计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2017.20170403
54(10):2334-2343,2017

高级持续性威胁中隐蔽可疑 DNS 行为的检测

王晓琪[1] 李 强[1,2] 闫广华[1] 玄光哲[3] 郭 东[1,2]
[1](吉林大学计算机科学与技术学院 长春 130012)
[2](符号计算与知识工程教育部重点实验室(吉林大学) 长春 130012)
[3](吉林大学大数据和网络管理中心 长春 130012)
(xqwang15@mails.jlu.edu.cn)

Detection of Covert and Suspicious DNS Behavior in Advanced Persistent Threats

Wang Xiaoqi[1], Li Qiang[1,2], Yan Guanghua[1], Xuan Guangzhe[3], and Guo Dong[1,2]
[1](*College of Computer Science and Technology, Jilin University, Changchun* 130012)
[2](*Key Laboratory of Symbol Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun* 130012)
[3](*Center of Big Data and Network Management, Jilin University, Changchun* 130012)

Abstract In recent years, advanced persistent threats (APT) jeopardize the safety of enterprises, organizations and even countries, leading to heavy economic losses. An important feature of APT is that it can persist in attacking and can lurk in the target network for a long time. Unfortunately, we cannot detect APT effectively by current security measures. Recent researches have found that analyzing DNS request of the target network will help detect APT attacks. We add a time feature in the DNS traffic which is combined with change vector analysis (CVA) and reputation score to detect covert and suspicious DNS behavior. In this paper, we propose a new framework called APDD to detect covert and suspicious DNS behavior in long-term APT by analyzing a mass of DNS request data. We execute the data reduction algorithm on DNS request data and then extract their features. By using the CVA and the sliding time window method, we analyze the similarity between the access records of the domains to be detected and those of the related domains of current APT. We build a reputation scoring system to grade the domain access records of high similarity. The APDD framework will output a list of suspicious domain access records so that security experts are able to analyze the top-k records in the list, which will surely improve the detection efficiency of APT attacks. Finally, we use 1 584 225 274 pieces of DNS request records which come from a large campus network and then simulate the attack data to verify the effectiveness and correctness of APDD. Experiments show that the APDD framework can effectively detect covert and suspicious DNS behavior in APT.

Key words advanced persistent threats (APT); DNS request data; data reduction; change vector analysis (CVA); reputation score

摘 要 近年来,高级持续性威胁(advanced persistent threats, APT)危害企业、组织甚至国家安全,给目标带来了巨大的经济损失,其重要特征是攻击持续时间跨度大,在目标网络内长期潜伏.现有的安全防御措施还无法有效检测 APT.现有研究认为通过分析 APT 攻击中目标网络的 DNS 请求,可以帮助

收稿日期:2017-06-07;**修回日期**:2017-07-28
基金项目:国家自然科学基金项目(61472162,61772229)
This work was supported by the National Natural Science Foundation of China (61472162, 61772229).
通信作者:郭东(guodong@jlu.edu.cn)

检测 APT 攻击.增加 DNS 流量中的时间特征结合变化向量分析和信誉评分方法来检测隐蔽可疑的 DNS 行为.提出一种协助检测 APT 的框架 APDD,通过分析大量的 DNS 请求数据检测长时间周期下 APT 中隐蔽可疑的 DNS 行为.将收集到的 DNS 请求数据执行数据缩减并提取特征;利用变化向量分析方法(change vector analysis, CVA)和滑动时间窗口方法分析待检测域名访问记录与现有 APT 相关域名之间的相似度;建立一个信誉评分系统对相似度较高的待检测域名访问记录进行打分;APDD 框架输出一个可疑域名访问记录排名列表,可用于后续人工优先分析最可疑的记录,从而提高 APT 攻击的检测效率;利用一个大型校园网中收集的包含 1 584 225 274 条 DNS 请求记录的数据加入仿真攻击数据来验证框架的有效性与正确性,实验结果表明:提出的框架可以有效地检测到 APT 中隐蔽可疑的 DNS 行为.

关键词 高级持续威胁;DNS 请求数据;数据缩减;变化向量分析;信誉评分

中图法分类号 TP393

近年来,高级持续性威胁模式的攻击不断出现,如 Google 极光、夜龙、360 追日团队揭发的海莲花攻击[1]、腾讯反病毒实验室揭发的黑暗幽灵木马[2]等.海莲花攻击是境外组织对国内有组织、有计划、有针对性的长时间不间断攻击,通过鱼叉攻击[3]和水坑攻击[4]等方法结合社会工程手段渗透,向特定目标传播木马,秘密控制目标系统并窃取机密资料.黑暗幽灵木马通过监听常用聊天软件的语音文字信息,接受外部指令进行远程操控,将自动化收集到的各种信息文件打包外传.这些利用高级持续威胁(advanced persistent threats, APT)模式的网络攻击,涉及大规模网络监控、网络窃密等活动,已经严重危害国家安全和用户信息安全,其对关键基础设施乃至整个经济社会的神经中枢带来了灾难性后果.

APT 攻击的一个显著特点是持续性,攻击持续时间跨度大,在目标网络内潜伏时间长,通过定制恶意软件,利用变形、加密等技术逃避检测,并在目标网络内不断扩散,收集和窃取高价值的数据信息并向外传输.根据文献[5-7],几乎所有 APT 攻击都必须经历 4 个阶段:准备、攻击、潜伏和数据传回.APT 攻击的检测难处在于攻击周期长和攻击过程中的隐蔽性,每个攻击环节混杂在大量正常流量中,使得 APT 攻击表现出来的微弱信号很难被检测到.正是因为 APT 的特点和难点,现有的安全防御措施如防火墙、反病毒工具、入侵检测系统等还无法有效检测 APT.

APT 通常是低速、低流量的攻击模式.在 APT 最重要的访问阶段,攻击者必须通过 DNS 将恶意程序、后门木马等恶意软件安装到内部主机才能进一步获得该主机的控制访问权,在数据传回阶段,内部感染主机也需要跟外部 C&C 服务器建立联系才可以将收集到的机密信息传递出来,总之 APT 经常需要通过与 DNS 通信发挥作用.当前,国内外研究人员已尝试将 DNS 分析用于检测 APT 攻击,Infoblox 公司将其 DNS Firewall[8]与 FireEye 公司的恶意软件防御系统整合用于防御 APT.文献[9]将 DNS 数据用于机器学习来检测恶意域名;文献[10]发现了 DNS 中域名之间的关系,提出了一种确定性算法来检测恶意域名;文献[11]提出了基于图论中置信传播的框架来检测 APT 早期阶段中的可疑域名;文献[12]根据 APT 攻击潜伏期的特点提取 DNS 数据中的特征,利用机器学习来检测 APT 攻击;文献[13]通过提取 DNS 流量中的特征对域名进行分类,达到检测恶意域名的目的.虽然以上工作在利用 DNS 分析 APT 上取得了突破,但还存在 2 个问题:检测框架或方法的周期短,不符合 APT 持续性的特点;没有利用到 APT 攻击过程中时间这一重要特征.

为了解决现有工作存在的问题,本文重点关注 APT 攻击在 4 个阶段表现出来的与 DNS 数据相关的微弱信号.例如在访问阶段,攻击者精心准备攻击方式并且只攻击单个主机或者很少的主机,大大降低了被检测到的可能性;在潜伏阶段,被感染的内部主机为了避免边界检测,几乎不会访问外部 C&C 服务器.通过扩大检测时间窗口和加入时间相关特征来提高检测可疑 DNS 行为的准确率,结合变化向量分析和信誉评分方法将低速、低流量模式下的隐蔽可疑的 DNS 行为突显出来.

本文分析和总结了大量的 APT 攻击报告[14],总结出 APT 的 4 个阶段的持续时间周期以及 APT 在各个阶段所表现出来的微弱信号(特征),提出一个协助检测框架 APDD,该框架基于 CVA 和信誉

评分方法来检测可疑域名访问记录.1)将采集到的DNS数据执行不失真的数据缩减算法,减少数据规模可以提高数据分析和处理的效率;2)进行特征提取,总共提取了与APT活动相关的5类共10个特征;3)用第1,3类特征组成特征向量用于CVA方法得出与标准APT相关域名行为相似度较高的域名访问记录,将得到的域名访问记录输入到一个信誉评分系统用第2,4,5类特征对待检测域名访问记录进行打分,输出可疑域名访问记录排名列表;4)从国内教育网中最大的校园网之一吉林大学校园网中,收集了连续99 d共1 584 225 274条的DNS请求记录,用这些数据对算法和系统进行测试和评价.

本文的主要贡献有4个方面:

1)提出了一种协助检测APT的系统框架APDD,通过分析大量的DNS请求数据来检测隐蔽可疑的DNS行为;

2)提出了一种不失真的数据缩减算法,降低大数据的规模,提高检测效率;

3)将变化向量分析方法和信誉评分方法结合,分析DNS数据中抽取出的特征,从而将DNS数据中的微弱信号突显出来;

4)利用吉林大学校园网收集的大量DNS请求数据(加入仿真的APT攻击DNS流量)验证方法和框架的有效性.

1 相关工作

国内外具有代表性的相关工作主要如下:

综述类文献[5,15-17]分析了当前流行的APT案例,总结出APT攻击的多阶段性、特点、难点以及未来的研究方向,并没有给出具体的检测方法和框架.

首先介绍基于DNS的方法.

1)佐治亚理工学院提出了一种检测特定APT的系统框架ghost& rae[12],提出APT在整个过程中表现出来的特点:在内部网络中将外部域名解析成本地或者非路由IP,只在与外界联系时才解析成外部IP,根据该特点提取特征进行机器学习,找出与特定APT相关的外部恶意域名,然而这种方法的只针对一种特定的APT攻击,这就使得该框架具有很大的局限性.

2)佐治亚大学提出了Segugio系统[9],该系统在大型ISP网络中可以有效的追踪到未知的被恶意控制的域名,用大量监测的DNS流量建立machine-domain双边图,标记节点提取特征,利用机器学习方法可检测到未知的被恶意控制的域名,该方法受ground-true数据的限制,如果被标记数据不准确产生的结果也不准确.

3)RSA实验室提出了一种基于图论中置信传播的新框架来解决企业感染的早期检测问题[11],置信传播可以工作在以专业人员提供的被感染主机以及恶意域名为种子的模式也可以工作在没有种子的模式,进而建立了一个C&C通信检测器用来检测主机与C&C服务器之间的通信,最终通过置信传播的算法输出可疑域名,该框架只检测从来没有访问过或者很少访问的目的地址,数据量大大减少,这也是该框架最大的限制,使得攻击者可以采取一定技术和策略躲避检测.

4)卡塔尔计算研究所在他人研究基础之上提出了一种补充的方法[10],发现并分析了域名之间的关联,假设与恶意域名有强关联关系的域名也很有可能是恶意的,提出一个确定性算法检测未知的恶意域名,该算法只能作为一种补充方法并不能独立检测.

5)清华大学提出了系统框架IDnS[18],该系统使用恶意的DNS分析技术检测可疑的APT C&C域名,然后使用基于签名的和基于异常的检测技术分析与可疑IP相关的流量,建立一个信誉评价引擎来计算待检测IP地址的信誉分值.

6)美国的东北大学和加利福尼亚大学提出一种检测框架EXPOSURE[13],提取DNS流量中15种特征(其中9种特征之前的研究没有提及),利用分类方法检测恶意活动中的域名.本文与之相比,扩大了检测周期,利用变化向量分析和信誉评分相结合的方法将长时间窗口下隐蔽可疑的DNS行为突显出来.

除关注DNS的方法之外,意大利的University of Modena and Reggio Emilia提出了检测数据泄露或其他APT活动所产生的微弱信号的方法[19],收集IP报头的某些字段从数据中提取特征,然后利用信誉评价方法,输出一个可疑内部主机的排名列表,这有助于安全人员集中分析小部分的主机.该文献抽取的特征较少,这样会造成结果出现偏差.意大利的University of Modena and Reggio Emilia还提出一种新的框架[6],大数据与安全情报相结合分析最后得出可能被感染的主机集合.该方法类似于信誉评分也是通过各种指标的分值最后得出可疑的主机排名.

本文提出的框架解决了现有相关工作中的 2 个不足之处:1)文献[11]使用 1 个月的训练数据,文献[9-10]只用大约 2 周的数据集,而本文的框架使用了连续 99 d 的 DNS 请求数据集,扩大检测周期有利于发现 APT 整个过程中的长期隐蔽信号;2)文献[9-12,18-19]中都没有用到和时间相关的特征,APT 是一种长期潜伏的攻击模式,时间特征是一种重要的特征表现,本文中加入时间相关特征,结合相关工作中的部分特征将 APT 中隐蔽可疑的 DNS 行为突显出来.

2 检测框架

检测框架 APDD 通过分析大量的 DNS 数据来检测隐藏在其中的与 APT 相关的可疑 DNS 行为.框架 APDD 的流程图如图 1 所示:

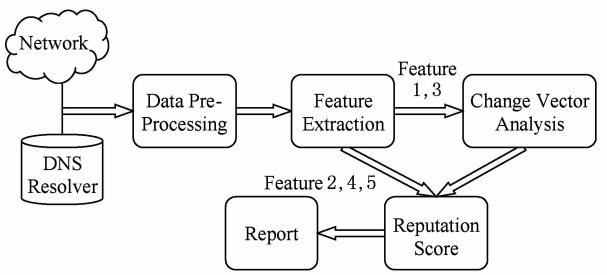

Fig. 1 APDD overview

图 1 检测框架 APDD

2.1 数 据

通过国内教育网中最大的校园网之一的吉林大学校园网,已经收集了 215 946 台主机设备(含无线接入)共 1 584 225 274 条的 DNS 请求记录,这些数据可用来对算法和系统进行测试.为了保护个人隐私,对数据的主机 IP 字段进行了匿名化处理.这批数据包括了 2016.05.23—2016.08.30 之内连续 99 d 的 DNS 请求记录,有源端口、内部主机 IP、访问域名、访问日期时间等字段,根据这些数据测试框架和方法的效率和性能,也可以从这些数据总结出 APT 各个阶段 DNS 数据的特征.

通过分析近期 APT 攻击的报告[20-25],发现这些报告有明确的时间前后关系并且给出了部分相关恶意域名以及对应的访问时间等,以不改变文献中域名访问时间差的方式生成 6 种独立的 DNS 仿真数据(将仿真数据标记为 $S1\sim S6$)混入到已有的 DNS 请求数据中.例如文献[20]中提到了恶意域名 ustar5.PassAs.us 的访问时间为 03/12/2015 与 15/12/2015,考虑已有 DNS 数据的时间跨度,将域名改为 exp.PassAs.us 或其他格式,时间改成 06/10/2016 和 06/22/2016 生成对应的 DNS 记录混入到 DNS 数据中去,主机 IP 以及访问具体时间选择随机生成的方式.以上数据将在后文中用来测试算法和框架的有效性.

文献[11]中用到的实验数据是美国洛斯阿拉莫斯国家实验室(Los Alamos National Laboratory,LANL)提供的 DNS 数据,该数据集是在正常 DNS 流量中加入仿真攻击数据流.本文的实验数据是在吉林大学大数据和网络管理中心收集的真实 DNS 数据中加入从已有 APT 报告中模拟出来的攻击数据流.实验部分用此数据集验证提出方法和框架的有效性.

2.2 数据预处理

APT 攻击整个过程时间跨度长,积累的数据量也是巨大的,在长达几个月的时间里,数据的分析和处理异常困难,而其中绝大部分都是正常用户访问的记录,这就需要一种数据缩减算法来降低数据规模,提高数据的处理效率,第 3 节 DNS 数据缩减算法中将详细介绍提出的数据缩减算法.

2.3 特征提取

介绍特征之前,先定义一些变量.

$T1$:APT 攻击准备阶段的时间窗口;

$T2$:APT 攻击访问阶段的时间窗口;

$T3$:APT 攻击潜伏阶段的时间窗口;

$T4$:APT 攻击数据传回阶段时间窗口;

W:$T1+T2+T3+T4$,整个 APT 攻击时间窗口.

基于 APT 各阶段 DNS 数据表现出来的特征以及 DNS 请求数据本身的特征总结列出提取的特征集合,总共提取出 5 类特征:

$F1$:域名访问次数,内部主机访问待检测域名的次数;

$F2$:域名访问时间点,内部主机访问待检测域名的时间点;

$F3$:域名访问时间间隔,内部主机访问待检测域名之间的时间间隔;

$F4$:域名的流行程度,待检测域名在检测时间点之前的访问量;

$F5$:域名相似度,待检测域名与网址白名单之间的相似度.

1) 域名访问次数. APT 攻击是一种人为定制的攻击模式,在整个攻击过程中为了躲避边界检测,攻击者在内部网络中的行为非常小心,会非常少的

访问外部域名或者服务器,所以认为域名访问次数少的主机-域名记录是可疑的[19].提取内部主机在APT各个阶段时间窗口内访问待检测域名的次数作为特征值,定义为 $T1\_count$,$T2\_count$,$T3\_count$,$T4\_count$.

2) 域名访问时间点.用户在正常情况下的访问时间点是有一定规律的,所以认为不正常的访问时间点是可疑的.例如某主机在0:00—5:00时间段有访问记录是可疑的行为.提取APT整个过程不正常时间点的访问次数作为特征值,定义为 $T\_timepoint$.

3) 域名访问时间间隔.在APT的 $T3$ 阶段有一段很长的潜伏期,该期间攻击者主要进行2个方面的活动:横向移动获取其余目标主机的访问权限;收集有用的信息并集中起来.攻击者会尽量的隐藏自己的行为所以几乎不会访问外部域名或者服务器,这就使得长时间窗口下访问域名的时间间隔变大.所以提取各阶段最接近中心点访问记录的时间间隔作为特征值,定义为 $T12\_interval$,$T23\_interval$,$T24\_interval$.

4) 域名的流行程度.经常被内部主机访问的网址基本全是受信任的,如果一个域名访问之前从来没有被任何一台内部主机访问或者很少访问,则认为这个域名是可疑的[11],所以提取待检测域名之前访问过的次数作为域名流行程度的特征值,定义为 $raredomain$.

5) 域名相似度.攻击者可能利用与非常流行的网站相似的域名来实施攻击,用户观察不仔细就会成为受害者,内部主机就容易被感染,如果一个域名与网址白名单中的网址相似度很高,则认为该域名是可疑的,所以提取待检测域名与网址白名单中的相似度作为特征值,定义为 $domainsimilarity$.

以上的第1,3类特征组成特征向量($T1\_count$,$T2\_count$,$T12\_interval$,$T3\_count$,$T23\_interval$,$T4\_count$,$T24\_interval$),该向量用于第2.4节的CAA算法.第2,4,5类特征用于信誉评分系统.

2.4 变化向量分析

根据最新的APT相关报告[14]以及各类综述性论文[5,15-17]中,总结出APT整个过程的自定义标准值 $T1$,$T2$,$T3$,$T4$,W,以及在各个阶段中最有可能的特征值($F1$,$F3$)组成标准特征向量 $\boldsymbol{F}$,$\boldsymbol{F}$ 与 $\boldsymbol{f}$ 一样都是7维的向量.具体的参数设置在实验部分给出.

变化向量分析方法[26]通过对不同期的影像各个波段的数据进行差值运算,求得每个像素在各个波段的变化量,再由各个波段的变化量组成变化向量.变化向量中,变化的强度用变化向量的欧氏距离表示,变化的内容用变化向量的方向表示.变化向量分析方法的变化向量以及变化方向如图2所示:

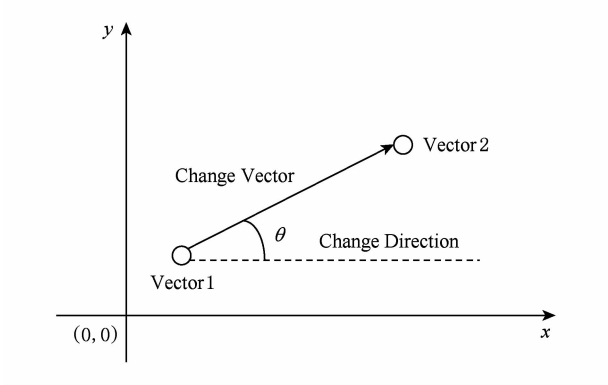

Fig. 2 Change vector analysis

图2 变化向量分析方法

$$\Delta v_i = r_i - s_i, i = 1,2,\cdots,n, \tag{1}$$

$$|\Delta v| = ((r_1 - s_1)^2 + (r_2 - s_2)^2 + \cdots + (r_n - s_n)^2)^{1/2}, \tag{2}$$

式(1)表示变化向量,式(2)表示变化向量的变化强度.在获得变化向量之后,通过阈值对变化向量进行分类,将特征向量分为未变化区域和变化区域,阈值的确定可以人工进行设置也可以根据一些算法进行计算.

本文基于变化向量分析方法提出CAA算法分析待检测域名访问记录的 $\boldsymbol{f}$ 与标准 $\boldsymbol{F}$,得出待检测域名访问记录与标准APT活动的相似度(未变化向量区域即为相似度较高的区域),详细的算法将在第4节中介绍.

2.5 信誉评分

第2.4节得到的属于未变化区域的域名访问记录存储着与标准APT特征向量相似度较高的一部分域名访问记录,建立一个信誉评分系统结合2.2节提出的特征($F2$,$F4$,$F5$)对这些记录进行打分,最后输出一个可疑域名访问记录的排名列表.信誉评分系统的大致流程如图3所示:

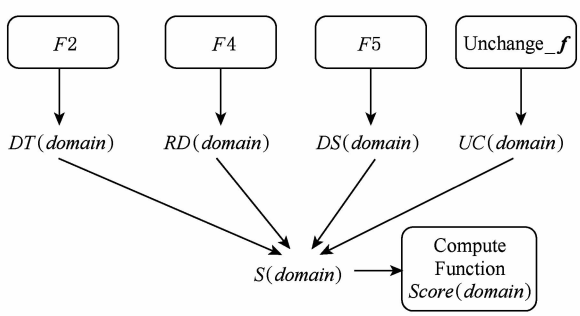

Fig. 3 Reputation score

图3 信誉评分

$RD(domain)$计算域名流行程度 *raredomain*,本文所指的流行程度只针对内部网络,如果一个域名从来没有被内部网络中的主机访问过或者很少访问,认为该域名的访问记录是可疑的.所以将待检测域名之前的总访问次数作为计算值.例如在检测 ms.wifienjoy.com 域名之前,整个内部网络一共访问过该域名 5 次,则 RD 值为 5.RD 值越小代表该域名访问记录越可疑.

$DT(domain)$计算域名访问时间点 *T_timepoint*,将不正常访问时间点的域名访问视为可疑的,在本文的实验中将 0:00—6:00 的时间段作为可疑时间段.假设在 W 窗口下,一台主机上某域名访问总次数为 n,在上述时间段的访问次数为 a,取 $DT=a/n$,DT 值越大代表该域名访问记录越可疑.这样取值是为了下面 *Score* 函数的计算方便.

$DS(domain)$计算域名相似度 *domainsimilarity*,有的攻击者会利用与流行网站相似度高的域名实施攻击,利用编辑距离算法求域名相似度,将待检测域名与 Alexa 网站排名前 20 万的网站作比较,最小的编辑距离作为 DS 值.DS 值越小该域名的访问记录越可疑.

$UC(domain)$取 2.4 节中得到的待检测域名对应的变化向量的变化强度(欧氏距离)作为 UC 值,变化向量求的是待检测域名的特征向量与 APT 标准向量的相似度,所以变化向量的变化强度越小,即 UC 值越小,域名访问记录越可疑.

$Score(domain)$计算待检测域名访问记录的最终分值,通过:

$$Score=\frac{1}{RD}+\frac{1}{DS}+\frac{1}{UC}+DT \tag{3}$$

计算带检测域名访问记录的 S 值,表示该记录的可疑程度.对于 DS 值,因为之前的数据预处理中已经将网址白名单的访问记录去除,所以待检测域名与流行网站的相似度是大于等于 1 的,不需要考虑 $DS=0$ 的情况,DS 取值范围是$(0,1]$;对于 RD 值,$RD=0$ 时,说明内部网络中从未访问过待检测域名,为了区别于 $RD=1$ 时的情况取 $1/RD=1.2$;对于 UC 值,$UC=0$ 时,该域名访问记录与给出的标准向量相比没有变化,为了区别 $UC=1$ 的情况取 $1/UC=1.2$.通过 *Score* 函数的计算得到一个最后的分数,代表域名访问记录的可疑程度,值越大,代表该记录越可疑.最终会输出一个域名访问记录的可疑程度列表,安全人员可以直接分析列表中靠前的记录,提高 APT 攻击的检测效率.

3 DNS 数据缩减算法

针对 10 亿以上的 DNS 请求数据,处理起来非常困难繁琐,提出一种数据缩减算法如算法 1 所示.该算法大规模减少原始数据的规模,可以加快对数据的分析和处理效率,算法主要基于 3 种规则:

1) 去掉非常流行的域名请求记录.在 APT 攻击过程中,如果攻击者想要通过攻克非常流行的网站发动攻击,需要付出的代价太大,所以认为非常流行的域名请求记录是良好的访问行为.

2) 去掉一天中单个主机访问单一域名次数超过 α 次(例如 11.22.33.44 主机在 2016 年 5 月 26 日访问 apple.com 超过 α 次,则删除 11.22.33.44 在 2016.5.26 所有访问 apple.com 的请求记录)的请求记录.攻击者为了躲避边界检测,会尽量少地访问外部 C&C 服务器或者域名.

3) 去掉检测时间窗口下总的访问记录少于 β 次的记录.明显地,长时间窗口下特别不活跃的主机对检测系统是没有帮助的,保守起见,只去掉总的访问记录少于 5 次的域名访问记录.

将数据的内部主机、访问时间和访问域名字段存入 Redis 数据库中,内部主机作为键,时间和域名一起作为值.Redis 是一个远程内存数据库,它不仅性能强劲,而且还具有复制特性以及为解决问题而生的独一无二的数据模型,它还是一个高性能的支持主从同步的键值存储系统,这样处理可以加快数据的分析和处理效率.

算法 1. 数据缩减算法.

输入:原始的 DNS 请求记录 *raw_records*;

输出:缩减后的 DNS 请求记录 *test_records*.

/* *whitelist* 为 Alexa 网站排名前 100 万白名单集合,*host_set* 为主机集合 */

/* 规则 1 */

① *raw_records* 存储在 Redis 数据库中;

② foreach *record*∈*raw_records* do

③ if *record* 中域名对应的顶级域名在 *whitelist* 中 then

④ 删除 *record*;

⑤ end if

⑥ end for

/* 规则 2 */

⑦ foreach *host*∈*host_set* do

⑧ if *host* 在一天中访问 *domain* 的次数超过 α 次 then
⑨ 删除当天 *host* 访问 *domain* 的记录;
⑩ end if
⑪ end for
/ * 规则 3 * /
⑫ foreach $host \in host\_set$ do
⑬ if *host* 总的访问记录数少于 β then
⑭ 删除 *host* 的所有访问记录;
⑮ end if
⑯ end for
⑰ 剩余的记录存储在 *test_records*;
⑱ return *test_records*.

4 CAA 算法

根据变化向量方法提出一种新的基于 CVA 的 CAA 算法如算法 2 所示,该算法加入滑动时间窗口方法,提高检测算法的效率. CAA 算法用来找到与标准 APT 中 DNS 行为相似的域名访问记录,属于未变化向量区域的记录即为相似度比较高的.

算法 2. CAA 算法.

输入:经过数据预处理后的 DNS 请求记录 *test_records*;

输出:未变化区域的记录 *unchange_f*.

/ * *host_set* 为主机集合,*enddate* 为已有数据的最后一天日期,*visitdate* 为记录的访问日期,*W* 滑动时间窗口 * /
/ * *f_records* 存储域名访问记录以及对应的特征向量,*unchange_f* 存储属于未变化特征向量的域名访问记录 * /
/ * δ 为划分变化向量的阈值,Δv 为变化向量 * /
① do while $W\_end < enddate$
② foreach $host \in host\_set$ do
③ foreach *host* 访问过的域名 *domain* and $visitdate \in W$ do
④ $\boldsymbol{f} = ComFeaVector(domain)$; / * 计算该域名访问记录的特征向量 * /
⑤ 将 *host*,*domain*,$\boldsymbol{f}$ 存储在 *f_records* 中;
⑥ end for
⑦ end for
⑧ foreach *host*,*domain*,$\boldsymbol{f} \in \boldsymbol{f}\_records$ do
⑨ $\Delta v = \boldsymbol{f} - \boldsymbol{F}$;
⑩ if Δv 的变化强度 $< \delta$ then
⑪ 将属于未变化区域的记录存储在 *unchange_f* 中;
⑫ end if
⑬ end for
⑭ $W += 1$;
⑮ end while
⑯ return *unchange_f*.

CAA 算法以数据预处理之后的 DNS 请求记录为输入,*test_records* 为初始数据经过不失真处理之后的结果,*unchange_f* 为输出的属于未变化向量区域的域名访问记录,该算法同时还利用了滑动时间窗口方法,以自定义的 APT 整个周期 *W* 为滑动时间窗口,采用滑动时间窗口方法只考虑 *W* 窗口之内的 DNS 请求数据,降低了数据规模,减少了一次算法的运行时间,提高了算法的效率. 算法主要分为 3 部分:

1) 设定滑动时间窗口 *W*;

2) 针对 *W* 下的所有域名访问记录通过 *ComFeaVector* 函数计算得到该记录的特征向量 $\boldsymbol{f}$;

3) 变化向量分析方法分析 $\boldsymbol{f}$ 得出属于未变化向量范畴的域名访问记录. CAA 算法的输出结果被用于框架中的信誉评分系统.

CAA 算法通过 2 个方面来提高运行效率:1)直接操作 Redis 数据库中存储的 DNS 数据,Redis 是一个高性能的 key-value 数据库,直接运行在内存中,大大提高了算法的运行效率;2)算法还加入滑动时间窗口方法,只考虑设定时间窗口 *W* 下的数据,减少了每一次分析的数据规模,提高了检测效率. 在实验部分中,实验表明在针对长时间周期下的大数据时算法 CAA 也能达到较好的效率.

5 实 验

利用 2.1 节提到的数据对提出的框架以及方法进行测试,检测框架的目的是输出与 APT 相关的可疑域名访问记录排名列表,可以让安全人员优先分析这些记录来提高对 APT 攻击的检测以及防范.

进行实验的目的是证明 3 个方面:

1) 说明提出的框架可以在真实的操作系统环境中实现;

2) 验证框架检测与 APT 相关可疑域名访问记录的能力;

3) 验证框架的可扩展性以及灵活性.

5.1 实验环境和数据情况

本节介绍实验的具体环境以及使用的数据的具体情况.

在 1 台 4 核 8 线程数、内存 32 GB、硬盘 2 T 的 CentOS 系统服务器上部署提出的框架,并且在该系统上安装了 Redis 用于存储数据,可以看出提出的框架对于硬件设备的要求是比较低的.

原始数据与经过数据预处理后的数据的对比情况如表 1 所示. 由表 1 可以看出,经过数据缩减算法处理后 DNS 请求记录大大减少,提高了数据分析和处理效率.

Table 1 Scale of Data

表 1 数据规模

Data	Devices	Records
raw data	215 946	1 584 225 274
data pre-processing	146 234	45 122 416

5.2 参数设置和实验结果

通过阅读分析近年来大量的 APT 报告[14,20-25],总结出 APT 的 4 个阶段 $T1 \sim T4$ 的时间跨度以及标准的特征向量 $\boldsymbol{F}$ 如表 2 所示:

Table 2 Parameter Setting

表 2 参数设置

Experiment	$T1$/d	$T2$/d	$T3$/d	$T4$/d	W/d	$\boldsymbol{F}$
Experiment 1	5	5	5	10	25	(0,3,0,0,0,2,12)
Experiment 2	5	5	10	10	30	(0,3,0,0,0,2,17)
Experiment 3	5	5	20	10	40	(0,3,0,0,0,2,27)
Experiment 4	5	5	30	10	50	(0,3,0,0,0,2,37)

将准备阶段($T1$)设置为 5 d,这一阶段对于实验没有太大的影响,因为准备阶段不会产生与攻击相关的 DNS 流量,访问阶段($T2$)设置为 5 d,该阶段攻击者只会以单个主机或者少数主机为攻击目标,为了获得该主机的访问控制权,目标主机一定需要访问外部服务器或者主机这就产生了 DNS 流量,一旦获得访问控制权,为了躲避边界检测争取更多的时间进行剩余的操作,该主机就会进入一定时间的潜伏期,最大的差别是在潜伏阶段($T3$),所以在参数设置时选取了 4 种不同的时间窗口来代表 $T3$ 阶段(分别是 5 d,10 d,20 d,30 d). 这一阶段攻击者只在内部网络中有少许的活动并不会产生 DNS 流量,最后是数据回传阶段($T4$),该阶段设置为 10 d,并不是因为数据回传阶段时间跨度也大,而是潜伏期的时间跨度不统一,所以将紧接的 $T4$ 阶段也设置大些,可以用来调节并适应 $T3$ 阶段的时间窗口大小. 综合分析 APT 攻击的 4 个阶段,总结出标准特征向量 $\boldsymbol{F}$. $T1\_count$, $T3\_count$, $T12\_interval$, $T23\_interval$ 肯定是 0,根据报告以及实际情况设置 $T2\_count$ 为 3 次, $T4\_count$ 为 2 次, $T24\_interval$ 为 $T2$ 阶段与 $T4$ 阶段最接近各个阶段时间窗口中心的访问记录的访问时间差(天数), $T24\_interval$ 根据 $T3$ 的变化而变化. 对于数据缩减算法以及 CAA 算法中的参数设置如下:为了保守起见,数据缩减算法中的 $\alpha=20$,CAA 算法中的 $\beta=4$.

实验结果输出的是可疑的域名访问记录列表,实验结果中仿真攻击的检测情况如表 3 所示:

Table 3 Experiment Result

表 3 实验结果

Experiment	Rank of Emulation Attack
Experiment 1	$S1$:50, $S4$:65
Experiment 2	$S4$:36, $S6$:49
Experiment 3	$S2$:11
Experiment 4	$S3$:15

实验结果表明:提出的框架可以有效地检测到仿真攻击数据并且排名相对靠前,都在前 100 排名. 不难发现,仿真攻击样本 $S5$ 没有被检测出来,单独分析 $S5$ 样本数据,发现该攻击潜伏周期已经超出了实验的参数设置范围,经过测试,在扩大参数 $T3$ 至 40 d 后,实验结果可以检测到 $S5$ 排名在 5.

通过阅读分析 APT 报告以及进行的 4 组实验可以得出, $T1$, $T2$ 和 $T4$ 阶段的参数的大小设置比较容易,对实验结果影响较大的是 $T3$ 阶段的窗口大小设置. 不同的潜伏周期决定了不同的 $T3$ 的值,参数设置也直接影响了实验结果. 例如,不同的实验检测到的攻击样本不同,实验 3 只能检测到 $S2$ 样本,实验 4 只能检测到 $S3$ 样本;实验 1 和实验 2 都检测到了 $S4$ 样本,但是明显的实验 2 比实验 1 的结

果要好. 所以参数设置对实验影响较大,必须根据实际情况严格控制参数设置,使之达到最理想的实验结果.

5.3 实验结果总结

通过实验证明提出的框架具有较好的灵活性以及扩展性,试验中只设置了 4 组参数,参数设置对实验结果影响较大,必须根据现有的 APT 相关报告严格控制参数设置,该框架可以根据实际情况更新参数的设置. 整个框架都是基于特征的,一旦发现新的特征完全可以量化之后加入到框架中达到提高检测的正确率和效率的效果.

根据所有的实验结果,提出的框架有效地检测到仿真攻击样本,再者提出的方法是严格基于 APT 多阶段攻击以及 DNS 数据本身的特点,所以有理由相信输出的可疑域名访问记录排名对于安全人员快速分析检测 APT 攻击是有一定帮助的.

6 讨论和总结

本文提出了一种协助检测 APT 的框架 APDD,用于检测 DNS 数据中与 APT 活动相关的微弱信号,通过提取分析 APT 的 4 个阶段和 DNS 数据本身的特征,结合变化向量分析和信誉评分系统的方法,最终输出可疑域名访问记录的排名列表,安全人员可以优先分析最可疑的记录来提高 APT 的检测效率.

APDD 是一个检测可疑 DNS 行为的框架,APT 攻击是一种针对性很强的多步攻击,在发动攻击时只感染单个或者很少的主机数,在检测到可疑的域名访问记录时,安全人员可将主机和域名一起分析来提高检测效率. 框架还拥有较好的灵活性以及扩展性,可以适应潜伏周期不同的多种 APT 攻击模式.

本文提出的框架和方法也有一定的局限性. 如果攻击者在潜伏期也不定期访问外部 C&C 服务器则框架检测的概率会降低,但是这样也增大了边界防御系统检测到的概率. 如果攻击者改变了正常的攻击模式行为,也会降低检测率,APDD 有一个很大的影响因素是参数设置,安全人员可以根据各种 APT 最新报告中总结出攻击行为的变化趋势从而改变参数设置来达到适应各种 APT 攻击模式的效果.

在以后的工作中,可以借助其他方法,例如改进的机器学习、改进的图算法等来检测 APT 攻击. 除了利用 DNS 数据,还可以收集多种日志数据进行关联日志分析,实现 APT 攻击过程的还原.

参 考 文 献

[1] Reed T. OceanLotus malware attacks China [EB/OL]. 2015 [2017-05-23]. http://www.thesafemac.com/oceanlotus-malware-attacks-china

[2] Tencent PC Manager. DCM detailed analysis [EB/OL]. 2015 [2017-05-24]. http://www.freebuf.com/articles/system/101447.html (in Chinese)
(腾讯电脑管家. 黑暗幽灵(DCM)木马详细分析[EB/OL]. 2015 [2017-05-24]. http://www.freebuf.com/articles/system/101447.html)

[3] Ekawade S, Mule S, Patkar U. Phishing attacks and its preventions [J]. Imperial Journal of Interdisciplinary Research, 2016, 2(12): 1766-1769

[4] Alrwais S, Yuan Kan, Alowaisheq E, et al. Catching predators at watering holes: Finding and understanding strategically compromised websites [C] //Proc of the 32nd Annual Conf on Computer Security Applications. New York: ACM, 2016: 153-166

[5] Li Meicong, Huang Wei, Wang Yongbin, et al. The study of APT attack stage model [C] //Proc of the 15th Int Conf on Computer and Information Science. Piscataway, NJ: IEEE, 2016: 1-5

[6] Marchetti M, Pierazzi F, Guido A, et al. Countering advanced persistent threats through security intelligence and big data analytics [C] //Proc of the 8th Int Conf on Cyber Conflict. Piscataway, NJ: IEEE, 2016: 243-261

[7] Hutchins E M, Cloppert M J, Amin R M. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains [J]. Leading Issues in Information Warfare & Security Research, 2011, 1(1): 80

[8] Zuo Ming. Infoblox and FireEye display a new APT defence scheme [EB/OL]. 2013 [2017-05-25]. http://security.ctocio.com.cn/363/12762363.shtml (in Chinese)
(佐名. Infoblox 联合 FireEye 推出全新 APT 防御方案 [EB/OL]. 2013 [2017-05-25]. http://security.ctocio.com.cn/363/12762363.shtml)

[9] Rahbarinia B, Perdisci R, Antonakakis M. Segugio: Efficient behavior-based tracking of malware-control domains in large ISP networks [C] //Proc of the 45th Annual IEEE/IFIP Int Conf on Dependable Systems and Networks. Piscataway, NJ: IEEE, 2015: 403-414

[10] Khalil I, Yu Ting, Guan Bei. Discovering malicious domains through passive DNS data graph analysis [C] //Proc of the 11th ACM on Asia Conf on Computer and Communications Security. New York: ACM, 2016: 663-674

[11] Oprea A, Li Zhou, Yen Tingfeng, et al. Detection of early-stage enterprise infection by mining large-scale log data [C] // Proc of the 45th Annual IEEE/IFIP Int Conf on Dependable Systems and Networks. Piscataway, NJ: IEEE, 2015: 45-56

[12] Nadji Y. Understanding DNS-based criminal infrastructure for informing takedowns [D]. Atlanta, GA: Georgia Institute of Technology, 2015

[13] Bilge L, Kirda E, Kruegel C, et al. EXPOSURE: Finding malicious domains using passive DNS analysis [C] //Proc of the 18th Annual Network and Distributed System Security Symp. Reston, VA: Internet Society. 2011: 195-211

[14] Kbandla. APTnotes [EB/OL]. 2016 [2017-05-20]. https://github.com/kbandla/aptnotes

[15] Ussath M, Jaeger D, Cheng Feng, et al. Advanced persistent threats: Behind the scenes [C] //Proc of the 50th Annual Conf on Information Science and Systems. Piscataway, NJ: IEEE, 2016: 181-186

[16] Zhou Tao. Abnormal network behavior detection technology based on statistical learning [J]. Big Data Research, 2015, 1(4): 38-47 (in Chinese)
(周涛. 基于统计学习的网络异常行为检测技术[J]. 大数据, 2015, 1(4): 38-47)

[17] Fu Yu, Li Hongcheng, Wu Xiaoping, et al. Detecting APT attacks: A survey from the perspective of big data analysis [J]. Journal on Communications, 2015, 36(11): 1-14 (in Chinese)
(付钰, 李洪成, 吴晓平, 等. 基于大数据分析的 APT 攻击检测研究综述[J]. 通信学报, 2015, 36(11): 1-14)

[18] Zhao Guodong, Xu Ke, Xu Lei, et al. Detecting APT malware infections based on malicious DNS and traffic analysis [J]. IEEE Access, 2015, 3: 1132-1142

[19] Marchetti M, Pierazzi F, Colajanni M, et al. Analysis of high volumes of network traffic for advanced persistent threat detection [J]. Computer Networks, 2016, 109: 127-141

[20] Yip M. ELISE: Security Through Obesity [EB/OL]. 2015 [2017-05-27]. http://pwc.blogs.com/cyber_security_updates/2015/12/elise-security-through-obesity.html

[21] Dell SecureWorks Counter Threat Unit Threat Intelligence. Threat Group 3390 Cyberespionage [EB/OL]. 2015[2017-05-27]. https://www.secureworks.com/research/threat-group-3390-targets-organizations-for-cyberespionage

[22] Costin R, Maxim G. The Chronicles of the Hellsing APT: The Empire Strikes Back [EB/OL]. 2015 [2017-05-27]. https://securelist.com/analysis/publications/69567/the-chronicles-of-the-hellsing-apt-the-empire-strikes-back

[23] Andy S, Nicholas G, Abel T. Monsoon Analysis of an APT Campaign [EB/OL]. 2016 [2017-05-27]. http://docplayer.net/42098499- Monsoon-analysis-of-an-apt-campaign.html

[24] PassiveTotal. Operation DustySky [EB/OL]. 2016[2017-05-27]. http://blog.passivetotal.org/operation-dustysky-notes

[25] Reporting and Analysis Center for Information Assurance MELANI. Technical Report about the Malware used in the Cyberespionage against RUAG [EB/OL]. 2016 [2017-05-27]. https://www.melani.admin.ch/melani/en/home/dokumentation/reports/technical-reports/technical-report_apt_case_ruag.html

[26] Johnson R D, Kasischke E S. Change vector analysis: A technique for the multispectral monitoring of land cover and condition [J]. International Journal of Remote Sensing, 1998, 19(3): 411-426

Wang Xiaoqi, born in 1992. Bachelor. His main research interests include network security and the detection of APT.

Li Qiang, born in 1975. PhD, associate professor. His main research interests include network security and the detection of malicious code (li_qiang@jlu.edu.cn).

Yan Guanghua, born in 1995. Bachelor. His main research interest is network security (yangh2113@mails.jlu.edu.cn).

Xuan Guangzhe, born in 1965. Bachelor, associate professor. His main research interests include network security and network management (xgz@jlu.edu.cn).

Guo Dong, born in 1975. PhD, lecturer. His main research interest is cloud storage security.