**算法阶段实验报告模板**

修订记录

| 日期 | 修订版本 | 修改章节 | 修改描述 | 作者 |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 实验目的

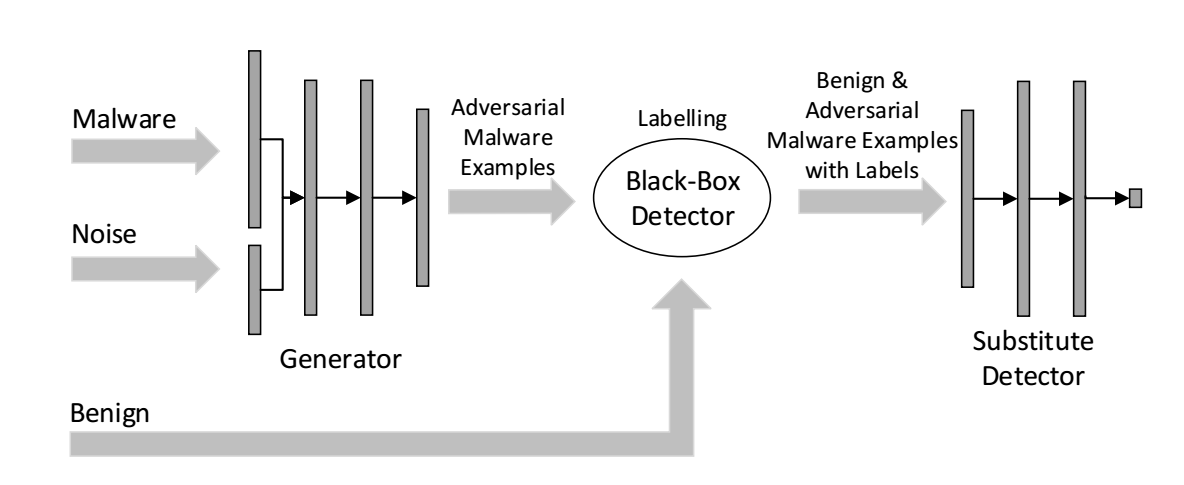
1.掌握 GAN的基本原理与训练方法，并通过python进行编程实现；

2.了解API feature的基本思想，并通过Cuckoo进行批量提取；

3.熟悉Malware欺骗的策略，并构建基于API feature的GAN模型框架。

* 测试MalGAN产生的对抗样本过检测的有效性
* 在各种分类器下MalGAN是否有良好的迁移性
* 验证MalGAN是否对对有一定防御能力（如重训练）的黑盒分类器依旧有显著效果

# 算法原理和流程



Generator是一个生成malware的网络，它接收一个随机的噪声z，通过这个噪声生成对抗malware样本，记做G(z)。

Substitute Detector是一个判别网络，判别一张图片是不是“真实的”。它的输入参数是x，x代表一张图片，输出D（x）代表x为malware的概率，如果为1，就代表100%是malware，而输出为0，就代表不可能是malware。

在训练过程中，生成网络G的目标就是尽量生成对抗样本去欺骗判别网络D。而D的目标就是尽量把G生成的样本和真实的样本分别开来。这样，G和D构成了一个动态的“博弈过程”。

# 实验环境/实验数据

## 环境配置基本情况

深度学习框架:Keras2.0+Tensorflow1.80

CPU:Intel Core i7-6700

RAM:16.00GB

Windows 7,

Cuckoo Sandbox （需在linux下安装windows虚拟机）

https://cuckoo.readthedocs.io/en/0.3.2/usage/

Virusshare.com (账号：yanminglai 密码：123456)

已经筛选好的128维二进制特征向量（运行extract\_apifeatures.py）

## 训练/测试数据说明

网上下载了5000个malware，windows系统里挑出了1357个benign，利用cuckoo对文件进行分析并提取API feature，最终获得了1733个malware特征以及738个benign特征。

采用两种方式进行训练集和测试集的分割：第一种是将80%的数据集作为训练集，其余作为测试集，并且black detector以及substitude detector共用这部分训练集；另一种是将80%的训练集进一步对半分，分别作为black detector以及substitude detector的训练集。

## 算法默认参数情况说明

api\_feature\_dims：决定了输入特征的维数，默认为128；

噪声维数z\_dims：决定了输入generator的噪声维数，默认为20；

隐层维数latent\_dims：决定了generator以及subsitite detector的隐层节点数；

black\_box\_detector：决定了选用的分类器类型，可选参数RF、LR、SVM、MLP以及多种算法的加权组合VOTE；

same\_train\_data：决定了black detector以及substitude detector是否使用相同的训练数据。

# 实验结果

1. True positive rate (in percentage) on original samples and adversarial examples when MalGAN and the black-box detector are trained on the same training set. “Adver.” represents adversarial examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Training Set | | Test Set | |
|  | Original | Adver. | Original | Adver. |
| RF | 98.32 | 2.73 | 98.87 | 3.56 |
| LR | 98.53 | 10.32 | 99.27 | 11.67 |
| SVM | 98.08 | 17.36 | 97.44 | 14.23 |
| MLP | 98.08 | 0.71 | 98.17 | 0 |
| VOTE | 98.72 | 12.13 | 96.71 | 13.51 |

1. True positive rate (in percentage) on original samples and adversarial examples when MalGAN and the black-box detector are trained on different training sets. “Adver.” represents adversarial examples.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Training Set | | Test Set | |
|  | Original | Adver. | Original | Adver. |
| RF | 96.88 | 4.64 | 95.27 | 7.93 |
| LR | 94.07 | 12.53 | 96.84 | 15.24 |
| SVM | 98.17 | 18.84 | 96.71 | 14.26 |
| MLP | 97.98 | 1.52 | 97.81 | 0.63 |
| VOTE | 98.35 | 13.81 | 98.9 | 14.15 |

（3）True positive rate (in percentage) on the adversarial examples after the black-box detector is retrained

|  |  |  |
| --- | --- | --- |
|  | Before Retraining | After Retraining |
| Training Set | 100 | 7.13 |
| Test Set | 100 | 12.31 |

# 实验结果分析和总结

实验一中能够看出RF、LR、SVM、MLP以及VOTE对训练数据都能进行较好的分类，其TPR在当前数据集上基本相当，但其中MLP在对抗样本中的TPR最小，达到了0.71%，这说明了MLP在当前实验中最难以抵抗生成样本的攻击；

实验二中black\_box\_detector及substitute detector分别使用不同的训练集，因此其TPR相对于实验一约下降了一到两个百分点，对抗样本的TPR同样升高了一到两个百分点。

通过实验一和实验二我们知道：1.为了提高black box detector的抗攻击能力，需要尽量使用泛化能力更强的分类器； 2.隐藏black box detector所使用的训练样本也有助于防止black box detector被攻击。

实验三中我们测试了black box detector经过对抗样本再训练的实验，虽然经过对抗样本再训练后，black box detector能够检测出对抗样本，但对新的black box detector再次进行对抗样本生成后，绝大多数的对抗样本依然可以欺骗检测器，这也直观上体现了算法的有效性。