大规模 DGA 域名检测方法研究

司俊俊[2]，刘丙双[1]，戴帅夫[1,*]，张建宇[2]
CNCERT/CC[1]
长安通信科技有限责任公司[2]

摘要：僵尸网络通常使用域名生成算法（Domain Generation Algorithm, DGA）生成大量域名以隐藏其命令与控制（C&C，Command and Control）通信，这类 DGA 域名也常见于域名阴影和域名散列攻击。因此，对 DGA 域名的识别工作非常必要。本文通过设计高效的域名特征，利用机器学习算法对省级海量 DNS 访问日志进行分析以检测 DGA 域名。本方法目前可识别 19 类已知 DGA 域名，并支持发现新的 DGA 类型，从而能够实现对已知和新型僵尸网络的监测和发现。
关键字—机器学习，域名安全，DGA，DNS

LARGE SCALE DGA-BASED DOMAIN DETECTION

Junjun Si[2]，Bingshuang Liu[1]，Shuaifu Dai[1,*]，Jianyu Zhang[2]
CNCERT/CC[1]
CHANCT, Inc.[2]

Abstract: Domain Generation algorithm (DGA) is used in botnets to conceal the command and control while generating large scale domains. Those DGA domains are also widely seen in domain shadowing and DNS based non-exist domain flood attacks. Therefore, it's import to detect DGA domains from massive DNS records. This paper proposes a solution which uses machine learning techniques to detect new DGA types besides 19 known DGAs. From those detected DGA domains we can provide threat intelligence information for further security analytics.
Index Terms—Machine Learning, Domain Security, DGA, DNS

1 引言

僵尸网络是由被感染僵尸程序的主机组成的一个可控网络。攻击者通过命令和控制信道(C&C，Command and Control)对僵尸主机发送指令，从而进行信息窃取、拒绝服务攻击等网络攻击和犯罪。自上世纪九十年代末出现伊始，僵尸网络结构和形态从最初简单的集中式 C&C 发展到基于 P2P 的分布式 C&C，所使用的域名则从最初的固定域名演变为域名迁移（Domain Flux）[8]。目前，基于 DGA（Domain Generation Algorithm）[1]的僵尸网络变得越来越流行。感染 DGA 恶意代码的主机周期性生成大量的域名，并对其进行 DNS 查询，攻击者只注册使用其中一小部分域名，并周期性地变化。因此，无法利用传统的黑名单拦截等手段对其进行防御。

除僵尸网络外，DGA 域名还常被用于域名阴影攻击（Domain Shadowing）[9]。攻击者窃取受害者（比如网站站长）的域名账户后，创建大量子域名，然后利用这些子域名进行网络钓鱼等恶意行为。臭名昭著的钓鱼工具包 Angler Exploit Kit 在 2015 年即采用了这种域名阴影的新技术。由于被盗取的域名通常是合法域名，防御者无法知道攻击者下一个目标是什么，且攻击者会创建大量子域名，使用较短时间后便放弃，黑名单防御方法同样失效。

*通信作者：dsf@cert.org.cn

DGA 还被应用于基于 DNS 的域名散列攻击（Non-Exist Domain Flood Attack）[10]。攻击者通过 DGA 算法构造大量未经注册的随机子域名并发起 DNS 查询，本地缓存找不到域名解析记录时，DNS 服务器就会向上级 DNS 服务器进行递归查询，直至权威域名服务器。与传统 DNS 查询泛洪攻击类似，当查询量很大时，DNS 服务器便无法及时响应合法的 DNS 查询请求。由于随机构造的未注册域名无法在中间域名服务器上命中缓存，从而使得 DNS 查询压力最终集中到权威域名服务器，因此这种基于 NXDomain 的 DDoS 攻击往往会造成更严重的后果和损失。

通过对 DGA 算法进行逆向从而提前获知域名列表是一种可能的防御方法，但需要专业的逆向人员，且难度很大。同时由于一部分 DGA 域名算法种子的不确定性，针对这一部分 DGA 根本无法提前生成域名列表。

本文提出一种基于机器学习的 DGA 检测算法，并应用于某省运营商 DNS 访问记录。实验表明，该方法能够有效检测已知和未知 DGA 域名，从而为进一步打击僵尸网络安全事件提供有力线索。本文第二章介绍研究背景与相关工作，第三章详细描述 DGA 检测算法流程，第四章给出实验结果，最后对本文工作进行总结。

2 研究背景与相关工作

DGA 分类 根据域名生成方式，DGA 可分为四大类：一是 TID(Time Independent and Deterministic)，也就是种子确定，且不依赖于时间，该类 DGA 产生确定的域名，如 Kraken 家族。这类 DGA 可以直接用黑名单进行过滤。TID 类 DGA 数量较多，约占总量 40%；第二类是 TDD（Time Dependent and Deterministic），也就是种子确定，但产生的域名会随着 DGA 运行时间的变化而不同，如 Conficker。此类 DGA 最多，约占总量的 55%；三是 TDD（Time Dependent and Non-deterministic），即种子不确定，且随时间不同而产生不同域名，目前发现的样本只有 Bedep 和 Torpig；最后一类是 TND（Time Independent and Non-deterministic），即不依赖时间，但种子不确定的 DGA，目前尚未发现有此类样本。

DGA 检测方法 代码逆向是人们认识和研究 DGA 的一种重要手段。Johannes Bader[2]一直致力于该工作并成功逆向出多种 DGA 及其变种，如 Pykspa、Murofet 等。从其工作中可以发现，很多 DGA 可能有许多不同的种子，每个种子每天能产生几万个域名，而攻击者只使用其中一部分。因此，即便是逆向后的 DGA，也很难预先产生完备的拦截名单。相应地，学术和工业界愈来愈侧重如何智能地检测和识别 DGA 域名。Damballa 公司的 Manos Antonakakis[3]等人提出了一种从 DNS 访问记录中的不存在域名（Non-exist Domain, NXDomain）入手，提取域名特征，利用机器学习算法检测恶意 DGA 域名的方法。他们的做法是，事先利用已知 DGA 样本生成每类 DGA 的域名，把若干个域名分为一组（例如 10 个），提取一组（32 个）域名特征形成特征向量，利用分类和聚类算法判断未知域名是否合法，或属于某类 DGA。该系统部署于北美某 ISP 网络中，能够成功检测出基于 DGA 的恶意攻击。但也存在以下局限：一是该系统并不支持大流量环境下的大数据处理能力，无法应对海量 DNS 日志；二是该系统对一组域名提取一个特征向量进行训练和分类，容易产生较高的漏报和误报；三是基于 HMM 的 C&C 识别效果欠佳。本文提出基于机器学习的 DGA 域名检测方法，有效解决了以上三个问题，并实际用于某省海量域名访问日志分析，识别大量 DGA 域名，进一步通过人工分析发现新的僵尸网络家族、域名阴影事件以及针对域名服务器的 DDoS 攻击事件，为基于域名的安全事件分析提供了有效的线索。

3 系统流程与实现

本文提出的 DGA 域名检测算法流程如图 1 所示。在对 DNS 记录进行预处理后，抽取不存在域名，使用 Spark-MLlib 的聚类算法，对域名进行聚类，发现其中的 DGA 域名。进一步利用黑白名单离线训练域名分类器，通过线下训练、线上分类对检测到的 DGA 域名进行类型标定，即 DGA 分类。在此基础上，根据 DGA 识别结果发现被感染的僵尸主机，利用已识别的 DGA 域名在线训练针对该 DGA 的域名分类器，以检测疑似 DGA C&C 域名。

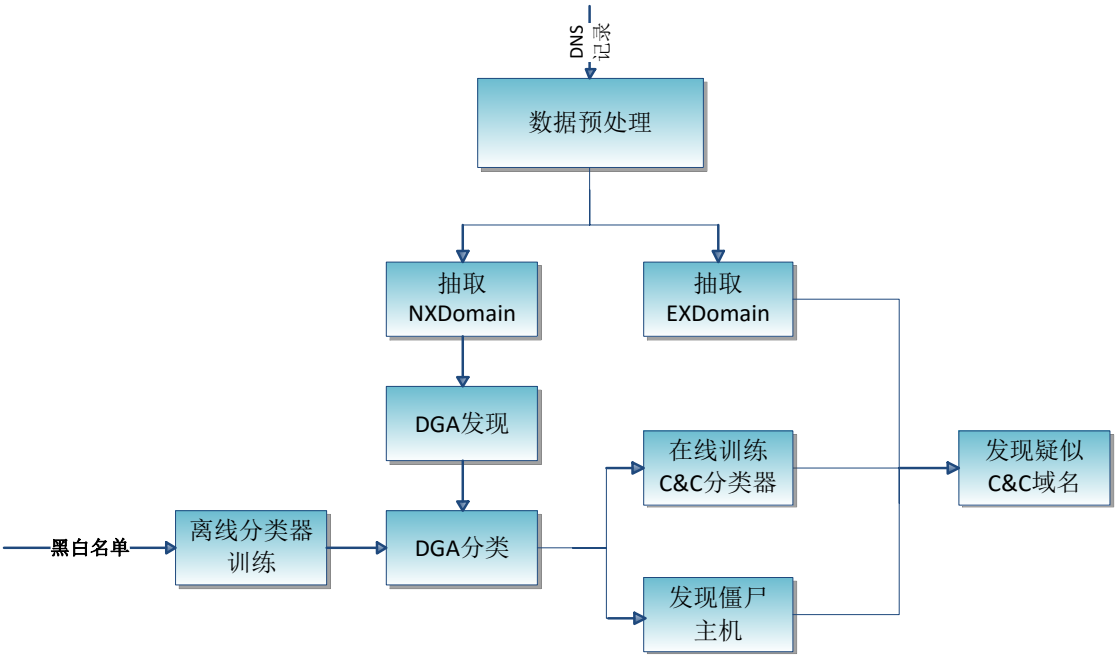

图 1:基于机器学习的 DGA 域名检测系统流程

3.1 数据预处理

数据预处理是为了降低计算量而对原始 DNS 记录进行的数据去重、相关字段抽取、黑白名单过滤、不活跃主机和域名过滤等操作。

抽取 NXDomain 基于 DGA 的僵尸网络每天会生成大量的域名并对其进行 DNS 查询，而其中大部分域名是未注册的不存在域名。根据某省一天的 DNS 访问日志统计结果，独立的 NXDomain 域名数量约占常规域名的七分之一。此外，DGA 域名通常具有字符随机、易读性差、域名较长等区别于正常域名的特点。因此，从 DNS 访问日志中的 NXDomain 域名入手提取域名特征，利用聚类和分类算法识别其中的异常域名一方面能有效降低计算量，另一方面也充分利用了 DGA 僵尸网络的特点，即单个僵尸会访问大量无效域名。在实际中我们发现一些 NXDomain 域名的 DNS 应答记录可能会被改写，重新填入一些 IP 地址。这些 IP 通常是带有广告性质。因此，本系统建立了一个广告类 IP 库，并定时更新，依赖此 IP 库把 DNS 访问记录中被改写的 NXDomain 查找出来。

3.2 DGA 发现

DGA 发现模块针对 NXDomain 的 DNS 访问记录进行分析，以发现 DGA 恶意域名。该模块包含域名特征提取、基于域名特征的聚类、基于主机-域名访问关系的聚类以及交叉去噪四个部分。从 NXDomains 发现 DGA 域名集的过程如图 2 所示。

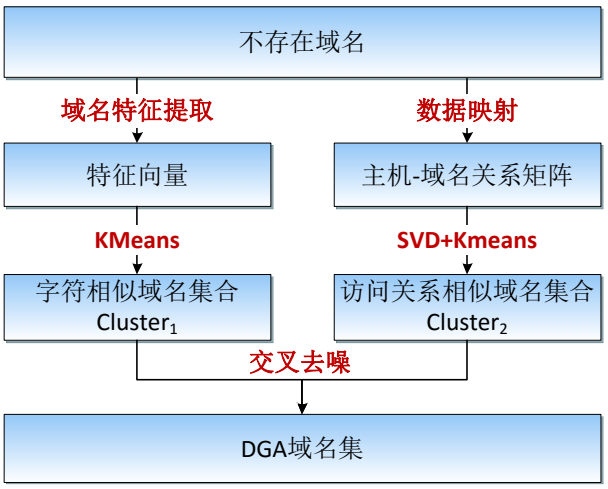

图 2: DGA 域名发现流程

域名特征提取 域名特征提取是本系统的一个重要阶段，通过设计一组高效的特征变量把字符串形式的域名转换为可计算的值向量，使得具有相似字符特性的域名被归为一类。DGA 域名通常具有随机性、可读性较差等和常规域名不同的特性，基于这些先验知识，本系统对每个域名提取 16 个字符特征，包括 2/3-gram 子串出现频次的中值、均值和方差，连续重复数字/字母个数，元音字母/辅音字母/数字字符个数，去重后的字符个数，域名长度，域名/二级域名信息熵，以及顶级域名是否是常见顶级域名（com/cn/net/org/com.cn/gov.cn/hk 等）。

基于域名特征的聚类 提取域名特征后，域名转换为特征向量，利用 Spark-MLlib 中的 KMeans 聚类算法对特征向量进行聚类，进一步通过数据关联得到域名类 $Cluster_1(D_1,D_2,\ldots,D_p)$，其中每个子集是具有相似字符特性的一类域名。

基于主机-域名访问关系的聚类 根据 DNS 记录中源 IP（SIP）和域名访问关系可以建立一个二维矩阵 M，行表示 SIP（m 个），列表示域名（n 个）。如果 sip_0 访问过 k 个域名，则矩阵 M 对应 sip_0 及其访问的 k 个域名位置的值应设为 $1/k$。这里需要注意的是，M 是一个庞大且稀疏的矩阵，应以稀疏矩阵进行存储和计算以降低资源开销。根据 DNS 访问记录设置好矩阵 M 后，首先对矩阵 M 进行归一化处理：

$$\sum_{i}^{m} M_{i,j} = 1 \qquad (1)$$

然后利用矩阵的奇异值分解算法[6]求取矩阵 M 的特征根和特征向量，即对矩阵 M 求一个分解，使得：

$$M = U * S * V^{'} \qquad (2)$$

其中，S 是 $m*n$ 的矩阵，其对角线元素是矩阵 M 的特征值；U 是一个 $m*m$ 维的特征向量矩阵，这些向量是 $M*M^{'}$ 的特征向量，$M^{'}$ 是 M 的共轭矩阵；V 是一个 $n*n$ 维的特征向量矩阵，这些向

量是 $M^{'}*M$ 的特征向量。由于最终是要对域名进行聚类，因此需选取矩阵 V 中的特征向量。之后利用 KMeans 算法对特征向量进行聚类并映射得到域名的集合 $Cluster_2(D_1^{'},D_2^{'},...,D_q^{'})$，即被相同主机群访问的域名被聚为一类。

交叉去噪 对域名集合 $Cluster_1$ 和 $Cluster_2$ 求笛卡尔集，得到 $Cluster_3(D_1 \cap D_1^{'}, D_1 \cap D_2^{'},..., D_i \cap D_j^{'},..., D_p \cap D_q^{'})$。$Cluster_3$ 中每个子集是 $Cluster_1$ 和 $Cluster_2$ 中任意两个子集的交集。这样做的原因是一个 DGA 病毒被一些主机感染后，在这些主机上产生相同的 NXDomains，也就是说，这些 NXDomains 即具有字符相似性，又具有相同的访问者。因此，认为 $Cluster_3$ 中的每个域名类即是一个 DGA 域名类。

3.3 DGA 分类标定

发现 DGA 域名后，进一步通过机器学习算法可对发现的 DGA 域名进行类型标定，即进行 DGA 分类标定。

黑白名单 本系统使用 Alexa 网站排名中国访问量 Top 1 万的域名[7]作为白名单，并借助 Johannes Bader 的 DGA 逆向工作以及运行病毒样本等方式获取了 Conficker、Pykspa、cryptolocker、gameoverP2P、kraken、necurs、Murofe 等十类 DGA 域名样本作为黑名单。

DGA 分类器 包含正反样例，本系统需要训练一个能够进行十一种域名分类的 multi-class 分类器。经过实验对比，本系统选取了 Spark-MLlib 中的 RandomForest 分类算法进行 DGA 分类器训练，分类器整体召回率和准确率达到 86%（某些 DGA 产生的域名具有一定的特征相似性，甚至两个不同的 DGA 可能会产生相同的域名，如 Conficker 和 Pykspa2[4]，这在一定程度上影响了 DGA 分类器的整体性能）。DGA 分类器训练离线完成。在获取 DGA 域名类后，再调用离线训练好的 DGA 分类器模型进行 DGA 域名在线分类。

C&C 分类器 得到恶意 NXDomain 域名的 DGA 分类后，通过数据关联，即可发现被该 DGA 病毒感染的主机群。根据被感染主机的 DNS 访问记录，本系统可以检测该 DGA 的疑似通信控制（C&C）域名。除了早期的一些 DGA 使用固定的域名作为通信控制域名，越来越多的 DGA 病毒的通信控制域名通常也是由该 DGA 算法来生成的，即通信控制域名和 NXDomains 具有相同的字符特性。为了提高通信控制域名的识别率，本系统利用 3.2 节中获取的该 DGA NXDomains 类作为黑名单，在线训练针对该 DGA 的二类域名分类器，该分类器识别率达到 95% 以上。然后利用该分类器对被感染主机的 DNS 访问记录中正常解析的域名进行分类，得到疑似 C&C 域名。

3.4 大数据处理引擎 Spark

根据本文统计，省级 DNS 访问日志即使按十比一抽样后，每天记录数仍平均高达 17 亿条。为了能够处理海量 DNS 日志，基于 Spark 实现了本文提出的 DGA 域名检测方法。Spark 是一个应用广泛的开源大数据处理引擎，比 Hadoop MapReduce 计算效率高 100 倍以上，具有 Spark-SQL、Spark-mllib、Spark-streaming、Spark-graphx 四大组件，因此具有很好的通用性。Spark 以 RDD(Resilient Distributed Datase)[5]形式组织数据，对数据的操作分为两类，其一是 Transform，如 map、filter，其二是 Action，如 reduce、saveAsTextFile。Transform 操作是 lazy 的，只有当 action 发生时，相关的 transoform 操作才会执行。因此，对多次用到的 RDD 进行缓存能有效提高计算效率。

4 实验验证

根据 Plohmann 在 Botconf2015 上的报告[4]，目前已发现 40 种 DGA，经作者查证部分 DGA 脚本相同，如 Shifu 和 Simba、Gameover P2P 和 Gameover NewGoz 的代码一样，另外一部分 DGA 如 Qakbot 等无法获取代码或样本。最后经过筛选，本文对 Hesperbot、Fobber、Simda 等九类 TID 类型的 DGA 采用黑名单过滤的方式进行检测，对 Conficker 家族、Murofet 家族、Pykspa 家族等十类 TDD 类型的 DGA 运用机器学习的方法进行检测。

本文提出的 DGA 域名检测系统部署在某省大数据服务器集群上，以离线计算的形式，每天根据该省前一天的 DNS 访问记录来检测 DGA 域名。考虑到存储能力，该省的 DNS 访问记录是十比一抽样后的。抽样后一天的 DNS 访问记录约 17 亿条（如图 3 所示），按源 IP 和域名去重后约为 5 亿条，独立主机数量约为 300 万台，独立被访问域名数量约为 2000 万个，其中 NXDomain 约为 300 万个。本系统利用 6 台大数据计算节点定时对一天的数据进行汇总与分析。

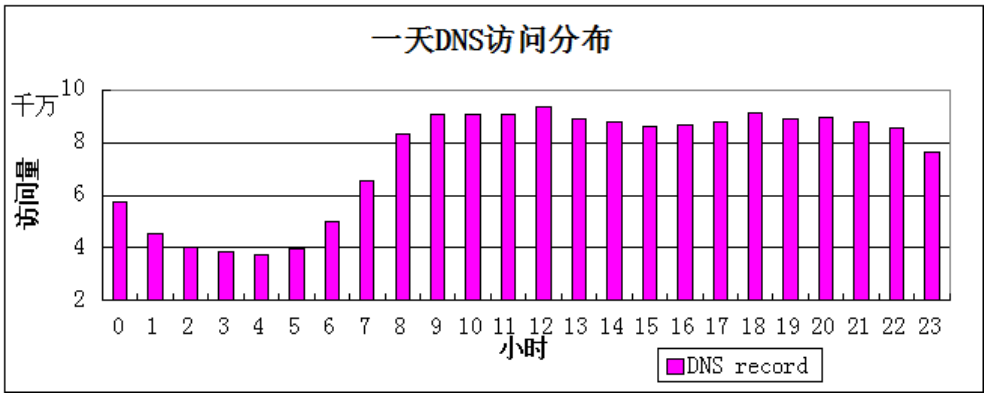

图 3: 某省一天的 DNS 访问分布

表 1 为系统检测出的属于 Conficker DGA 产生的域名及访问域名的主机数量。从字符特性上，这些域名符合 Conficker DGA 二级域名长度在 4 到 11 之间、字符随机、常用 org/cc/com/biz 等顶级域名的特点。

表 1：2016-03-27 日检测出的部分 Conficker 域名（NXDomain）

域名	DGA 类型	访问该域名的主机数量
lkocqhrb.biz	Conficker	77
pldteqjp.com	Conficker	82
kesfzrlf.biz	Conficker	90
rrinykmdfa.biz	Conficker	151
rqkgsinizs.biz	Conficker	143
rbuwryql.cc	Conficker	69
itfmrwsajs.biz	Conficker	72
vsojnfzaq.cc	Conficker	140
itfmrwsajs.biz	Conficker	72

表 2 所示为系统检测出的表 1 中 Conficker DGA 的 C&C 域名。对这些域名进行分析发现，其中有些域名已经在公共黑名单中出现过，或在网页发现有恶意样本。查询 whois 信息后，对这

些域名的注册者或注册邮箱进行 whois 反查也发现有关联的恶意域名，且许多相关域名生存期很短，十分可疑。

表 2：2016-03-27 日检测出的部分 C&C 域名

域名	DGA 类型	C&C	访问该域名的主机数量
czwnd.com	Conficker	是	229
drxmn.com	Conficker	是	214
jkmhn.com	Conficker	是	284
qzbrd.com	Conficker	是	184
ythcn.com	Conficker	是	759

表 3 中的域名是本系统发现的部分新 DGA 域名，这些域名与常规 DGA 不同的地方在于，随机变化的部分是子域名，二级域名保持不变，且二级域名通常是合法的域名。图 4 所示为 **.1st.attackd9.m.cdn30.com(记为 D1)* 域名访问量分时分布，异于常规 DNS 查询，这些域名在凌晨 3 至 6 时有很高的访问量。对这类域名，我们通过调研分析，认为可能与域名阴影或针对域名服务器的 DDoS 攻击有关。

表 3：2016-03-27 日检测出的部分新 DGA 域名

域名	DGA 类型	访问该域名的主机数量
qh.s.ysyese.cn	newDGA	199
wd.s.ysyese.cn	newDGA	187
cnmr.s.ysyese.cn	newDGA	8
qt.www.138wq.com	newDGA	206
erif.www.138wq.com	newDGA	16
mxwj.www.138wq.com	newDGA	18
a.1st.attackd9.m.cdn30.com	newDGA	373
nbcqefguijxlz.1st.attackd9.m.cdn30.com	newDGA	103
aocdrsguijkym.1st.attackd9.m.cdn30.com	newDGA	95

目前该系统已稳定运行半年，能有效检测 DGA 域名。在运行期内，每天都能检测到数万个类似的异常域名，被窃取的二级域名则通常频繁变化，有的只出现一天，有的则会连续出现，甚至有些被窃取的二级域名在被攻击者利用一次后，几个月以后又被重新启用。由于目前本系统只有域名访问信息，无法获取相关目标后续访问流量，因此无法确认具体攻击事件是基于域名阴影的网络钓鱼攻击，还是针对域名服务器的 DDoS 攻击。需要进一步结合主动探测手段或流监测记录进行深入关联分析。

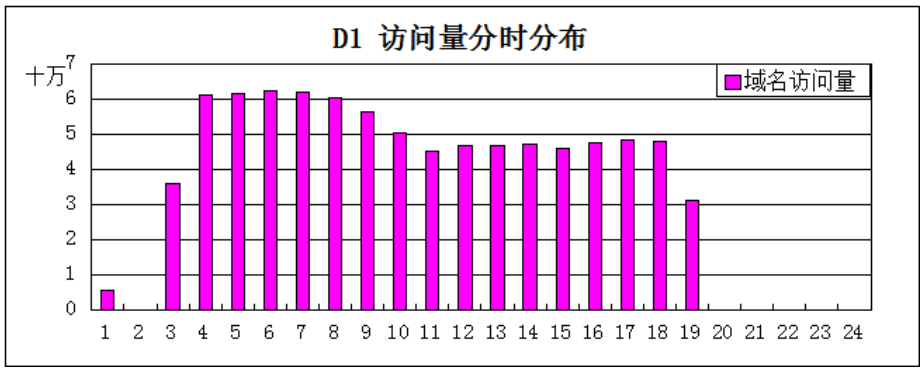

图 4: *.1st.attackd9.m.cdn30.com(D1) 访问量分时分布

5 讨论与结论

基于机器学习的检测方法高效快捷，然而也面临一些问题，比如大量的合法域名采用数字或拼音首字母缩写等方式，从而会产生和 DGA 域名相似的特征，增加系统误报率。并且由于域名阴影、散列攻击等攻击形式的存在，不能单纯把其作为白名单过滤掉。此外，该方法对于基于单词库的 DGA（如 Matsnu）识别，也有很大检测难度。

本文提出了一种基于机器学习的 DGA 域名检测方法，通过设计一组有效的域名特征向量使得每个域名转换为可计算的值向量，利用 KMeans 及基于矩阵奇异值分解的聚类算法对域名进行聚类，同时利用黑白名单离线训练 DGA 分类器，再通过在线预测进行 DGA 分类。在此基础上，进一步发现和监测僵尸网络及其疑似 C&C 域名。本文基于 Spark 平台实现对某省海量 DNS 访问日志的分析，可以有效检测出 DNS 访问记录中的 DGA 域名，从而为发现僵尸网络 C&C 通信、域名阴影以及域名散列攻击等恶意行为提供有力线索。

参考文献

[1] https://en.wikipedia.org/wiki/Domain_generation_algorithm
[2] http://johannesbader.ch/tag/dga/
[3] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon. From throw-away traffic to bots: Detecting the rise of dga-based malware. In Proc. of the 21th USENIX Security Symposium (Security'12), Bellevue, Washington, USA, pages 48–61. USENIX Association, August 2012.
[4] D. Plohmann, DGArchive A deep dive into domain generating malware, Paris, France, December 2015.
[5] M. Zaharia et al. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In NSDI, 2012.
[6] G. H. Golub and C. Reinsch, Singular value decomposition and least squares solutions, Numer. Math., vol. 14, pp. 403-420, 1970.
[7] http://www.alexa.com/topsites/countries/CN
[8] https://de.wikipedia.org/wiki/Domain_Flux
[9] http://blogs.cisco.com/security/talos/angler-domain-shadowing
[10] https://community.infoblox.com/t5/Company-Blog/Understanding-NXDOMAIN-Attack-Methods/ba-p/3949
[11] P. Royal. Analysis of the kraken botnet. http://www.damballa.com/downloads/pubs/Kraken Whitepaper .pdf, April 2008.
[12] S. Yadav, A. Reddy, A. Reddy, and S. Ranja. Detecting algorithmically generated malicious domain names. In Proc. of the 10th ACM SIGCOMM Conference on Internet Measurement (IMC'10), Melbourne, Australia, pages 48–61. ACM, November 2010.

[13] P. Porras, H. Saidi, and V. Yegneswaran. A foray into conficker's logic and rendezvous points. In Proc. of the 2nd USENIX Conference on Large-scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (LEET'09), Boston, Massachusetts, USA. USENIX Association, April 2009.
[14] M. Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A multifaceted approach to understanding the botnet phenomenon. In Proc. of the 6th ACM SIGCOMM Conference on Internet Measurement (IMC'06), Rio de Janeiro, Brazil, pages 41–52. ACM, October 2006.
[15] BankPatch. Trojan.Bankpatch.C. http://www.symantec.com/security_response/writeup.jsp?docid=2008-081817-1808-99&tabid=2, 2009.
[16] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi. EXPOSURE: Finding malicious domains using passive dns analysis. In Proceedings of NDSS, 2011.
[17] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In Advances In Neural Information Processing Systems, pages 849–856. MIT Press, 2001.
[18] S. Shevchenko. Domain name generator for murofet. http://blog. threatexpert.com/2010/10/ domain-name-generator-for-murofet.html, 2010.
[19] J. Stewart. Bobax trojan analysis. http://www.secureworks.com/research/threats/bobax/, 2004.
[20] 张雪松，徐小琳，李青山，算法生成恶意域名的实时监测，现代电信科技，2013.
[21] 蔡冰，马旸，王林汝，一种恶意域名检测技术的研究与实现，江苏通信，2015.