遼寧大學
Liaoning University

基于带权有向图的Android强制访问控制模型

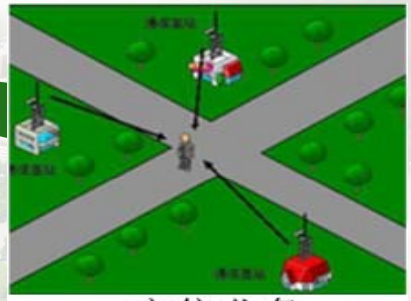
辽宁大学信息化中心
程子傲
2015年11月20日

研究背景及现状

- 移动智能终端以其方便便捷、移动性较强的特点被人们利用在日常生活的各个方面。

健康监测

购物



定位业务

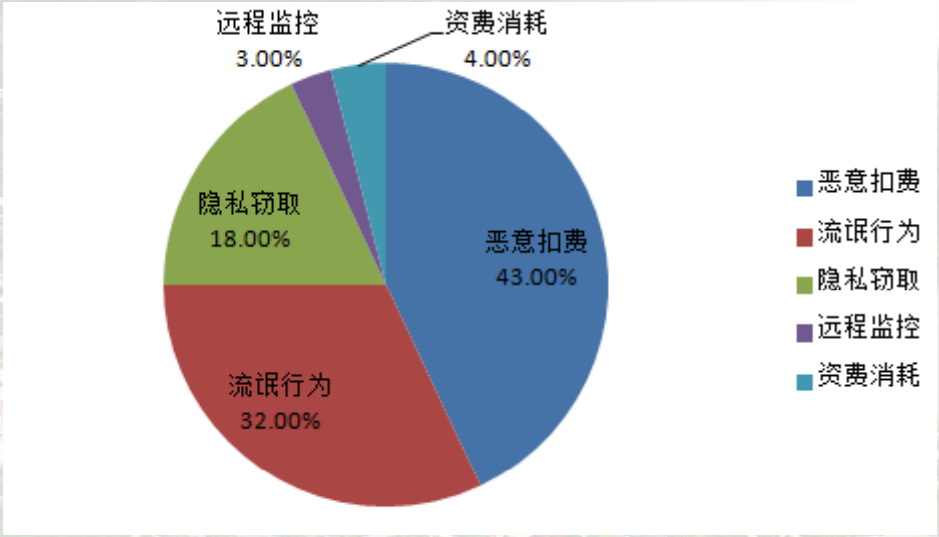
手机支付

聊天

视频会议

研究背景及现状

- 移动智能终端威胁类型
  - 隐私窃取
  - 恶意扣费
  - 远程监控
  - 资费消耗
  - 流氓行为



- 移动智能终端的不足
  - 操作系统自身漏洞
  - 系统资源有限

研究背景及现状

入侵检测研究现状

- Felt等人通过检查进程调用堆栈，防止特权提升攻击中的混淆代理人攻击，但是由于进程间通信方式的多样性，该模型无法检测利用隐蔽信道发生的合谋攻击。
- Bugiel等人通过构建应用间通信连接图的方式，实现了一个可以同时防御混淆代理人攻击和合谋攻击的XmanDroid模型,但模型中的通信连接图是一个无向图，应用程序被卸载后，没有更新通信连接图。
- 蒋邵林等人通过扩充XmanDroid模型节点属性、加入节点颜色、构建有向通信连接图的方式，达到记录通信历史目的改进的强制访问控制模型。但仅考虑了节点颜色的渲染，没有考虑到应用程序结束后，颜色的清洗问题。

主要工作内容

课题研究的主要内容：

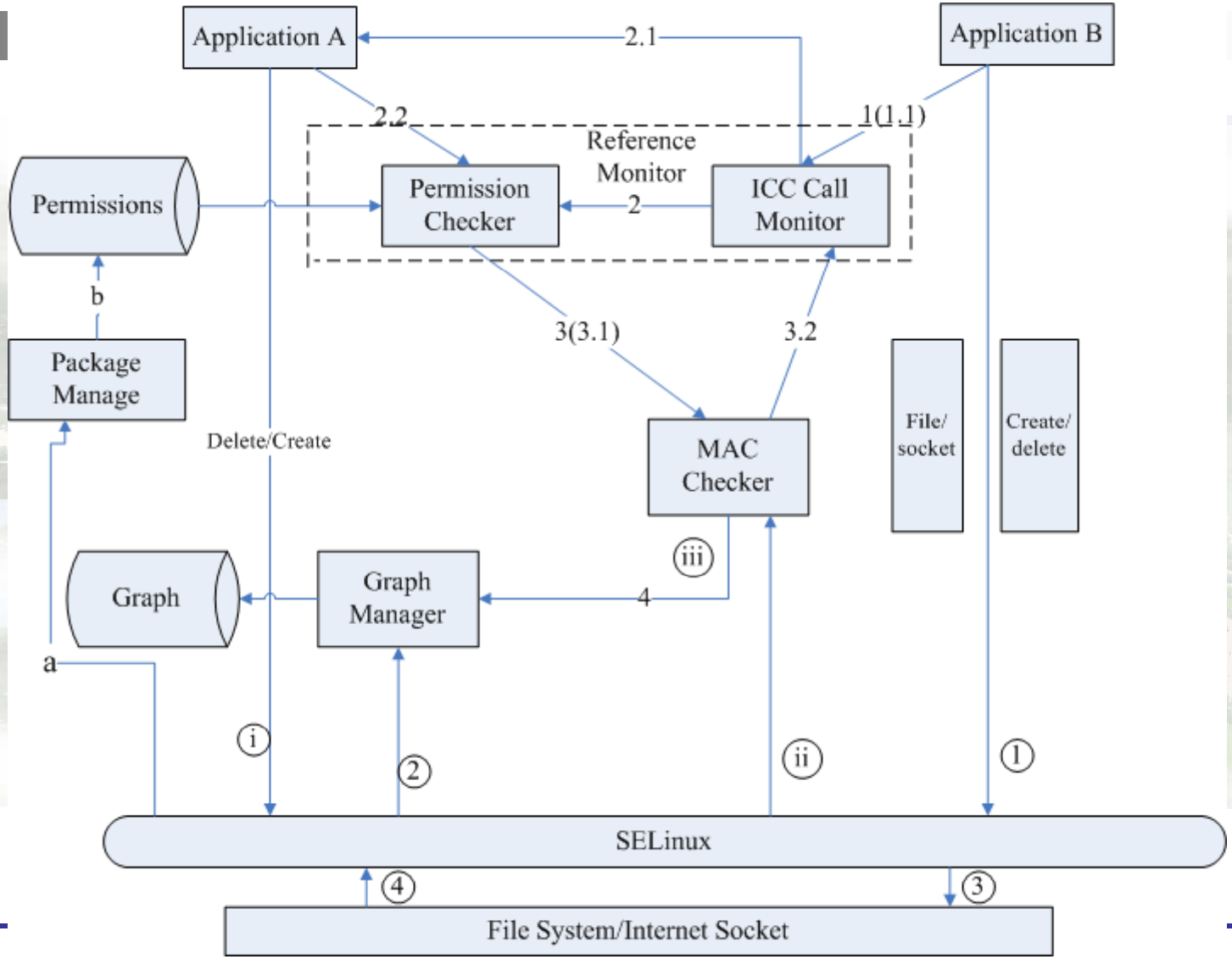
基于带权有向图的强制访问控制模型

主要工作内容

- 改进强制访问控制模型存在的问题
  - 没有考虑到应用程序被卸载后，通信链接图的更新。
- 解决方案

  在模型原有概念的基础上提出基于带权有向图更新节点间存在的通信链路。
  - 引入带权有向图算法，将系统被允许的通讯连接以带权有向图的方式存储。
  - 软件被用户卸载后，利用逆向广度和深度优先搜索算法查找并更新通信连接图。

❖基于带权有向图的强制访问控制的体系结构图



主要工作内容

1.强制访问控制算法

确定因素：节点的Uid属性、节点可信级别。

控制方法：

（1）Uid属性相同，请求被允许；

（2）Uid属性不相同，若两节点可信级别都是trusted，请求被允许；若其中一节点为untrusted，请求被拒绝。

主要工作内容

- 算法1强制访问控制算法。
- 输入：软件程序运行时通信节点V(Gi)，上一通信节点V(Gj)，两节点的属性V(Gi).Uid, V(Gj).Uid
- 输出：是否允许通信请求

```
1.If V(Gi).trustLevel trusted &&  V(Gj).trustLevel trusted
2.return true;
3.Else If V(Gi).trustLevel untrusted && V(Gj).trustLevel untrusted &&
    V(Gi).Uid ==  V(Gj).Uid
4.return true;
5.Else If V(Gi).trustLevel untrusted && (V(Gj).trustLevel trusted ||
    V(Gj).trustLevel untrusted） && V(Gi).Uid  V(Gj).Uid
6.  return false;
```

主要工作内容

2.创建带权有向图的步骤

（1）权限判定：将其交给Permission Check在Permissions数据库中查询，如果存在，调用MAC Check将权限提交给Graph Manager。

（2）构造有向图：应用软件程序运行调用权限节点时，判断节点是否存在节点集中；如果不存在，将节点添加到节点集中；若存在，用有向边将该节点与上节点连接。

（3）确定权值：根据两节点的Uid属性确定权值（w）：

若V(G).Uid==V(G).last.Uid，则w=X；

若V(G).Uid $\neq$ V(G).last.Uid，则w=Y。

主要工作内容

- 带权有向图创建算法如下。
- 算法2 带权有向图Graph1(&DG,V,VR,)
- 输入：软件程序运行时节点V(Gi)
- 输出：按V、VR、w 构造带权有向图G

```
1.  V(Gi)
2.  If V(Gi) ∈ V(G)
3.     Add V(Gi) to V(G)
4.  Else If V(G).index == 1
5.     goto line 1
6.  Else
7.     VR(i)= (V(Gi).last，V(Gi))
8.     Add VR(i)to VR
9.     V(Gi). Indegree = V(Gi). Indegree +1
10.    V(Gi).last.Outdegree = V(Gi).last.Outdegree+1
11.    If(V(Gi).Uid == V(Gi).last.Uid)
12.       w(VR(i))=X
13.    Else
14.       w(VR(i))=Y
```

主要工作内容

3.应用软件被卸载，更新通信链接图

（1）监听软件删除：当应用软件发出删除请求后，Linux操作系统会第一时间监听到请求；

（2）寻找最后更新节点：Graph 数据库中获取最后更新的节点；

（3）逆向搜索过程：

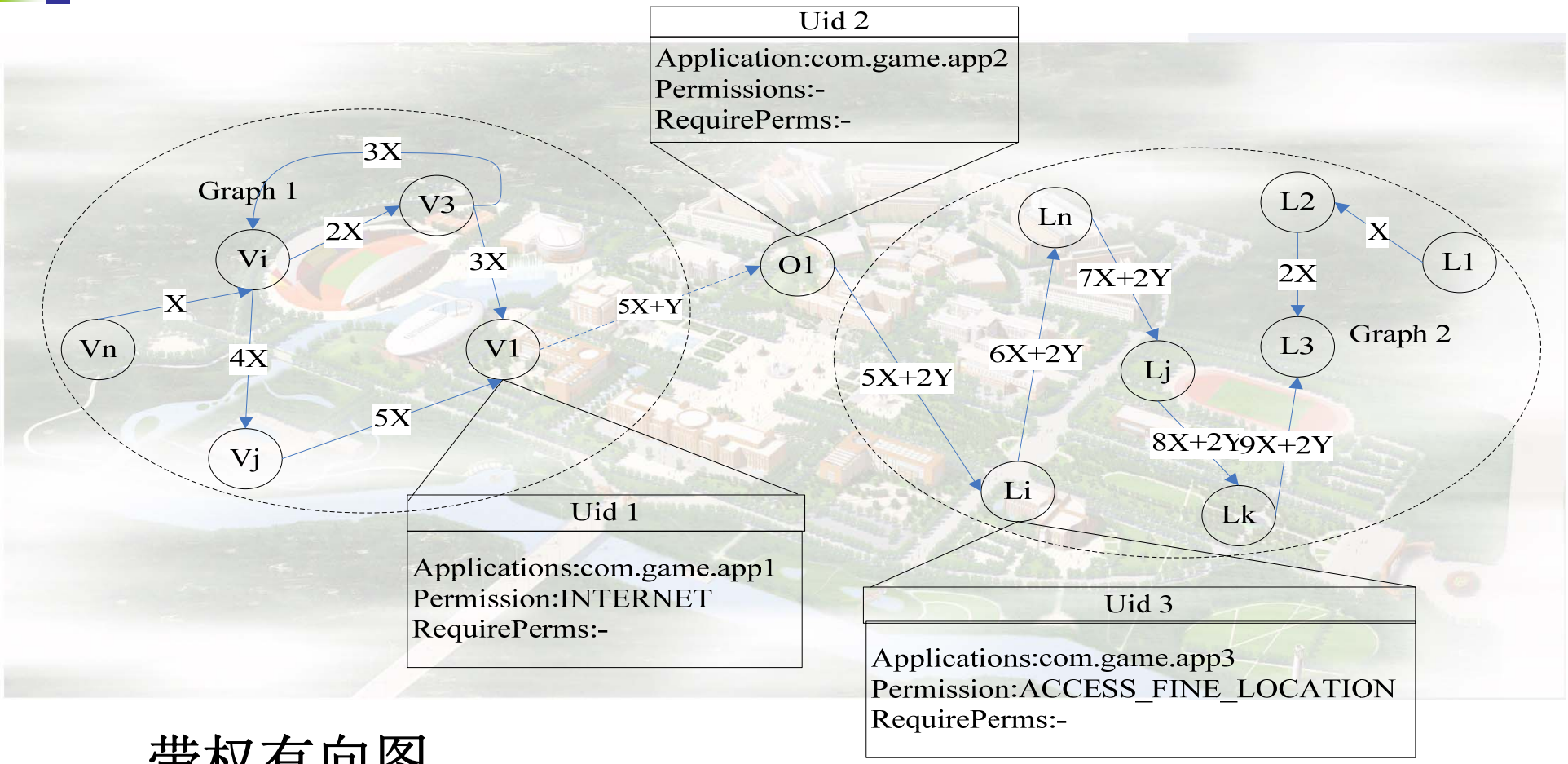
a.搜索以节点为入度的有向边；

b.搜索过程：以广度优先搜索方式与节点链接的有向边的权值与Y相比：

如果权值大于Y继续按照a以深度优先搜索的方式对节点进行递归搜索，直到有向边中两个节点的Uid属性不同时，结束搜索并将有向边删除；

若相反，抛弃不再继续向下搜索。

主要工作内容



带权有向图

主要工作内容

- 逆向广度和深度优先搜索算法。

  算法3 逆向广度和深度优先搜索算法

  输入：带权的有向图graph1(DG,V)

  输出：删除有向图中最后一个大于Y的有向弧

  1. V(Gi)
  2. For j< V(Gi).Indegree
  3. If $\omega(VR(i)) \geq Y$ && V(Gi).Uid== V(Gi).last.Uid
  4. graph1(DG, V(Gi).last)
  5. Else if $\omega(VR(i)) \geq Y$ && V(Gi).Uid V(Gi).last.Uid
  6. Remove VR(i) from VR

主要工作内容

遼寧大學
Liaoning University

二阶隐半马尔可夫模型数据分析算法

- 隐马尔可夫模型存在的问题
  - t+1时刻的状态仅于t时刻状态有关
  - 与状态驻留时间无关
- 解决方案
  - 二阶隐马尔可夫模型中t+1状态不仅与t状态有关，还与t-1时刻状态有关。
  - 隐半马尔可夫模型中通过用状态驻留时间的概率确定任意分布的状态。

主要工作内容

二阶隐半马尔可夫模型

二阶隐半马尔可夫模型 $\lambda=(S,Z,A1,A2,B1,B2,\pi,D)$ ；其中

S代表状态序列，$S=\{s_1,s_2,\cdots,s_N\}$

Z代表观测状态，$Z_M=\{z_1,z_2,...,z_M\}$

A1,A2代表状态转移概率分布：$A1=\{a_{ij}\},A2=\{a_{ijk}\}$ 。其中，a_{ij} 是从状态i转移到状态j的概率，a_{ijk} 是指从状态i转移到状态j，再从状态j转移到状态k的概率；其公式分别为

$$a_{ij}=P(q_t=s_j\,|\,q_{t-1}=s_i),\sum_{i=1}^{N}a_{ij}=1,a_{ij}\geq 0,1\leq i,j\leq 2,1\leq t\leq T \qquad (1.1)$$

$$a_{ijk}=P(q_{t+1}=s_k\,|\,q_t=s_j,q_{t-1}=s_i),\sum_{k=1}^{N}a_{ijk}=1,a_{ijk}\geq 0,1\leq i,j\leq N \qquad (1.2)$$

主要工作内容

二阶隐半马尔可夫模型

B1，B2代表观测值的概率分布， B1表示当前处于状态i时，系统所调用权限序列为 z_u 的概率；B2 表示当前状态处于状态j，而前一个状态处于状态i时，观测到系统所调用的权限序列为z_u 的概率，公式为

$$b_i(u) = P(Z_1 = z_u \mid q_1 = s_i), 1 \le i \le N, 1 \le u \le M \tag{1.3}$$

$$b_{ij}(u) = P(Z_1 = z_u \mid q_1 = s_j, q_{t-1} = s_i), 1 \le i, j \le N, 1 \le u \le M \tag{1.4}$$

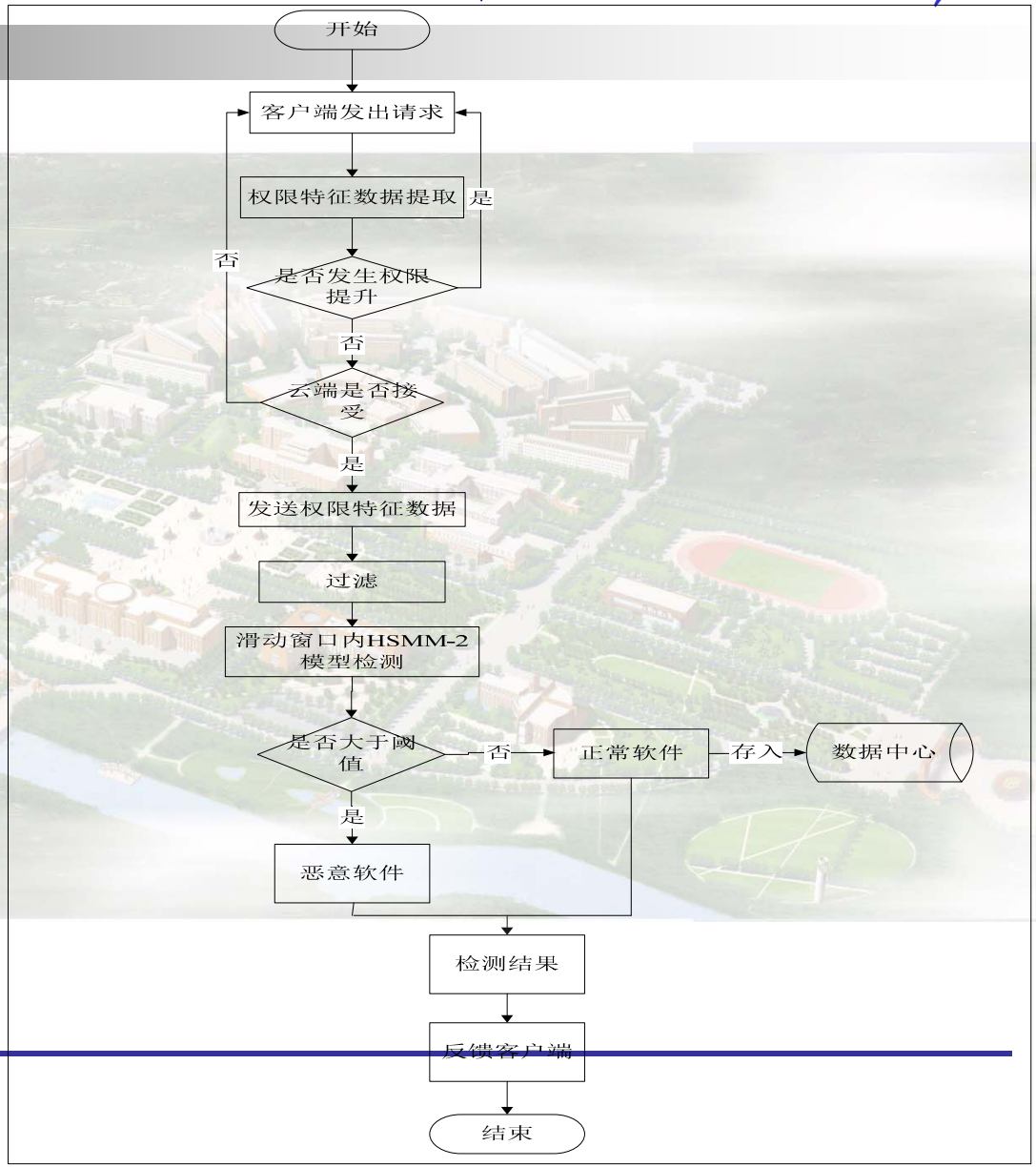
D在状态 s_i 驻留时间为 d 的概率

$$P_i(d) = (a_{ii})^{d-1}(1-a_{ii})\left(d = 1,2,\cdots,D, \sum_{d=1}^{D} P_i(d) = 1\right) \tag{1.5}$$

实验结果与性能分析

工作流程图

客户端和云端通过网络连接,当客户端接受到入侵检测请求时,将调用的权限信息发送到云端,然后由云端采用入侵检测机制算法实现安全检测,并把结果返回给客户端。



实验结果与性能分析

实验数据来源

正常软件主要来自Google Play商城，包括社交、通讯、天气、媒体与视频、购物以及摄影类等程序；恶意软件主要来自VirusShare恶意软件库，包括sky省钱电话、英雄联盟控、鳄鱼爱洗澡等恶意程序。

将数据源获取的实验数据分为6组，每组包括正常软件和恶意软件，且以随机组合的方式存在；每组软件个数分别为100,、200、400、600、800和1000。

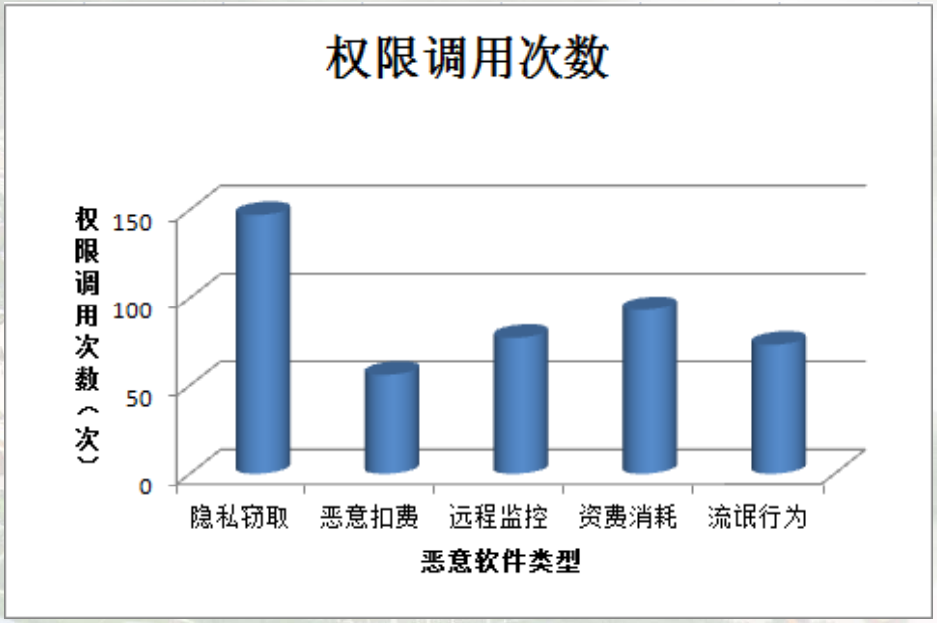
实验结果与性能分析

X,Y值确定

将X设置为1，Y值确定的公式：

$$Y = \sum_{i=1}^{5} \nu_i C_i \qquad (2.1)$$

ν_i 为频繁程度；C_i 为调用的权限次数。5为智能终端五类恶意软件，即隐私窃取、恶意扣费、远程监控、资费消耗、流氓行为计算后Y=85.57，即Y=86

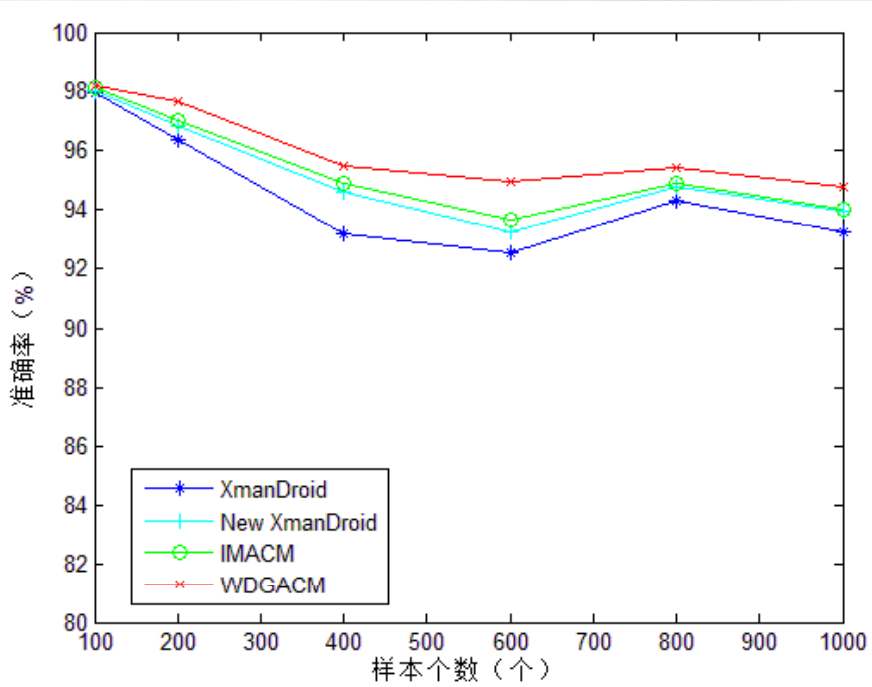


权限调用次数

实验结果与性能分析

❖**XmanDroid、New XmanDroid、IMACM与WDGACM**

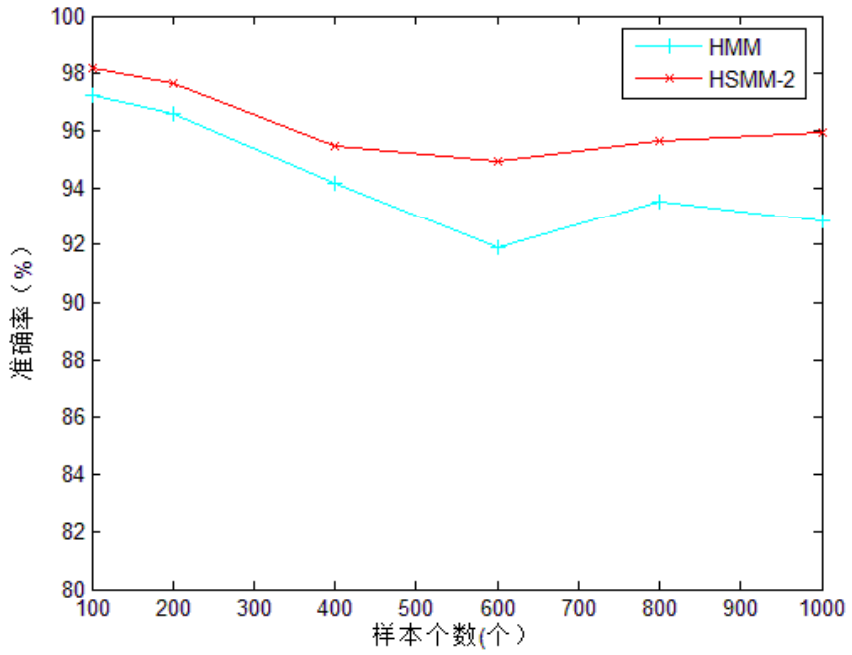
带权有向图强制访问控制模型中更新了被卸载软件的相关通信连接图，避免了对其它软件的影响，提高了检测的准确率。



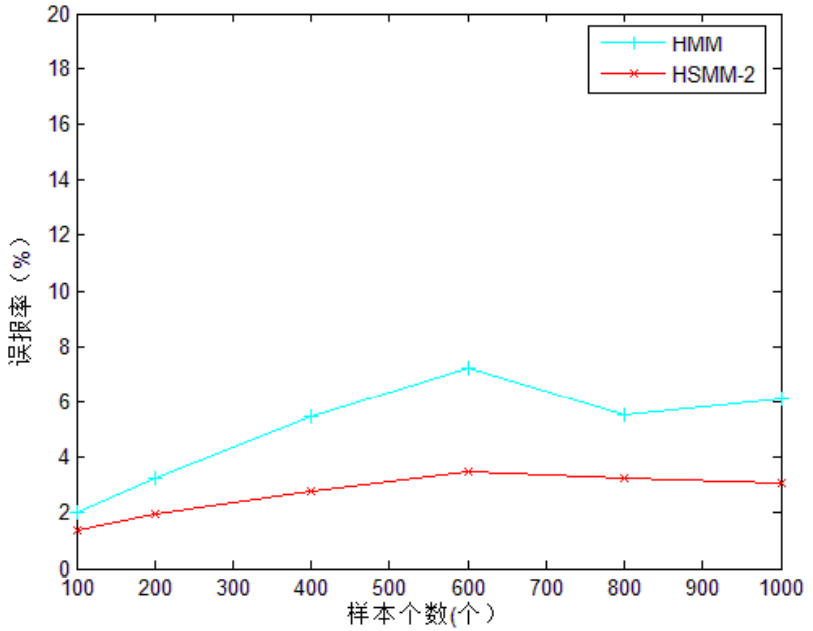
准确率折线图

实验结果与性能分析

❖**HMM、HSMM-2检测准确率和误报率**



准确率折线图



误报率折线图

总结和展望

- 总结

在移动智能终端安全方面进行深入研究，针对终端中软件安装时权限申请的粗粒度性，提出了基于带权有向图的强制访问控制模型，从而保证软件被卸载后，不影响其它软件检测的准确率；

针对软件中存在三个权限组合威胁以及恶意软件攻击的短暂性，提出了二阶隐半马尔可夫模型检测算法。实验表明，以上提出的两种算法在准确率方面表现出了较好的性能，但是第二种算法计算量较大，存在平均检测时间长的缺点。

总结和展望

- 下一步展望

尽管做了大量工作，但仍存在着诸多不足，还可以从以下几方面作出改进：

- 移动智能终端向云端传输数据时，实验中仅按照一个长度传输，并未考虑权限之间相关性或者其它关系，有待进一步提高。
- 入侵检测的算法中，利用二阶隐半马尔可夫模型仅考虑了两个或三个权限组合导致的攻击行为，实质上存在少数的四个或更多危险权限组合，有待进一步研究。
- 检测实验搭建了单节点环境，负载均衡问题有待进一步研究。

谢谢各位老师！