2017 年全国大学生信息安全竞赛

作品报告

作品名称： 基于自反馈学习的大规模恶意域名检测系统

电子邮箱： 2801500253@qq.com

提交日期： 2017 年 7 月 27 日

填写说明

1. 所有参赛项目必须为一个基本完整的设计。作品报告书旨在能够清晰准确地阐述（或图示）该参赛队的参赛项目（或方案）。

2. 作品报告采用A4纸撰写。除标题外，所有内容必需为宋体、小四号字、1.5倍行距。

3. 作品报告中各项目说明文字部分仅供参考，作品报告书撰写完毕后，请删除所有说明文字。(本页不删除)

4. 作品报告模板里已经列的内容仅供参考，作者可以在此基础上增加内容或对文档结构进行微调。

5. 为保证网评的公平、公正，作品报告中应避免出现作者所在学校、院系和指导教师等泄露身份的信息。

目　　录

摘要

本项目为适用于海量数据的在线恶意域名实时检测系统，针对现有的检测技术在海量数据处理以及检测模型更新上的不足，设计并实现了适用于大规模数据的恶意域名实时检测系统，创新性提出抽取小数据集验证更新的做法提升在线学习的效率。经理论论证与实验验证，我们的算法在面对新出现的恶意域名时可以作出及时地反应，并且有着出色的运行效率。本系统还实现了检出域名的进一步分析，对域名相关的威胁情报感知有着启示作用。

Abstract

This project is an online malicious domain name real-time detection system suitable for massive data. In view of the shortcomings of existing detection technology in massive data processing and detection model update, this paper designs and realizes the real-time detection system of malicious domain name suitable for large-scale data. The proposed extraction of small data sets to verify the practice of updating the efficiency of online learning. By theoretical argumentation and experimental verification, our algorithm can respond in time to face new malicious domain names, and has excellent operational efficiency. The system also realized the further analysis of the detection domain name, domain name related to the threat of intelligence perception has a inspiring role.

关键字：域名系统；恶意域名；在线学习；海量数据；云端数据处理平台

Keyword: DNS；Malicious Domain Name; Online Learning; Large-Scale Data；Cloud Computing Platform

第一章 作品概述

1.1 项目背景

随着互联网的发展，以木马、蠕虫、APT 和僵尸网络为代表的恶意软件日益猖獗，对公民隐私、社会经济和国家安全构成严重的威胁。

DDoS 攻击是一种常见的由受控的僵尸网络发动的攻击。僵尸网络是一系列被感染系统的集合，攻击者需要使用 DNS 来解析控制服务器的地址，同时更有 Fast Flux 等技术隐藏攻击的来源，将多个 IP 地址的集合链接到某个特定的域名，并将新的地址从 DNS 记录中换入换出，回避检测。DNS 通讯作为隐蔽通道也开始被攻击者们广泛应用。随机域名生成算法(DGA)是黑产为了逃避检测采用的域名生成算法, 如 conficker 蠕虫每天生成 5 万个可以作为主控服务器的备选域名,但是仅会联系其中 500 个域名完成后续攻击环节。传统的检测方法使用黑名单库的办法来检测恶意域名，但在这种新型隐蔽攻击技术显得束手无策。这样的恶意域名，生存时间短，当黑名单更新的时候很可能攻击已经结束或者域名已经不再可用，需要一种可以快速及时作出响应的检测系统来应对这种攻击。如果在企业中发现类似恶意域名解析请求，那么发起这些请求的设备很有可能是被感染了木马，企业安全团队可以轻易地根据 IP 或 MAC 地址定位，在杀毒软件更新特征库之前发现入侵。

被动 DNS 通过被动捕获内部 DNS 传输来重组 DNS 传输，从而收集数据。Florian Weimer 在 2005 年第 17 届 FIRST 会议上提出这项技术来缓解僵尸网络的传播。被动 DNS 整个流程捕捉到的是服务器到服务器之间的通信内容。这种方式拥有两大重要作用：一是服务器到服务器的通信内容量明显更少，即只包含缓存内不存在的内容。二是服务器到服务器通信不会被轻易关联到某个特定存根解析器处，因此涉及的隐私内容也就相对较少。相较于 URL 分析，域名的流量相对较小，在实时监测时的开销将大大减小。

本项目关注于 DNS 流量，旨在完成海量实时恶意域名监测。建立恶意域名识别能力，可以有效提升发现攻击行为的效果，第一时间迅速发现被木马感染的设备。同时，恶意域名识别，也是大数据安全的重要分析手段，是已建立起大数据安全平台的 CISO 们应首先考虑部署的分析引擎之一。

1.2 相关工作

1.2.1 恶意域名解析

我们以访问恶意域名“taobao.xxx.com”为例，来说明恶意域名的解析过程。

（1）首先请求提供互联网接入的运营商 DNS 服务器，如果不存在该域名，则请求根域名服务器

（2）如果根域名服务器中也不存在该域名，则请求.com 服务器。

（3）“.com”服务器中存在“xxx.com”域名，则请求“xxx.com”服务器。

（4）“xxx.com”服务器中存在“taobao.xxx.com”域名，则返回其 IP 地址

（5）用户连接到假冒的网站“taobao.xxx.com”

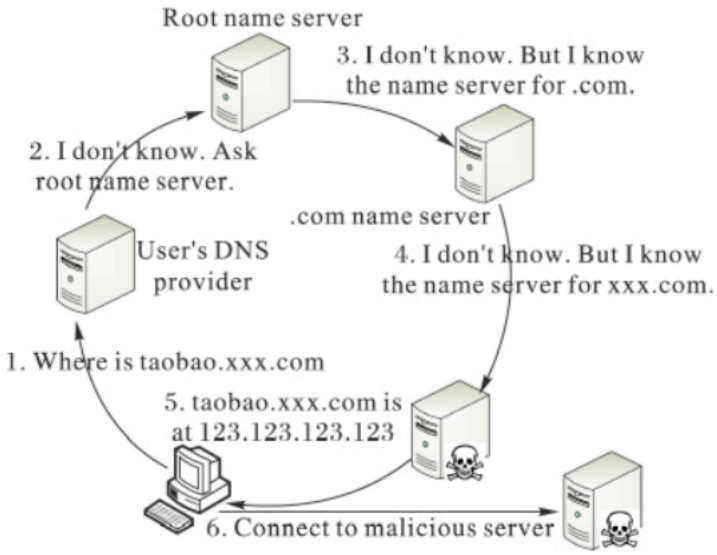

图 1.1 恶意域名解析过程

1.2.2 现有技术对比

近年来，针对恶意域名进行的分析逐渐深入，从最开始的单纯分析域名的词法特性，比如长度、熵值等，到进一步的根据 passive DNS 所表现出的网络特性来分析一个域名是否为恶意，诸如 DNS 响应的 TTL 值和响应类型等。针对恶意域名提取特征进行检测逐步深入。与此同时，一些学者也在原有监督学习的基础上提出了在线、半监督的学习方式，主要特点包括：训练样本数量要求不高，检测模型可更新等等。这些特点给予了我们小组一定的启示，我们认为在当下恶意域名更新迅速的特性下，需要一种可以运用到实践之中的自反馈检测技术，从而做到对于新涌现出的恶意域名的自动化学习与自主更新。

2011 年，Leyla Bilge 等人提出了 EXPOSURE 恶意域名系统[1]，基于监督学习（J48）的方式，抽取了时间、DNS 响应、TTL 值、域名等四个方面的十五维特征，对数据进行训练以完成检测。该系统对 Fast-flux 恶意域名与 Domain-Flux 恶意域名都具有较好的检测效果。

2013 的 sigcomm 会议上，Hongyu Gao 等人提出了使用 Anchor 来解决恶意域名[2]。恶意域名以群组出现而不是唯一的单一域名，如 driven-by exploits 是一连串重定向 url 域名操作，dga 是变换域名作为 c&c 的汇合点。因此，根据 anchor 的相关性挖掘域名集有助于发现恶意域名群组，从而对补充黑名单有所帮助。

已有的恶意域名检测工作取得了一些研究成果，但当应用在实际之中的海量数据时却存在一些问题。这是因为：（1）实际数据中的恶意域名是多种类型混合出现的，针对单一类型的检测模型不能很好的发挥效果。（2）若要做到实时监测，则数据应当流的形式进行处理。这意味着基于统计特征的信息将无法应用于检测当众，需要寻找新的可行特征。为了解决实际应用中的问题，我们认为相对传统的监督学习，半监督、在线学习的方式将更加适合当前环境下的恶意域名检测。

关于半监督、在线学习，Andrew B.Goldberg 等人在 2011 年提出了基于贝叶斯模型的主动半监督在线学习算法 OASIS[3]。使用蒙特卡洛算法模拟粒子生成来计算未标记样本的分布概率，从而基于这一后验概率主动学习。从而在海量未标定的数据中做出判断，并且更新检验模型。但其要求样本的维度较高，应对域名中的高维特征时较为乏力。

Justin Ma 等人提出了一种利用新鲜数据修正恶意 URL 检测模型的办法[4]。基于感知器在线学习算法 Confidence-Weighted。并且实验证明了在线学习中新鲜数据以及数据规模增大对于检测率的提高与帮助。Brad Miller 等人提出了利用人工专家的帮助实现数据标定与学习[5]，而人工标定是有代价且有限的，但对于发现新产生的威胁有着良好的效果。

半监督式的在线学习是可以被应用到恶意域名的检测之中的一个新思想，我们在分析了相关论文之后，创新性地提出了一种适用于海量数据的在线学习算法，在应对实时数据流中的新生成域名有着很好的效果，同时解决了大规模数据中检测模型更新所需的样本标定困难的问题。

1.2.3 市面产品分析

提及恶意域名检测，人们也许会想到市面上常见的 URL 检测系统，如 360 恶意检测，腾讯安全中心等。然而，URL 检测和恶意域名检测却有着许多的差别。

检测 URL 是否为恶意的开销较大，通常可能需要获取整个网页的文本信息，而且企业内部的 URL 链接数目巨大，这将消耗很大的流量资源。相比之下我们系统基于的 Passive DNS 日志具有匿名化、广泛部署的优点，且企业 DNS 解析服务的缓存功能可以大大减少 DNS 流量，待检数据更为简洁。

Passive DNS 数据库允许企业以近实时方式检测缓存投毒以及欺诈变更行为。企业能够定期查询被动 DNS 数据库，从而根据被动 DNS 传感器了解其关键域名当前被映射至哪个具体地址。权威区域数据映射关系中的任何变化都可能意味着企业已经开始遭受恶意攻击。

现有的针对域名是否为恶意的判断大多基于各大网站如 Malware Domain Block List [7]、Phishtank [8]等，这些黑名单根据安全爱好者或者相关机构的检测和上报来更新。当下恶意域名生存周期短，这样的黑名单明显滞后。而我们的系统能够从恶意域名在流量中的活跃特点来检测，并及时更新检测模型。

1.3 特色描述

1.3.1 面向海量数据流的大规模实时检测

本系统对接 DNS 解析流量，逐条域名解析记录抽取特征，并使用 Spark 云平台对海量的流量数据进行并行处理。对恶意域名可在分钟内检测并分析展示，应对大规模数据时有着优越的性能。

1.3.2 在线学习算法在实践中的有效应用

在分割的时间片中我们根据检测结果进行在线学习，更新纠正原有的分类模型。这使得我们在面对一些新出现的恶意域名时可以作出快速的相应，并且提高了检测精度。

1.3.3 创新性抽取小数据集验证

为解决训练样本标定的难题，我们提出了在检测过程中选择小数据集进行自动化验证的办法提高训练效率。

我们提出：距离支持向量机 SVM 训练出的分类超平面较近的样本只占到整个检测数据集的较小部分，而这一部分的检测是可疑的，需要进行二次检测。

在这一思想基础之上我们使用了两个 One-class SVM 一类分类器来分别训练良性模型与恶意模型。并定义两分类器的交集为可疑，用作进一步的标定与样本的更新训练。这一算法在标记成本敏感时尤其重要，可以极大的降低标定样本的成本。

1.3.4 结果实时展示，威胁智能关联

系统中云端检测平台的结果将会在 web 端实时展示，内容包括：

（1）实时恶意域名检出个数

（2）根据 IP 查询关联的恶意域名

（3）恶意域名分类：Phishing、勒索、木马等

1.4 应用前景

恶意域名相关的网络安全问题层出不穷，也在近几年越来越多地得到企业的重视。本系统的目的不仅仅是检测恶意域名，更是发现恶意域名背后的安全威胁。

APT 攻击中的木马程序通常依靠 DNS 通信来隐藏自己的行为，并且利用 DNS 缓慢而长期地传输敏感数据，DDos 攻击与僵尸网络更是和需要 DNS 解析服务来解析其控制服务器的地址。

应用我们的恶意域名检测系统，不仅可以应对大型企业中每天上亿的 DNS 请求，更可以及时地检测出恶意域名，并且精准地发现其背后的安全风险。网络是否正在遭受来自僵尸网络的攻击，内部员工是否正在通过攻击者精心设计的钓鱼网站将公司的机密信息泄露出去，或是内部的网络是否已经暗藏木马，正悄无声息地与攻击者进行着信息传输。随着对域名以及 DNS 的分析，这些威胁将能够被及时感知并扼杀于摇篮之中，所依靠的便正是基于 DNS 的恶意检测。

第二章 作品设计与实现

2.1 系统架构

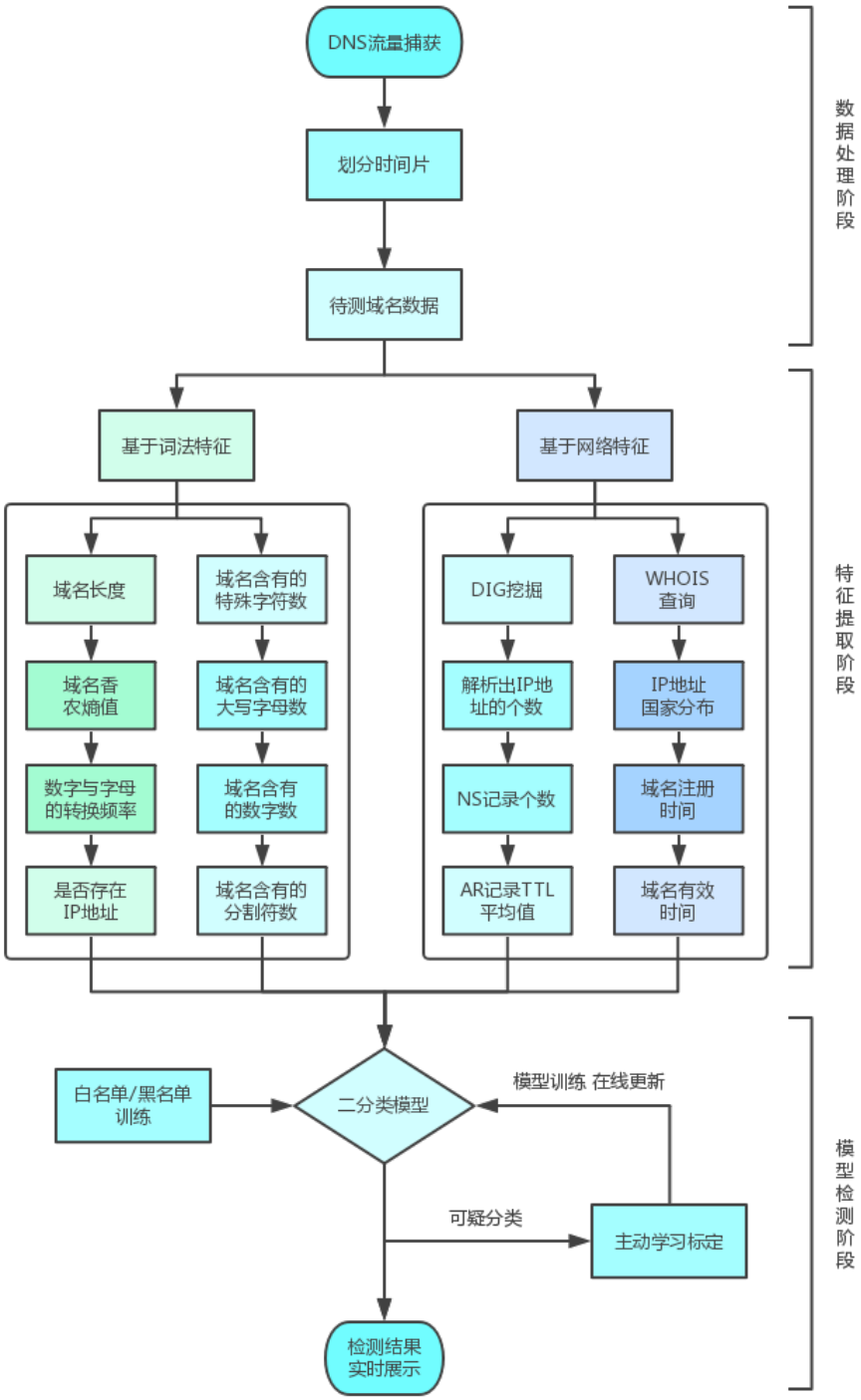

图 2.1 系统架构

如图 2.1 所示，我们整个系统可大致分为三个部分：

第一部分是在用户端，比如企业的流量出口处，图 2.2 中的企业 DNS 解析器。我们将 DNS 流量数据实时地传递给云端的检测平台。

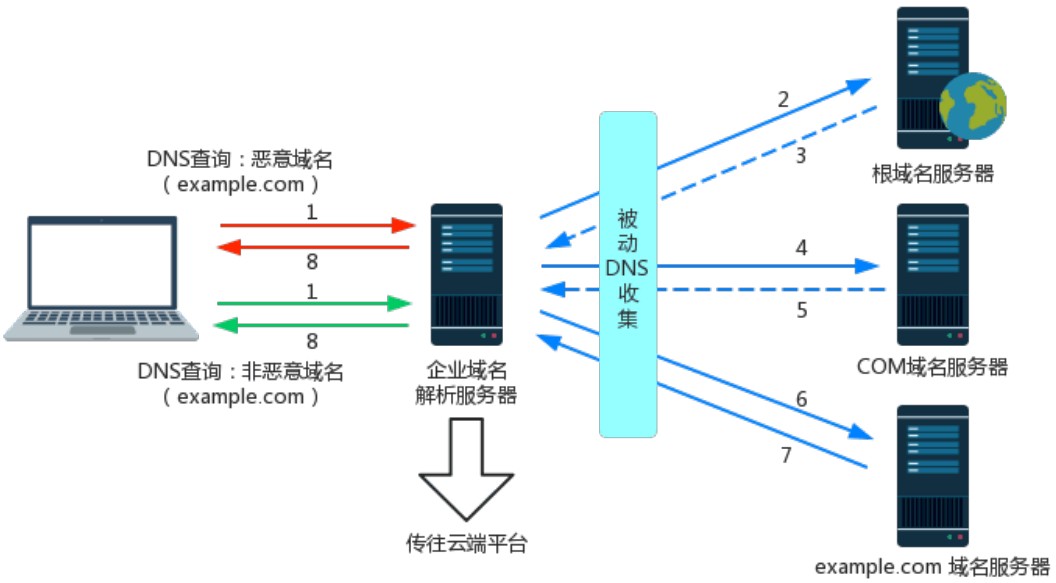

图 2.2 用户端的部署架构

第二部分是系统的核心即云端的检测平台。在这一部分完成对 DNS 流量的特征提取，检测模型训练以及更新，对数据流的并行检测，以及将检测结果传递给下一级 Web 端进行实时展示。

图 2.3 云端处理平台检测

第三部分是 Web 端的检测成果展示。这一部分将向用户提供基础的恶意域名检测数量，以及实时检测排名。同时提供以 IP 为线索查询恶意域名的数据接口。在此基础上我们还对检出的恶意域名再次聚类，清晰而有针对性的展示恶意域名的目的以及威胁。

图 2.4 Web 端展示效果

2.2 实现原理

2.2.1 算法概述

第一阶段是对域名特征的抽取算法。特征集分为词法与网络两部分，并将各个特征分量转化为数字形式进行训练。

第二阶段是检测模型的训练算法。我们使用 1：1 的黑白名单数据，经过第一阶段的特征抽取得到特征向量，采用 SVM 算法训练出第一个检测模型。

第三阶段是在线学习算法。分别使用一类支持向量机训练黑白名单得到两个模型，应用在检测之中得到两个模型交界处的可疑数据，使用可靠的标定系统进行标定得到准确标签后用于新检测模型的训练，得到新的检测模型。

2.2.2 特征抽取

恶意域名的特征可以分为词法方面和网络信息方面。

词法部分是域名本身的相关属性，如长度、熵值等；网络特征则是与 DNS 解析相关的属性如解析出的 IP 地址、域名服务器 NS 记录个数等。一些统计特征如解析出同一 IP 的域名个数等虽然也可以作为恶意域名的一个判断依据，但在流数据的处理

中较难处理。

以下是我们所选取的特征。

针对域名本身我们选择了如下词法特征：

编号	特征	恶意域名可能含有的特征
1	域名长度	域名通常较长
2	是否存在 IP 地址	含有 IP 地址
3	域名香农熵值	熵值较高
4	域名含有的特殊字符数	特殊字符较多
5	域名含有的数字数	数字数较多
6	域名含有的分割符数	分割符数较多
7	域名含有的大写字母数	大写字母较多
8	数字与字母的转换频率	转换频率较大
9	域名是否可读	域名不可读
10	域名中数字串的最大长度	含长字符串
11	域名中字母串的最大长度	含长串字母
12	域名中数字字母的转换率	数字字母转换频繁

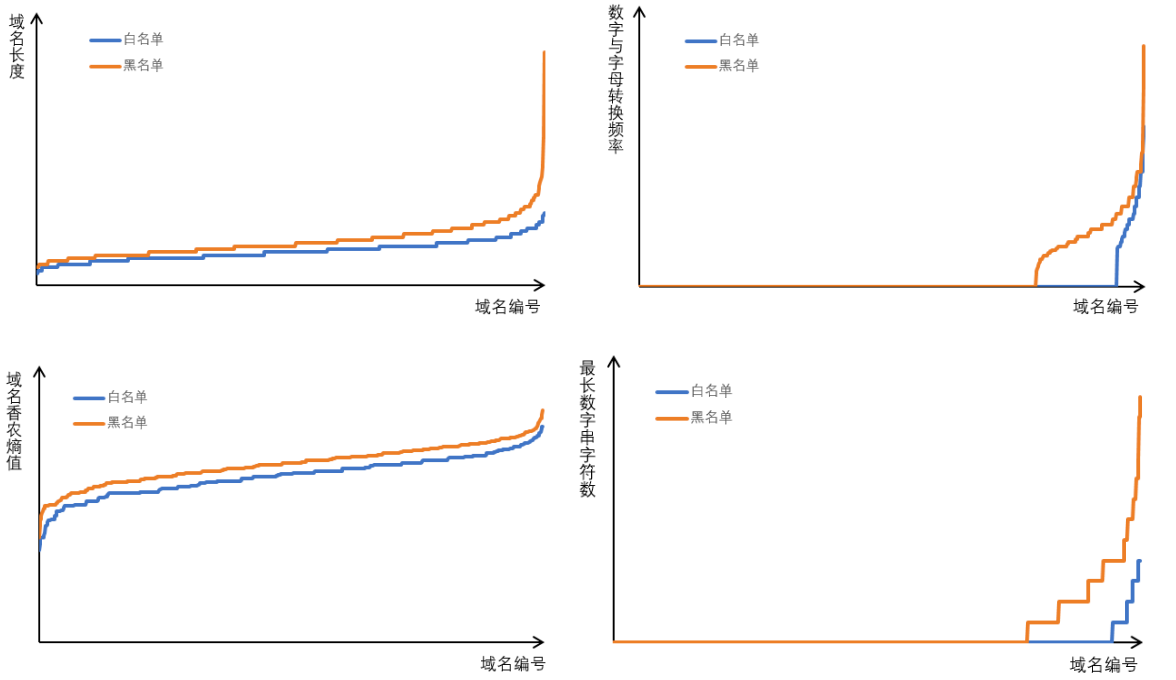

图 2.5 部分词法特征统计分析

在上图中我们选择了黑名单和白名单中各 1000 条数据进行作图，纵坐标为相

应的特征值，横坐标为域名编号，是根据特征值进行升序排列后得到。

- 域名长度：由图 2.5 中第一个统计结果可知，恶意域名的特征表现为长度更长的几率较大，例如 christianmensfellowshipsoftball.org；而我们熟知的非恶意域名 google.com,qq.com 则长度较短
- 是否存在 IP 地址：许多钓鱼域名中包含有 IP 地址,而在合法域名中这是几乎不存在的情况。
- 域名香农熵值：例如 DGA，fast-flux 等恶意域名均为快速随机生成的域名，相比合法域名而言随机性更大，熵值更高。
- 域名含有的特殊字符数：合法域名一般可读性较高，更容易记忆，因此含有特殊字符数较少。
- 域名含有的数字数：合法域名一般含有数字较少而恶意域名中一般含有大量数字如 911718.net。
- 域名含有的分割符数：合法域名含有分隔符数目一般较少甚至没有，而恶意域名则含有更多如 french-wine-direct.com
- 数字与字母的转换频率：由图 2.5 第三个图中统计结果可知，恶意域名的特征表现为数字与字母的转换频率较大的几率较大。

针对域名背后的网络信息我们选择了如下网络特征：

编号	特征	恶意域名可能含有的特征
1	解析出 IP 地址的个数	解析出多个 IP 地址
2	NS 记录个数	多个 NS 记录
3	NS 记录 TTL 平均值	TTL 平均值较小
4	AR 记录个数	多个 AR 记录
5	AR 记录 TTL 平均值	TTL 平均值较短
6	注册时间	注册时间较短
7	IP 地址国家分布	国家分布不均
8	有效时间	有效时间较短
9	A 记录 TTL 平均值	TTL 平均值较短

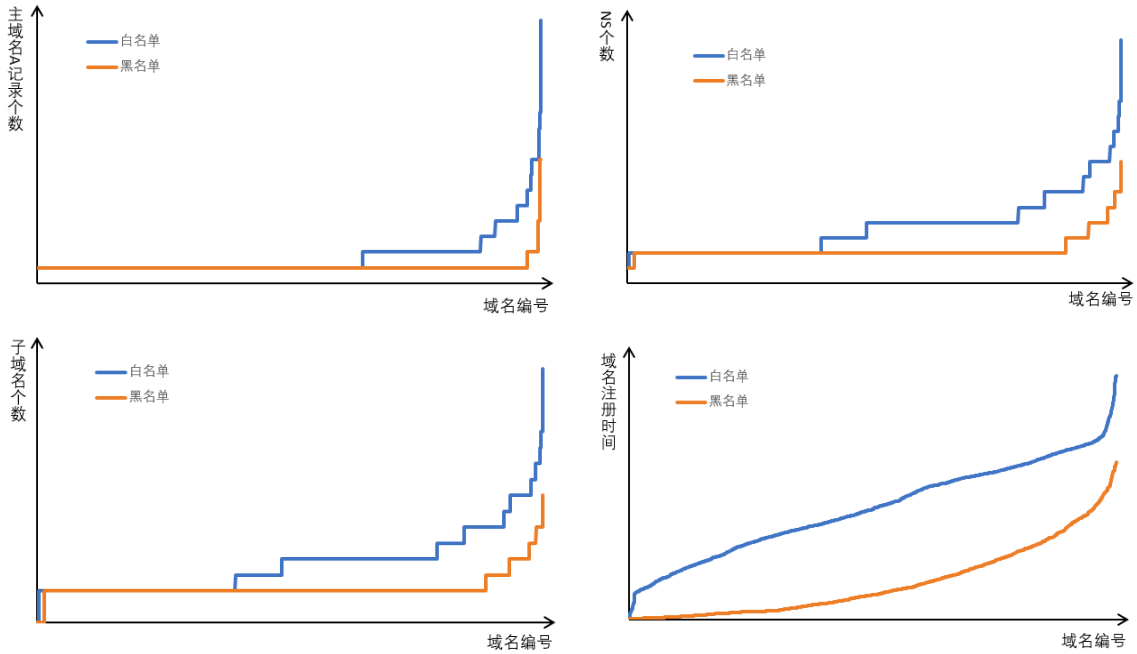

图 2.6 部分网络特征统计分析

在上图中我们选择了黑名单和白名单中各 1000 条数据进行作图，纵坐标为相应的特征值，横坐标为域名编号，是根据特征值进行升序排列后得到。

- 解析出 IP 地址的个数：恶意域名每隔一段时间都会更新其代理机器，因此在一定时间内查询该域名得到的 ip 地址会持续增加，累计得到的 ip 数也会非常大。
- NS 记录个数：由图 2.6 第二个图中结果可知恶意域名表现出 NS 个数更多的特征更明显
- AR 记录 TTL 平均值：恶意域名为提高健壮性，会频繁更换 ip，因此恶意域名拥有者会将 DNS 缓存时间设置较小。
- 注册时间：由图 2.6 第四个图中结果可知,恶意域名通常注册时间较短，大部分常用合法域名注册可达 10 年以上
- IP 地址国家分布：由于恶意域名拥有者往往掌控大量僵尸网络，因此受感染主机分布影响，恶意域名 ip 会分布在多个国家，而正常域名一般在一个或少数几个国家中。

2.2.3 SVM 模型训练

在机器学习中，支持向量机（Support Vector Machine，常简称为 SVM，又名支持向量网络）是在分类与回归分析中分析数据的监督式学习模型与相关的学习算法。给

定一组训练实例，每个训练实例被标记为属于两个类别中的一个或另一个，SVM 训练算法创建一个将新的实例分配给两个类别之一的模型，使其成为非概率二元线性分类器。SVM 模型是将实例表示为空间中的点，这样映射就使得单独类别的实例被尽可能宽的明显的间隔分开。然后，将新的实例映射到同一空间，并基于它们落在间隔的哪一侧来预测所属类别。

除了进行线性分类之外，SVM 还可以使用所谓的核技巧有效地进行非线性分类，将其输入隐式映射到高维特征空间中。

恶意域名的检测是一种分类问题，根据特征形成的向量来进行判断域名是良性还是恶意。支持向量机 SVM 便是这样一种二类分类模型，其基本模型定义为特征空间上的间隔最大的线性分类器，其学习策略便是间隔最大化，最终可转化为一个凸二次规划问题的求解，其模型本质是形成一个多维的超平面来实现分类。

具体原理：

(1)线性可分的:

对于一个两类的分类问题，数据点用 x 来表示构成一个 n 维向量，w^t 中的 t 代表转置，而类别用 y 来表示，可以取 1 或者 -1 ，分别代表两个不同的类。一个线性分类器的学习目标就是要在 n 维的数据空间中找到一个分类超平面,该超平面可描述为:

$$g(x)=w^t x+w_0 \qquad (1)$$

其中，w 是权向量， w0 是阈值权或者偏置。对于 g（x）＞ 0，则该样本在决策面的上方；反之，则在决策面的下方。 一个点距离超平面的远近可以表示为分类预测的确信或准确程度,SVM 就是要最大化这个间隔值。而在虚线上的点便叫做支持向量 Supprot Verctor。

最优超平面是使得每一类数与超平面距离最近的向量与超平面之间的距离最大的平面，即求使公式（2）最小化的 w。

$$f(w)=\frac{||w||^2}{2} \qquad (2)$$

对于这种二次凸优化问题，可以采用拉格朗日函数求解。问题的解是通过求公式（3），使拉格朗日乘子 α 最大化，w 和 w_0 最小化得到的。

$$L(w,w_0,\alpha)=\frac{||w||^2}{2}-\sum_{k=1}^{n}\alpha_k((wx_k+w_0)y_k-1) \quad (3)$$

上式中，α为拉格朗日乘子，n 是样本的个数，y_k是决策属性的值。通过对公式（3）的 w 和 w0 求偏导，利用 Kiihn—Tucker 条件，将 w 转化为α的函数，得到

$$w(\alpha)=\sum_{k=1}^{n}\alpha_k-\frac{1}{2}\sum_{j,k=1}^{n}\alpha_j\alpha_k y_j y_k(x_j x_k) \quad (4)$$

并且$\sum_{k=1}^{n}\alpha_k y_k=0$

通过公式（4），就能解出 w 和 w0。从而得到最优的决策面。

(2)线性不可分的：

对于线性不可分的情况，常用做法是把样例特征映射到高维空间中去(如图 2.7)：

对于非线性的情况，SVM 选择一个核函数，通过将数据映射到高维空间，来解决在原始空间中线性不可分的问题。由于核函数的优异特性，这样的非线性扩展在计算量上并没有比原来复杂多少。

在线性不可分的情况下，支持向量机通过某种事先选择的非线性映射(核函数)将输入变量映射到一个高维特征空间，在这个空间中构造最优分类超平面。我们使用 SVM 进行数据集分类工作的过程首先是同预先选定的一些非线性映射将输入空间映射到

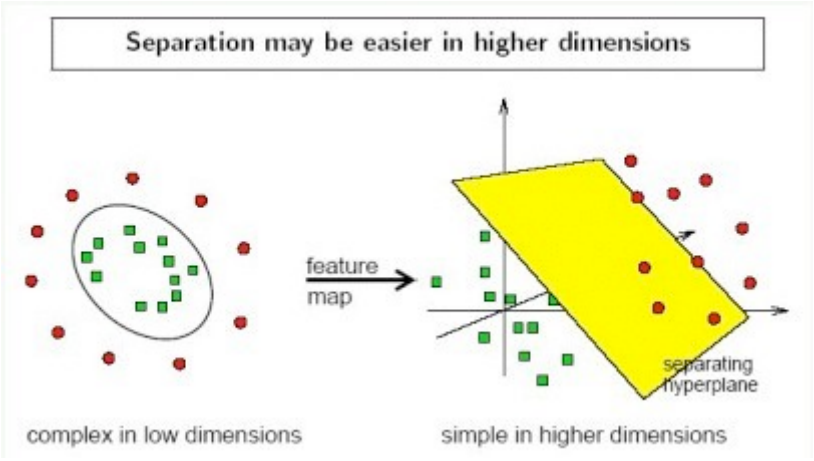

图 2.7 SVM 的高维映射

高维特征空间，使得在高维属性空间中有可能最训练数据实现超平面的分割，避免了在原输入空间中进行非线性曲面分割计算。SVM 数据集形成的分类函数具有这样的性质：它是一组以支持向量为参数的非线性函数的线性组合，因此分类函数的表达式仅和支持向量的数量有关。

但是如果不是因为数据本身是非线性结构的，而只是因为数据有噪音。对于这种偏离正常位置很远的数据点，我们称之为 outlier ，在我们原来的 SVM 模型里，outlier 的存在有可能造成很大的影响，因为超平面本身就是只有少数几个 support vector 组成的，如果这些 support vector 里又存在 outlier 的话，其影响就很大了。为了处理这种情况，SVM 允许数据点在一定程度上偏离超平面，而不会使得超平面发生变形。

2.2.3 fSVM 在线反馈学习算法

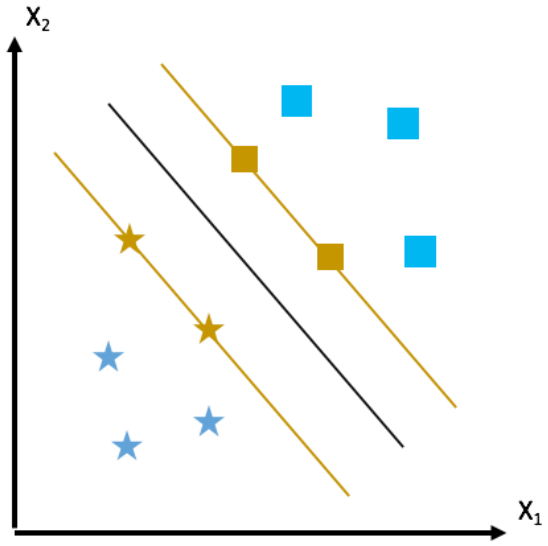

图 2.8 在训练数据集上计算超平面

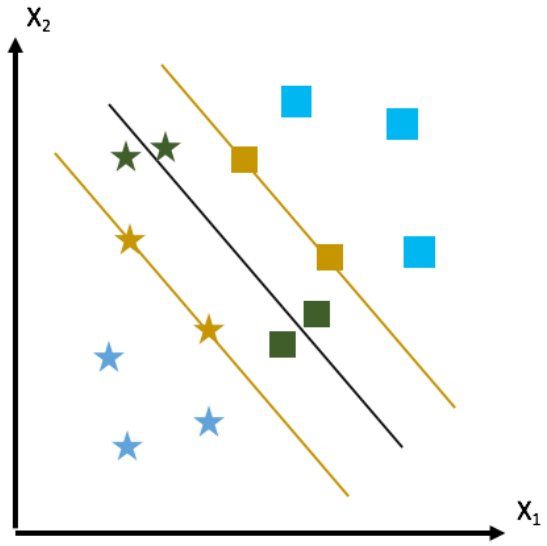

图 2.9 在检测过程中发生误检

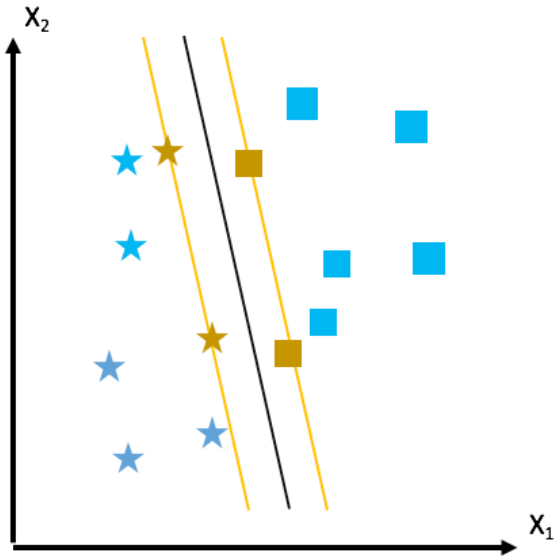

图 2.10 检测数据对训练模型起到更新作用

通过上面三张图所展示的情况可以看出，在训练数据集上训练好模型后仍有可能在实际的待检数据中产生误检，而如果能够对易误检的数据进行二次标定便可以对检测模型有反馈的更新作用。

通过对 SVM 原理相关文献的阅读我们推导得出了检测数据置信度的评估函数。

$$f(x_1) = \exp\left(-\frac{1}{|d(x_i)|}\right)$$

待测样本距离超平面的距离与其置信度呈指数反比关系。简言之，距离超平面较

近的点，置信度较小，容易发生误检；而距离超平面较远的点，置信度较大，结果更为可信。

在第三章测试部分中我们发现，真实的数据中单纯应用监督训练模型会有较大的误差，且往往这样的误检情况会持续一个短暂的时间。我们分析认为，这是因为恶意域名往往生存周期较短，且较为集中地出现，一些新产生的恶意域名特征可能没有被之前的训练模型所学习，从而造成误检。因此为了应对新产生的恶意域名，模型需要重新训练。

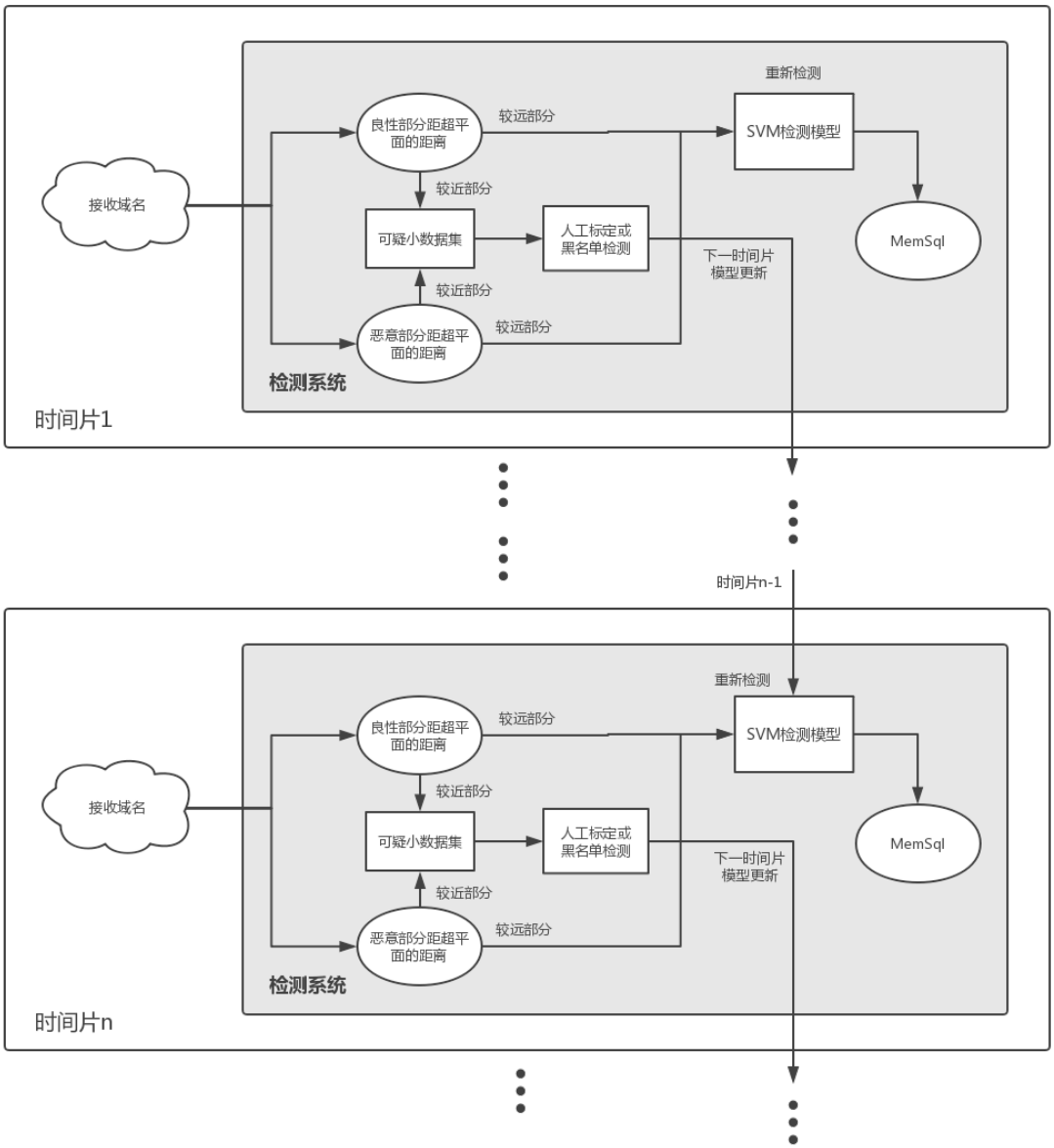

图 2.11 在线学习系统框架

然而，重新训练支持向量机所需要的代价很大，我们不可能每次都对新数据进行标定并训练。为解决海量数据中的标定问题，提高我们在线学习算法的效率，我们在大规模的域名数据中抽取了一部分容易产生分类错误的

我们将海量的流量数据划分时间片，在每一个时间片中检测出一个小数据集用于模型的在线学习与更新。具体算法如下：

算法1：在线学习算法

```
1: function OnlineLearning(D, svm)
2:     Gap = 0.2;
3:     Update = ∅;
4:     //Update
5:     for i in D do
6:         if |Distance(Di)| < Gap then
7:             Update = Update ∪ Di
8:     end for
9:     if size(Update) 大于时间片的 0.1
10:        Gap = Gap / 2
11:        repeat Update
12:    if size(Update) 小于时间片的 0.02
13:        Gap = Gap * 1.5
14:        repeat Update
15:    使用 Update 更新样本库
16:    重新训练 svm
17:    return(svm)
18: end function
```

图 2.12 在线学习算法

算法2：自动标定过程

```
1:function Label(domain)
2:      如果域名落在检测模型正侧：
3:          上传至 virustotal 检测
4:          结果为恶意 return 1
5:          获取 baidu 搜索结果
6:          结果在 10000 条以上
7:          return 0
8:          结果中包含恶意关键字
9:          return 1
10:     如果域名落在检测模型负侧：
11:         上传至 virustotal 检测
12:         结果为恶意 return 1
13:         获取 baidu 搜索结果
14:         结果在 5 条以下 return 1
15:         结果在 10000 条以上 return 0
16:         结果包含关键字 return 1
17: end function
```

图 2.13 自动标定算法

2.2.4 云端数据处理计算平台

云端计算平台是本系统的核心检测部分，基于 Hadoop/Spark 平台所搭建，具有高效、稳定、可伸缩的特点。它的主要功能有：

（1）SVM 和 One-classSVM 模型的训练

（2）海量域名流的接收与检测

（3）实时信息处理与展示

为实现实时信息处理功能，我们采用了 Spark Streaming 技术。

Spark Streaming 是一种构建在 Spark 上的实时计算框架，它扩展了 Spark 处理大规模流式数据的能力。

Spark Streaming 的优势在于：

- 能运行在 100+的结点上，并达到秒级延迟。

- 使用基于内存的 Spark 作为执行引擎，具有高效和容错的特性。
- 能集成 Spark 的批处理和交互查询。
- 为实现复杂的算法提供和批处理类似的简单接口。

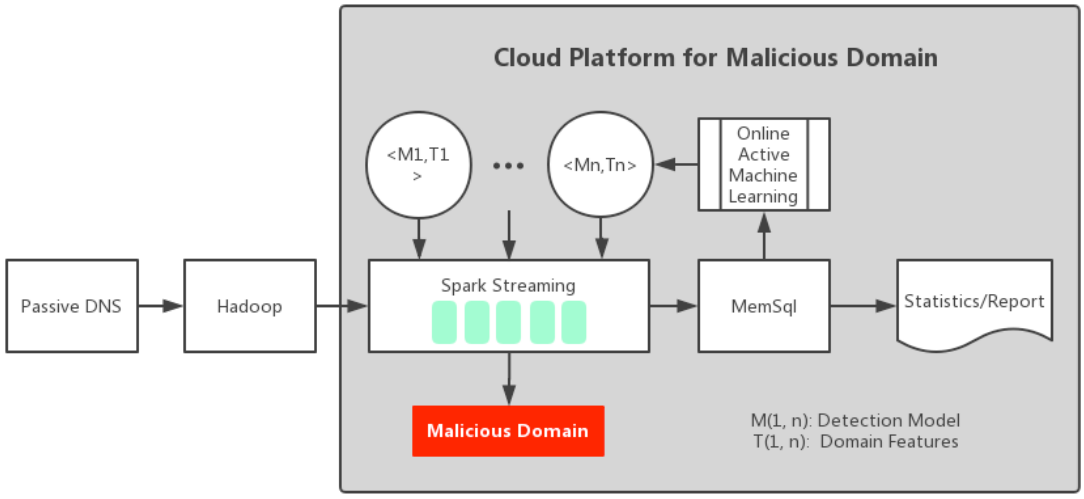

图 2.14 云计算平台处理示意

Spark Streaming 把实时输入数据流以时间片 Δt （如 1 秒）为单位切分成块。并会把每块数据作为一个 RDD，并使用 RDD 操作处理每一小块数据。每个块都会生成一个 Spark Job 处理，最终结果也返回多块。

Spark Streaming 的另一大优势在于其容错性，RDD 会记住创建自己的操作，每一批输入数据都会在内存中备份，如果由于某个结点故障导致该结点上的数据丢失，这时可以通过备份的数据在其它结点上重算得到最终的结果。

正如 Spark Streaming 最初的目标一样，它通过丰富的 API 和基于内存的高速计算引擎让用户可以结合流式处理，批处理和交互查询等应用。因此 Spark Streaming 适合一些需要历史数据和实时数据结合分析的应用场合。当然，对于实时性要求不是特别高的应用也完全胜任。另外通过 RDD 的数据重用机制可以得到更高效的容错处理。

2.3 功能简介

2.3.1 客户端功能

客户端收集 Passive DNS 数据并上传到云检测平台。Passive DNS 数据的具体收集方式多种多样，管理员可以使用许多免费的程序如 dnstap 等直接读取。本实验中所采用的 DNS 数据为通过 pcap 日志网络抓包处理得到，保留了后续分析所需要的特征值。

2.3.1 云端平台功能

（1）海量域名流的接收处理

（2）SVM 和 One-classSVM 检测模型的训练

（3）检测结果的存储、再聚类，并将组织结果信息发送至 Web 端实时展示。

2.3.2 Web 端展示功能

Web 端对实时检测的结果进行展示，内容主要包括：

（1）检测出的恶意域名

（2）恶意域名出现次数排名

（3）恶意域名服务器所在地

（4）访问恶意域名的本地主机 ip

2.4 指标

本节提出了用于检测算法有效性与系统性能的一些指标与作用，具体数据将在第三部分的测试报告中展示。

2.4.1 机器学习检测指标

常用检测指标包括以下四条：

True Positive（真正,TP）：被模型预测为正的正样本；

True Negative（真负, TN）：被模型预测为负的负样本；

False Positive （假正, FP）： 被模型预测为正的负样本；

False Negative（假负 , FN）被模型预测为负的正样本。

在模型训练阶段，我们引入 F1 分数（F1 Score），是统计学中用来衡量二分类模型精确度的一种指标。这一数值越高说明检测系统的精度越高。它同时兼顾了分类模型的准确率和召回率。F1 分数可以看作是模型准确率和召回率的一种加权平均，它的最大值是 1，最小值是 0。

$$F_1 = 2 \cdot \frac{1}{\frac{1}{\mathrm{recall}} + \frac{1}{\mathrm{precision}}} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

误判率(FPR) ＝ 误判定样本数(FP)/总负样本数（TN＋FP）

检测率(TPR)＝ 正判定检测样本数(TP)/总正样本数(TP+FN)

精确度(accuracy)=正确判定样本总数(TP+TN)/总样本数(TP+TN+FP+FN)

2.4.2 系统性能指标

本系统为应对海量数据的在线检测系统，针对数据接收和处理，定义检测时间为系统读入数据到给出准确判定结果期间的时间间隔。

第三章 作品测试与分析

3.1 测试方案

为证明我们系统的可用性以及算法的有效性我们设计了如下几个实验证明：

3.1.1 检测模型的特征工程

数据和特征决定了机器学习的上限,而模型和算法只是逼近这个上限。为了确定我们所选取的特征可以最精准、最高效地检测出恶意域名与疑似恶意的小数据集，我们对不同特征的组合进行了大量实验尝试，最终确定了我们应用到分类器当中的特征组合。

3.1.2 真实海量数据环境下的数据检测

我们的系统要解决的是真实世界中的安全检测问题，因此我们使用电信和校园网的DNS流量数据进行检测，对检测的各项指标进行了评价，证明了我们算法的有效性和在实际中应用的可行性。

3.1.3 对比验证在线学习的改进效果

本系统的创新关键点之一在于将在线学习应用到了实际并取得了良好的效果，测试中我们将没有添加在线学习模块的算法和运用在线学习的算法进行对比，发现当没有在线学习的模块时，数据流中出现了由于无法更新而在一小段时间内检测率降低的情况，加入在线学习之后这一问题得到了很好的解决。

3.1.4 针对小数据集二级标定的方法测试

对于需要二级标定的小数据集，因为这样的数据量只占到总数据的很小一部分。可以采用人工专家验证的方式，也可以采用黑名单自动化对比的方式。我们对这两种方法分别作了实验，比较检测的精度和效率。

3.2 测试环境搭建

3.2.1 Hadoop搭建

基本配置服务：hadoop用户组以及对应权限，ssh服务，java环境(openjdk-7-jdk)，hadoop2.4.0搭建：官网下载源码，解压至相应文件夹并修改权限，配置环境变量即可。

3.2.2 Spark搭建

基本配置服务：Hadoop2.4.0

Spark2.6搭建：获取安装包解压并赋予相应权限，并配置环境变量。

3.2.3 Memsql内存数据库搭建

基本配置：mysql-client 通过该客户端与memsql进行交互

Memsql14.0.35搭建：获取安装包解压至相应文件夹并运行安装脚本

3.3 测试设备

Ubuntu版本：Ubuntu 14.04.4 LTS (GNU/Linux 4.2.0-27-generic x86_64)

CPU：Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz

内存：15GB

硬盘：600GB

3.4 测试数据

我们实验数据集包括两个，一个是训练所用样本数据集，另一个是真实环境测试数据集。

本实验训练所用黑名单样本来自于以下三个网站：Malwaredomains.com、www.phishtank.com、www.malwaredomainlist.com。我们从这三个网站中筛选出部分恶意域名作为我们的训练黑名单样本，为确保黑名单的有效性，筛选出的恶意 URL 样本都经过了人工核实，最终获取的黑名单样本数量为 2317。

训练用部分黑名单样本如图 3.1 所示：

Date (UTC)	Domain	IP	Reverse Lookup	Description	Registrant	ASN	
2017/05/01_16:22	amazon-sicherheit.kunden-ueberpruefung.xyz	185.61.138.74	hosted-by.blazingfast.io.	phishing	-	49349	
2017/03/20_10:13	alegroup.info/ntnrrhst	194.87.217.87	mccfortwayne.org.	Ransom, Fake.PCN, Malspam	Lee Everton / lee_everton2002@yahoo.com	197695	
2017/03/20_10:13	fourthgate.org/Yryzvt	104.200.67.194	-	Ransom, Fake.PCN, Malspam	Charlie Dillon / goddaddy@638united.com	8100	
2017/03/20_10:13	dieutribenhkhop.com/parking/	84.200.4.125	125.0-255.4.200.84.in-addr.arpa.	Ransom, Fake.PCN, Malspam	-	31400	
2017/03/20_10:13	dieutribenhkhop.com/parking/pay/rd.php?id=10	84.200.4.125	125.0-255.4.200.84.in-addr.arpa.	Ransom, Fake.PCN, Malspam	-	31400	
2017/03/14_23:02	ssl-6582datamanager.de/	54.72.9.51	ec2-54-72-9-51.eu-west-1.compute.amazonaws.com.	redirects to Paypal phishing	goldanderwand@aol.com	16509	
2017/03/14_23:02	privatkunden.datapipe9271.com/	104.31.75.147	-	Paypal phishing	Registrar Abuse Contact abuse@namecheap.com	13335	
2017/03/06_21:09	www.hjaoopoa.top/admin.php?f=1.gif	52.207.234.89	ec2-52-207-234-89.compute-1.amazonaws.com.	Cerber ransomware	Registrant lecborbobl@rothtec.com	14618	
2017/03/06_21:09	up.mykings.pw:8888/update.txt	60.250.76.52	60-250-76-52.HINET-IP.hinet.net.	related to a Mirai windows spreader trojan	Registrant 30da1310f05f42d7a349460c551aee6f.protect@whoisguard.com	3462	
2017/03/06_21:09	down.mykings.pw:8888/ver.txt	60.250.76.52	60-250-76-52.HINET-IP.hinet.net.	related to a Mirai windows spreader trojan	Registrant 30da1310f05f42d7a349460c551aee6f.protect@whoisguard.com	3462	
2017/03/06_21:09	down.mykings.pw:8888/ups.rar	60.250.76.52	60-250-76-52.HINET-IP.hinet.net.	related to a Mirai windows spreader trojan	Registrant 30da1310f05f42d7a349460c551aee6f.protect@whoisguard.com	3462	

图 3.1 训练用黑名单样本示例

本实验训练所用的白名单样本来自于 alexa 访问排名前十万的域名，我们从中筛选出 2834 条域名构成训练所用的白名单样本。正负样本的比例接近 1：1。

本实验真实环境测试数据集中的所有数据来自于电信网络一周内的流量数据和大学校园网一周内的 DNS 服务器流量数据，在云平台上以分钟为单位划分时间片并行处理，检测模型每小时更新一次。

3.5 结果分析

3.5.1 特征工程

在我们的实验中应用了两种 SVM 模型，第一种是二类 SVM 分类器，用于检测恶意域名、第二种是一类分类器 One-class SVM，用于筛选易被误检的小数据集。

1、SVM 检测模型

为了验证我们SVM分类模型的检测效果，我们在数据集上采取了十折交叉验证的方法，将2800条白名单与2400条黑名单数据分为十份，轮流将其中的九分作为训练数据，一份作为测试数据进行试验。求取十次结果的指标的平均值作为最终的指标。

如 2.2.2 特征抽取部分所示，我们选取了共 21 维特征用于机器学习模型训练。这一部分我们进行了 24 次尝试，分别为单纯词法特征，单纯网络特征，去除某一项单一特征共 21 种，和综合特征。得到了准确率结果如下。

我们分为单纯词法特征，单纯网络特征，两种特征综合三种情况进行测试：

在单纯词法特征下，由于未考虑到域名的网络属性例如A记录个数，NS个数，TTL平均值等，会使大量伪装成合法域名的恶意域名成为漏网之鱼，同理单纯网络特征检测会导致忽视恶意域名本身特征，从实验结果中也不难发现进行特征综合检测之后，正确率均有大幅提高。

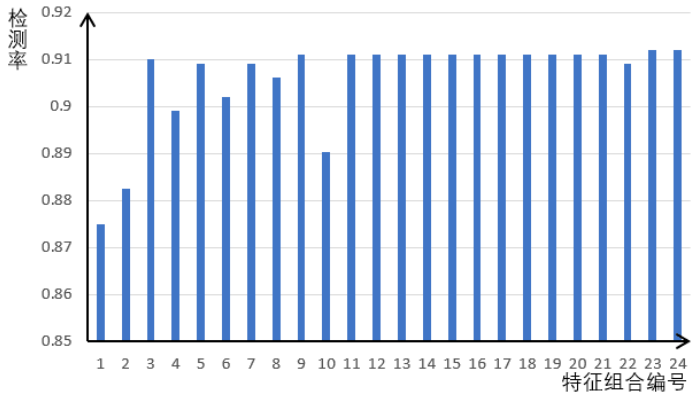

图3.2 SVM模型检测率变化

机器学习模型	检测率(Detection Rate)		
	词法特征	网络特征	综合特征
半监督学习	82.5%	79.7%	84.6%
基于23维特征的SVM机器学习	93.3%	88.1%	95.3%

表1 检测模型的精确度对比

半监督学习模型无法进行模型更新，并且使用半监督进行聚类会有大量数据被归为可疑类，无论是对人工标定还是基于黑白名单验证都增加了许多工作量。因此，只有使用基于SVM/ONE-CLASS SVM的在线主动学习，采用两个one-class模型进行交叉验证，生成小数据集进行主动学习标定并更新模型。

2、十折交叉训练

由于精确度并不能反映模型的综合性能，因此我们对基于SVM/ONE-CLASS SVM的在线主动学习模型做了进一步的测试，使用样本数据集中2800条良性域名和2400条恶意域名，结果如下表所示：

测试集序号	实际良性域名数量	实际恶意域名数量	检出良性域名数量	检出恶意域名数量	检出实际恶意域名数量	准确率	召回率	F1-score
1	283	231	282	232	230	99.14%	99.57%	99.35%
2	283	231	279	235	230	97.87%	99.57%	98.71%
3	283	231	282	232	230	99.14%	99.57%	99.35%
4	283	231	285	229	228	99.56%	98.70%	99.13%
5	283	231	278	236	230	97.46%	99.57%	98.50%
6	283	231	283	231	229	99.13%	99.13%	99.13%
7	283	231	280	234	229	97.86%	99.13%	98.49%
8	283	231	279	235	229	97.45%	99.13%	98.28%
9	283	231	286	228	227	99.56%	98.27%	98.91%
10	287	238	285	240	236	98.33%	99.16%	98.74%

表2 十折交叉训练结果

	准确率	召回率	F1-score
基于23维特征的SVM机器学习	95.3%	95.6%	95.5%
半监督学习	84.6%	88.5%	86.5%

表3 训练集平均检测指标

由表中可以看出，我们所使用的模型准确率提升了10.7%，召回率提升了7.1%，F1-Score提升了9%，说明我们的模型相比较于半监督学习综合性能有很大提升。

3.5.2 真实数据环境检测效果

1、真实环境检测率测试

我们收集了校园网DNS服务器三小时内DNS流量以及电信1天的数据流量，利用tcpreplay模拟真实环境进行测试，并使用Passive DNS收集数据，经过去重和白名单过滤之后筛选出615843437条数据进行分析。

	测试结果
样本总数	615843437
检出恶意	18384
误报数	1139

表3 真实数据检测结果

在真实环境中，我们无从获取所有域名的正确标定结果，而只能对检出的恶意域名进行标定。经过我们的二次检测，检出的18384条恶意域名中有1139条属于误报，正确率达到了93.8%。

2、运行速度测试

运行速度结果对比：

普通服务器单域名检测时间平均为1.36214613914s

基于Spark/Hadoop采用并行化技术，平均每100条约用时1.672441655s

3.5.3 在线学习的改进效果

在这一部分中我们将采用了在线学习模块的算法和未采用在线学习模块的算法进行对比。实验结果表明在未使用在线学习算法时，我们的检测模型出现长时间内的检测率降低，如图3.2。

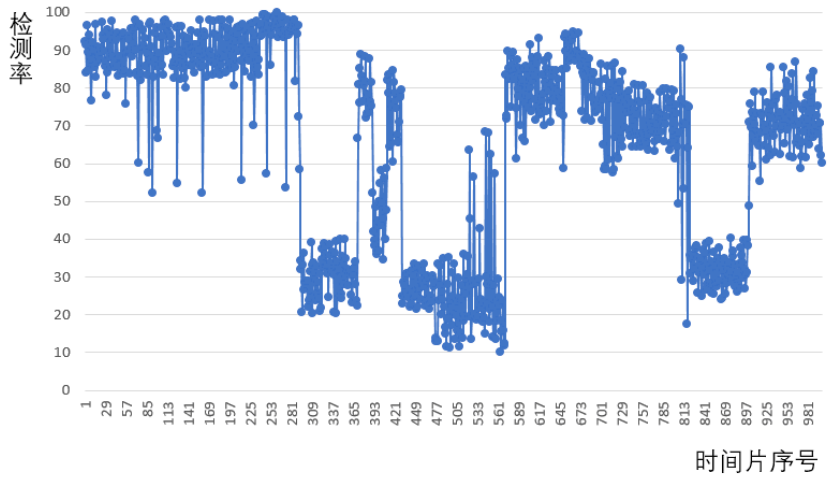

图3.4 未使用在线学习的检测效果

对其中第一个较低的检测率低谷放大之后如图3.3。可见如果我们在第285-300时间片之间能够发现这个无法检测的恶意域名并加以训练检测，会很好的解决这一低谷

中的数据误差。

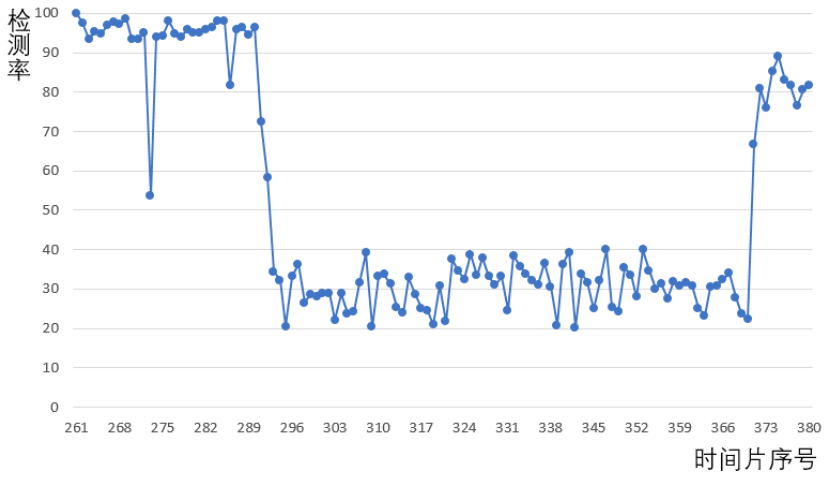

图3.5 一段时间片内的检测率下降

实际应用在线学习的算法也较好地解决了这一问题。见图3.3，原先长段的低谷被截短，实现了及时的发现与检测。

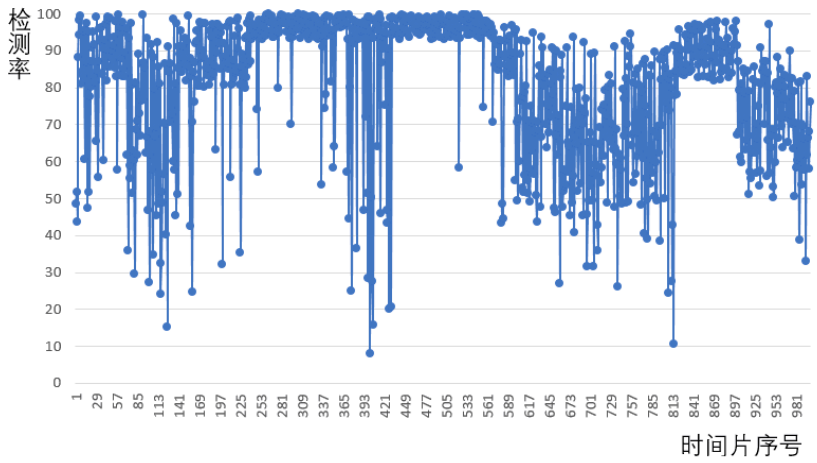

图3.6 在线学习算法的检测率

3.5.4 小数据集的效率提升

经过实验数据的检测和模型的应用，我们确定应用我们的算法所检测出的易误判的数据集可以很好的降低数据规模，从图3.4中可以看出，每个时间片中应用双模型所检测出的边界数据集只占到总数据规模的1%-5%，这样得到的小数据集用于模型更新再训练，可以大大提高效率。

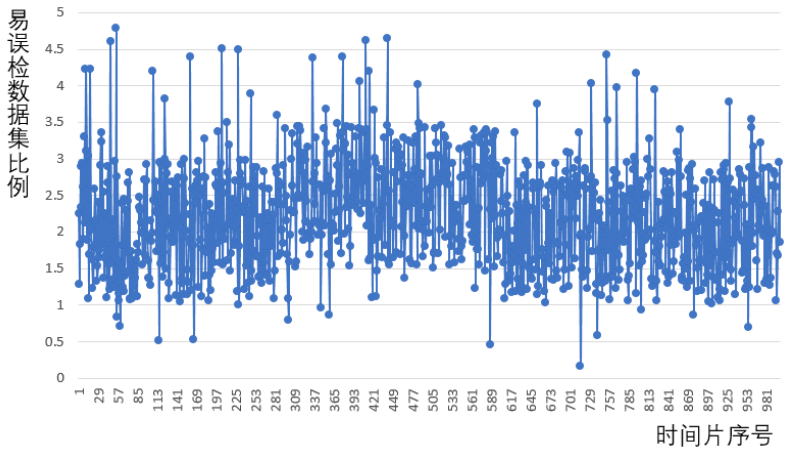

图3.7 易误检数据集所占总数据集比例

3.5.5 小数据集人工检测效果

在这部分实验中，我们在3.5.3中的时间片序列中，选取了50个时间片中出现明显误检数据进行对比试验以检验小数据集的更新效果。一组使用人工对小数据集进行标定，一组使用了五个黑名单进行自动化标定，得到下一时间片更新后模型的检测率，图示如下：

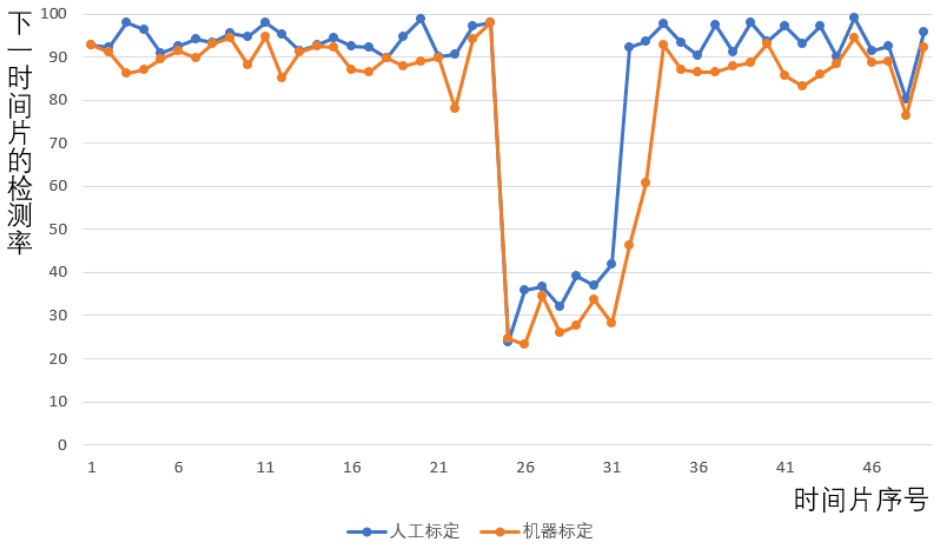

图3.8 小数据集标定对比效果

从图中可以分析得到，人工标定的效果要略高于机器标定，但使用综合自动标定算法的方式同样做到了在线学习的效果，最终我们决定在真实数据中采用综合自动化标定的方式来标定小数据集。

第四章 创新性说明

本系统实现了适用于大规模数据的基于自反馈学习的恶意域名实时监测系统。其创新点有三：

一、海量数据实时处理

一个大型企业每天多达上亿次的域名解析请求，会产生上千万条各不相同的域名记录。这已经远远超出了人工检测的能力范围，引入机器学习的检测可以较好减轻认理负担。但更多的，如何准确、及时地检测出这些域名中的威胁是我们小组关心的重点。

我们系统采用 Hadoop+Spark 的云端处理平台处理流模式化的 DNS 解析数据，提取了适用于日志流分析的特征来训练模型，经测试对大规模数据也有良好的处理效率与准确率。

二、在线学习

在我们的算法中解决了一个棘手的实际问题：当一个恶意域名刚刚出现，且只持续在几个小时或几天的时间内，市面上的黑名单还没有来得及将其加入到其中的时候，我们如何让我们的检测模型将其发现并及时的做出相应？

在线学习的算法在近几年被提出，目的在于使训练的检测模型得到及时的更新以面对多变的情景，我们的系统成功地将在线学习算法应用到实践之中，并取得了良好的效果。

三、抽取小数据集验证的创新

我们是一个工作在数据流上的系统，在检测的过程中并不能知道待检域名的准确标签值（即是否为恶意）。在新的时间片结束时，使用全部数据进行标定来更新模型工程巨大，不可能使用人工标定。

因此我们提出了只抽取其中占 1%~5%的小数据集进行精确地验证，而这一部分小数据集是我们原有固定检测模型所可能产生误判的部分。并且通过这部分小代价的标定，最终实现了检测率的提高和在线学习模型的更新。

第五章 总结

近年来恶意域名在各种攻击中起到了重要的作用，攻击者也尝试使用诸如DGA和Fast-Flux等技术躲避追踪。针对这样的安全趋势，本小组开发了适用于大规模数据的在线恶意域名检测系统。旨在及时发现海量数据中可能存在的恶意域名并对背后可能的攻击行为作出响应。

经过算法论证和实验测试，本作品通过在线学习模块在现有黑白名单的基础上提高了检出率，大大提高了实用性。且创新性地提出使用小数据集进行验证的办法，降低了标定成本，实现了自动化更新检测。相较于传统的关注URL的检测而言，数据规模更小更精，部署更加广泛。

基于以上优势，我们认为这样的大数据在线检测系统将会在未来的企业防御中起到重要的作用，对恶意域名相关的攻击实时检测与防御有着积极意义。

参考文献

[1] Anton Dan Gabriel et al. Detecting malicious URLs. A semi-supervised machine learning system approach. *18th International Symposium on Symbolic and Numeric Algotrithms for Scientific Computing* 2016

[2] L. Bilge et al., EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis *Network and Distributed System Security Symposium*, February 2011

[3] Hongyu Gao et al., An Empirical Reexamination of Gloabal DNS Behavior, *ACM Special Interest Group on Data Communication* 2013.

[4] Andrew B. Goldberg et al., OASIS Online Active SemI-Supervised Learning, *Association for the Advancement of Artificial Intelligence* 2011

[5] Justin Ma et al., Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs , *Acm Sigkdd International Conference on Knowledge Discovery & Data Mining* 2009

[6] Brad Miller et al., Reviewer Integration and Performance Measurement for Malware Detection Miller *Detection of Intrusions and Malware, and Vulnerability Assessment.* 2015.

[7] Malware Domain Block List. http://www.malwaredomains.com/.

[8] PhishTank http://www.phishtank.com/

[9] One-class SVM with non-linear kernel http://scikit-learn.org/stable/modules/svm.html

[10] Suyeon Yoo et al., Two-Phase Malicious Web Page Detection Scheme Using Misuse and Anomaly Detection *International Journal of Reliable Information and Assurance* 2014

[11] Doyen Sahoo et al. Malicious URL Detection using Machine Learning:A Survey *arXiv.org:1701.07179v2 [cs.LG]16* Mar 2017

[12] Ryan Mcdonald et al. Online Large-Margin Training of Dependency Parsers *Meeting on Association for Computational Linguistics* 2005

[13] Dengyong Zhou et al. Learning with Local and Global Consistency *International Conference on Neural Information Processing Systems. MIT Press,* 2003

[14] 张洋, 柳厅文, 沙泓州,等. 基于多元属性特征的恶意域名检测[J]. 计算机应用,

2016, 36(4)

[15] 马旸, 强小辉, 蔡冰,等. 大规模网络中基于集成学习的恶意域名检测[J]. 计算机工程, 2016, 42(11)

[16] 康乐, 李东, 余翔湛. 基于SVM的Fast—flux僵尸网络检测技术研究[J]. 智能计算机与应用, 2011, 01(3)

[17] M. Konte et al. Dynamics of Online Scam Hosting Infrastructure *International Conference on Passive and Active Network Measurement. Springer-Verlag,* 2009

[18] S. Hao, N et al. Monitoring the initial DNS behavior of malicious domains. *ACM SIGCOMM Internet Measurement Conference* 2011.

[19] J. S. Otto et al. Content delivery and the natural evolution of DNS: remote dns trends, performance issues and alternative solutions. *ACM SIGCOMM Internet Measurement Conference* 2012.

[20] R. Perdisci et al. Detecting malicious flux service networks through passive analysis of recursive DNS traces. *Annual Computer Security Applications Conference*, 2009.

[21] S. Yadav et al. Winning with DNS failures: Strategies for faster botnet detection. *SecureComm,* 2011.