是在本应用软件对第三方手机应用进行保护之后，攻击者就无法获取短信信息。


## 3.2 基本功能测试

### 3.2.1 测试 1-4 应用管理功能测试

**一、 测试方法**

点击应用管理模块进入并使用该功能，在该模块下，选择相应的程序添加/删除保护，保护的语义是：该应用涉及的敏感信息在传输的通道中被透明化加解密，恶意用户攻击的时候，从传输通道中获得的敏感信息是加了密的，所以起到保护的状态；未保护的语义是：通道中的信息是明文，恶意用户截获后，将会获得敏感信息。该模块还可以使用户查看应用所涉及的权限信息。


**二、 测试目的**

测试程序能否正确实现应用管理功能，引导用户完成防护方案的基本配置。

**三、 预期结果**

程序能实现预期的应用管理功能。包括显示，筛选系统程序。显示程序基本信息，以及将程序添加或删除信任。

### 3.2.2 测试 1-5 日志记录功能测试

**一、 测试方法**

点击日志记录模块进入并使用改功能，在该模块下，查看用户的操作记录，给用户提供一个清晰可视的操作记录。


**二、 测试目的**

测试程序是否能够正确实现日志记录功能，记录用户的操作行为。

三、 预期结果

程序能实现预期的日志记录功能，记录用户的操作行为。

### 3.2.3 测试 1-6 计划任务功能测试

一、 测试方法

点击计划模块进入并使用改功能，在该模块下：1.设置系统定时计划，设置系统特定时间段内进行系统全局保护。2.设置涉及短信敏感信息的应用的定时保护计划，在用户设置的特定时间段进行对该应用保护。3. .设置涉及 GPS 敏感信息的应用的定时保护计划，在用户设置的特定时间段进行对该应用保护


二、 测试目的

测试程序是否能够正确实现计划任务功能。

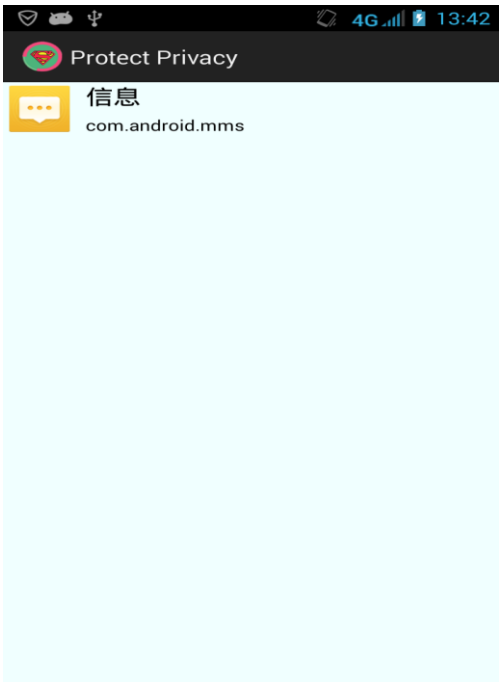
三、 预期结果

程序能实现预期的计划任务功能，完成定时保护的功能。

### 3.2.4 测试 1-7 安全配置功能测试

一、 测试方法

点击安全配置模块进入并使用改功能，在该模块下，管理当前保护方案，可以查看受保护的应用程序以及该应用程序涉及的敏感信息。




## 二、 测试目的

测试程序是否能够争取实现安全配置功能，管理当前的防护方案。

## 三、 预期结果

程序能实现预期的计划任务功能，管理当前已防护程序。