计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200100060

Kaminsky 攻击及其异常行为分析

陈 曦 冯 梅 江 波
中国石油勘探开发研究院计算所 北京 100083

摘 要 Kaminsky 攻击是一种远程 DNS 投毒攻击,攻击成功后解析域名子域的请求都被引导到伪造的权威域名服务器上,危害极大。通过模拟攻击实验并分析攻击特征提出一种新的针对 Kaminsky 攻击的异常行为分析方法,该方法先提取 DNS 报文中时间、IP、DNS 中 Flags 和 Transaction ID 等信息,然后使用滑动窗口对 DNS Transaction ID 去重之后计算相同 IP 地址条件下 Transaction ID 的条件熵,最后用改进的 CUSUM 算法分析条件熵时间序列以检测攻击时间。此外,调取检测出的攻击时间内的数据,相同 IP 地址条件下 Transaction ID 的条件熵可以追溯到投毒目标权威域名服务器的 IP 地址。将攻击流量与正常流量混合作为分析样本,通过调整攻击代码参数模拟不同攻击模式,结果表明该方法不仅时间复杂度小,而且有较低的误检率、漏报率和较高的检测率,是一种有效的检测和分析手段。

关键词:Kaminsky 攻击;域名系统;行为分析;条件熵;CUSUM 算法;追溯
中图法分类号 TP393

Analysis of Kaminsky Attack and Its Abnormal Behavior

CHEN Xi, FENG Mei and JIANG Bo
Institute of Computing Technology of Research Institute of Petroleum Exploration and Development, Beijing 100083, China

Abstract Kaminsky attack is a kind of remote DNS poisoning attack. Since the attack is successful, requests for resolving the name of second-level domain are directed to a fake authoritative domain name server. This article proposes a novel method for detecting abnormal behaviors against Kaminsky attack s based on attack signatures. First, features such as time, IP, DNS Flags, and DNS Transaction ID in DNS packets are extracted. Then sliding window its applied to deduplicate the Transaction ID and calculate the conditional entropy of Transaction ID under the condition of the same IP address. Finally, improved CUSUM algorithm is applied to analyze time series of the conditional entropy to detect attack time. In addition, with data within the detected attack time, the conditional entropy could be traced back to the IP addresses of the poisoning target named the authoritative domain name server. The analysis sample consists of attack traffic and normal traffic. With different parameters of the attack code, simulations verify that this method not only has a small time complexity, but also has a low false positive rate, a low false negative rate, and a high detection rate. It is an effective means of detection and analysis.

Keywords Kaminsky attack, Domain Name System, Behavior analysis, Conditional entropy, CUSUM algorithm, Retrospect

1 引言

近年来,网络安全问题日益严峻。DNS(Domain Name System)在设计时并未考虑太多安全问题,其作为关键应用遭受了各类攻击。目前,主流的 DNS 安全协议有 DNSSEC 等,但 DNSSEC 由于需要更多的系统、网络资源和后期的升级维护,并不能全面推广以弥补 DNS 安全缺陷。Kaminsky 攻击一旦成功,请求将被解析到伪造的权威域名服务器上,危害极大。

目前已有一些 Kaminsky 攻击的防御措施,最常用的有 SPR(Source Port Randomization,端口随机化),就是本地域名服务器采用随机端口,其本质是提高猜解难度[1];类似的还有增加查询域名大小写认证[2]。响应报文检查[3]、将请求不存在域名的 IP 加入黑名单、查询 DNS 响应是否有多个等措施是从通信过程的角度进行防御。而增加权威名字服务器个数提高了真实应答报文比伪造应答报文早到达本地域名服务器的概率。以上这些方法只是降低了投毒成功的概率,而启用 DNSSEC 几乎可以杜绝 Kaminsky 攻击检测。但是在实际应用中开启 DNSSEC 会增加额外的开销,不是所有本地域名服务器都会开启 DNSSEC。

攻击检测方法主要分为特征匹配、统计和机器学习 3 类。特征匹配方法检测效果好,但依赖大样本和特征库,并需要及时更新。文献[4]提出一种 DNS 缓存投毒检测系统,该系统由递归查询管理器、迭代查询管理器、数据库管理器、缓存验证管理器和警报管理器组成。系统通过查询数据库判断数据包是否伪造。统计方法有些误检率较高,有待进一步改进。如文献[5]先分别统计同一源 IP 地址和 DNS 请求关键字的 A 资源记录的信息熵,针对统计出的异常区间计算相邻 FQDN(Fully Qualified Domain Name)的 Damerau-Levenshtein 距离来检测异常。由于正常 DNS 流量中的 FQDN 变化较大,该方法具有较高的误检率。

机器学习方法的时间和空间复杂度更高,不利于及时、在线检测。文献[6]用五元组特征、时间和空间相关特征以及触发缓存的 DNS 请求应答特征进行回归分析(如支持向量机),具体检测效果未知。由于构造的攻击包通常只有一个应答

通信作者:陈曦(304533929@qq.com)

域，没有附加域和授权域，文献[7]用授权域作为关键特征并通过贝叶斯分类法识别攻击数据包。但攻击者控制的域名可能具有多个授权域，该方法具有一定局限性。

本文研究 Kaminsky 攻击的原理，并在此基础上提取攻击特征，提出 Kaminsky 攻击的异常行为分析方法。

2　DNS

DNS 用于提供域名与 IP 地址之间的映射关系，是网页、邮件等网络应用的基石。DNS 作为重要的网络基础设施，于 2005 年被互联网治理问题工作组列为关键互联网资源[8]。

2.1　DNS 查询解析过程

DNS 域名解析分为两种：递归查询和迭代查询。主机向本地域名服务器查询一般用递归查询，而本地域名服务器向根域名服务器查询采用迭代查询。

假设想要查询的域名为 www.example.com，具体查询步骤如图 1 所示。

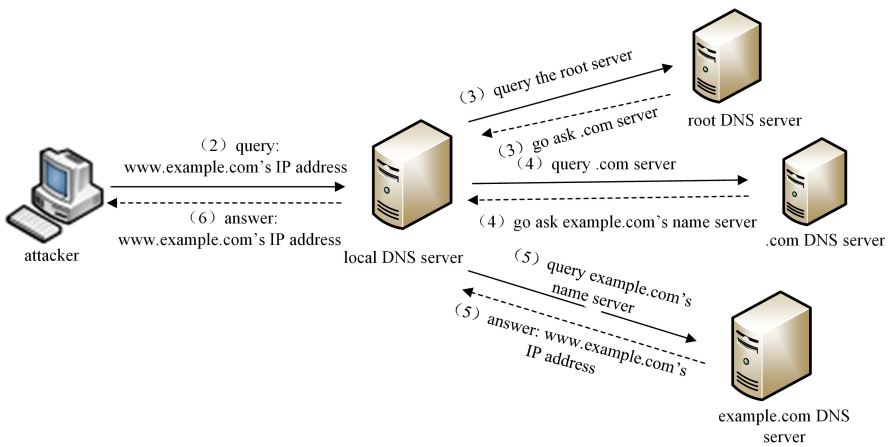

图 1　DNS 查询解析过程

Fig. 1　DNS query process

(1)本机查找 hosts 文件并判断客户端是否有该域名的缓存，如果有缓存则直接返回该域名对应的 IP 地址，否则转到(2)。

(2)本机向本地域名服务器请求 www.example.com 域名解析，本地域名服务器检查本地缓存，如果有就返回给本机结果，否则转到(3)。

(3)本地域名服务器向根域名服务器(.)查询 NS 记录和.com 域对应的 A 记录，根域名服务器返回结果。

(4)本地域名服务器向顶级域名服务器(.com)查询 NS 记，录和 example.com 域对应的 A 记录，顶级域名服务器返回结果。

(5)本地域名服务器向权限域名服务器(example.com)查询 NS 记录和 www.example.com 对应的 A 记录，权限域名服务器返回结果。

(6)本地域名服务器将查询到的结果保存在缓存中，并将结果返回给本机。

2.2　DNS 报文格式

DNS 报文分为查询报文和响应报文，一般采用 UDP 的 53 端口通信。DNS 报文结构如图 2 所示。

Transaction ID	Flags
Questions	Answer RRs
Authority RRs	Additional RRs
Queries	
Answers	
Authoritative nameservers	
Additional records	

图 2　DNS 报文结构

Fig. 2　structure of DNS packets

下面给出其中几个重要字段的含义。

Transaction ID：用于标识 DNS 报文 ID，查询报文及其对应的响应报文的 Transaction ID 相同。

标志(Flags)：DNS 报文的标志字段，描述 DNS 报文的类型。

权威名称服务器区域(Authoritative nameservers)：如解析 www.example.com 的过程中，顶级域名服务器(.com)给本地域名服务器响应.com 的 NS 记录(example.com)。

附加信息区域(Additional records)：如解析 www.example.com 的过程中，顶级域名服务器(.com)给本地域名服务器响应.com 的 NS 记录(example.com)的 A 记录。

3　DNS 投毒攻击

DNS 投毒攻击通过伪造响应报文投毒，如果本地域名服务器中缓存的域名-IP 对应关系不存在或已过期，而伪造的响应报文猜中 DNS 请求报文的 Transaction ID 并在真实响应报文之前到达，DNS 服务器就会接收伪造响应报文并缓存伪造的域名-IP 对应关系。

3.1　传统 DNS 投毒

传统 DNS 投毒攻击通过伪造响应报文的回答问题区域污染本地域名服务器缓存。传统 DNS 投毒受本地域名服务器缓存的 TTL 时间影响，即在真实域名-IP 地址记录不存在或过期时才有可能成功。

3.2　Kaminsky 攻击

Kaminsky 攻击通过伪造响应报文的权威名称服务器区域污染本地域名服务器缓存。为了绕过 TTL 约束，Kaminsky 攻击在传统 DNS 投毒的基础上变为请求主机名不存在而父域名存在的域名，如 abddc.example.com。

攻击过程具体如下：

(1)本机向本地域名服务器请求解析该域名；

(2)本地域名服务器查询缓存记录，由于没有 abddc.ex-

ample. com 的缓存，本地 DNS 服务器需要向根域名服务器(.)查询 NS 记录和. com 域对应的 A 记录，根域名服务器返回结果；

(3)本地域名服务器向顶级域名服务器(. com)查询 NS 记录和 example. com 域对应的 A 记录；

(4)本地域名服务器向权限域名服务器(example. com)查询 NS 记录和 www. example. com 对应的 A 记录；

(5)在本地域名服务器等待权限域名服务器的响应过程中，伪造的 DNS 响应包赶在真实响应包之前到达本地域名服务器并猜中 DNS 的 Transaction ID；

(6)伪造的响应包被本地域名服务器接收，伪造的 NS 记录被缓存；

(7)本地域名服务器请求伪造的权威域名服务器；

(8)伪造的权威域名服务器返回给本地域名服务器 abddc. example. com 的 IP 地址；

(9)本地域名服务器返回给本机 abddc. example. com 的 IP 地址。

投毒成功后，此时使用该本地域名服务器的正常用户如果请求解析 example. com 域名的子域名，都会被引导到伪造的 NS 记录上去。攻击过程如图 3 所示。

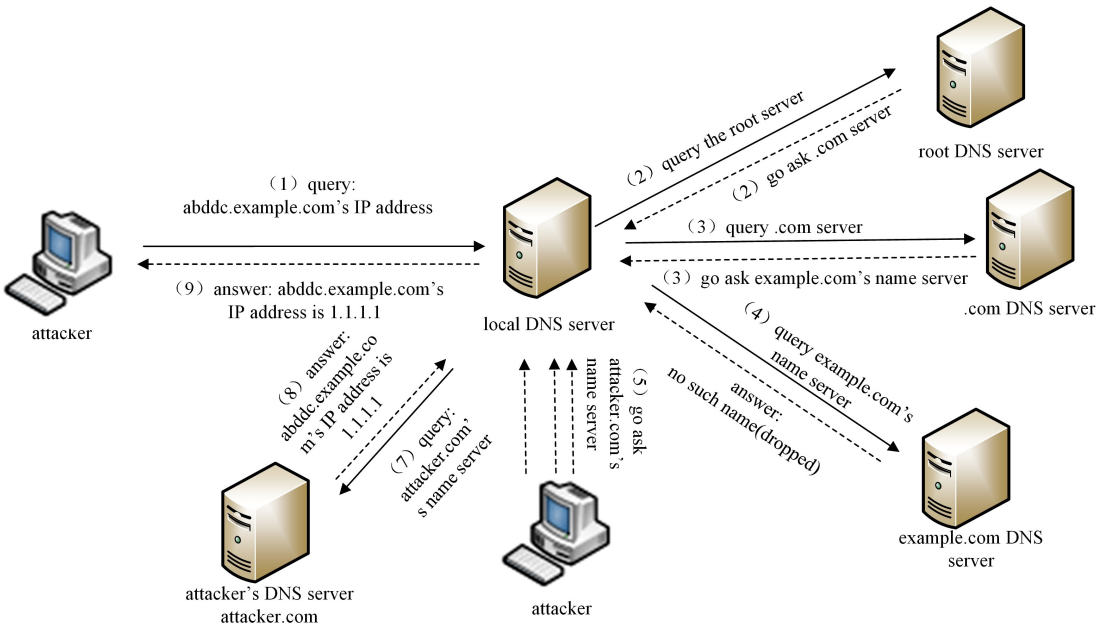

图 3 Kaminsky 攻击过程

Fig. 3 Kaminsky attack process

Kaminsky 攻击虽然绕过了一级域名的 TTL 约束，但是还受以下几个条件约束[9-10]。

(1)TTL 约束：域名如果在本地 DNS 服务器缓存中，本地 DNS 服务器就不会向 example. com 查询。

(2)Guess 约束：在本地 DNS 服务器向 example. com 的 DNS 服务器请求后，如果伪造的响应包没有猜中相应的 Transaction ID，本地 DNS 服务器就会丢弃该伪造响应包。

(3)Window Time 约束：如果真实响应包比伪造响应包早到达，本地 DNS 服务器会接受真实查询的结果并缓存下来。

4 Kaminsky 攻击检测

Kaminsky 攻击一旦成功，在污染记录有效期内请求该域名的所有子域都会被引到伪造的权威域名服务器解析地址，如在有效期内请求 example. com 下的所有域名都会直接向 attacker. com 请求解析。因此，本文从攻击流量行为特征出发，研究 Kaminsky 攻击的异常行为分析方法，以期在攻击初始阶段就能迅速发现异常，及时补救。

4.1 攻击特征

通过比较正常 DNS 报文和 Kaminsky 攻击时的报文，发现攻击过程除了伪造响应报文外，其他过程与正常情况下类似，总结出如下特征。

(1)请求应答报文不平衡：攻击时为了猜解 Transaction ID 会发送大量的伪造响应报文。而在正常且网络状况良好的情况下，在一定时间区间内请求报文和应答报文几乎是成对出现的。

(2)固定源目 IP 地址的 Transaction ID 信息熵大：为了猜解正确，例如猜解 Transaction ID，攻击者会尝试不同可能的 Transaction ID，这就造成在固定源 IP 地址和目的 IP 地址之间 Transaction ID 值的多样化。而正常情况下，由于存在 TTL，一定时间区间内固定源 IP 地址和目的 IP 地址之间不会出现频繁的请求或应答，Transaction ID 比较单一。

(3)网络流量增大：为了赶在真实应答报文之前猜中，伪造应答报文的数量大。

4.2 异常行为分析

本文的异常行为分析方法分为两大部分：时间序列分析和 IP 溯源。时间序列分析用于检测出攻击时间。在时间序列分析的基础上，对检测出的可疑时间范围内的数据计算进行 IP 地址溯源。

给定时间区间 T，在滑动窗口 m 内分析 DNS 报文，研究滑动窗口内报文特征随窗口移动的变化规律。

滑动时间窗口检测如下：

(1)取滑动窗口$[t, t+m]$时间范围内的数据，计算测量值；

(2)将计算出来的测量值作为 $t+m/2$ 时刻的测量值。

4.2.1　时间序列分析

(1)Transaction ID 去重

由于同一次请求应答报文具有相同的 Transaction ID 值，因此字段 Transaction ID 可以反映 DNS 请求和响应报文的对应关系。滑动窗口内将相同 Transaction ID 的样本去重，就是将滑动窗口内请求响应相对应的数据包去掉，留下的是不成对的请求和响应数据包，这样能降低正常流量的干扰，以提高检测率并减少后续步骤的计算量。

(2)经验条件熵

熵用来表示随机变量的不确定性，后来被香农(C. E. Shannon)引入信息论中。信息熵用于表示信息的不确定性[11-16]。随机变量 X 的信息熵被定义为：

$$H(X)=-\sum_{i=1}^{n}p_i\log_2 p_i \tag{1}$$

其中，p_i 表示第 i 个实例 X 出现的概率，n 表示特征 X 出现的总实例数。

假设随机变量 X 和 Y 的联合概率密度为：

$$P(X=x_i,Y=y_j)=p_{ij};i=1,2,\cdots,n;j=1,2,\cdots,m \tag{2}$$

条件熵 $H(Y|X)$ 表示在随机变量 X 给定的条件下随机变量 Y 的不确定性[17]，其计算公式如下：

$$H(Y|X)=\sum_{i=1}^{n}p_iH(Y|X=x_i) \tag{3}$$

其中，$p_i=P(X=x_i);i=1,2,\cdots,n$。

由于猜解需要尝试不同的 Transaction ID，因此 Transaction ID 的不确定程度增大，攻击时刻 Transaction ID 的信息熵增大。然而，正常流量中如果请求解析多个域名，触发多个会话，Transaction ID 的信息熵也会增大。正常流量 Transaction ID 的信息熵增大，会话数也会增多。攻击过程中源目 IP 地址相同的 Transaction ID 的信息熵增大，考虑用 IP 地址给定的条件下 Transaction ID 的熵作为测度以减小误报率。用数据估计概率，可以求得时间窗口内 Transaction ID 在 IP 地址给定条件下的经验条件熵。

(3)改进的 CUSUM 算法

CUSUM(Cumulative Sum)算法常用于检测随机序列参数的突变点。CUSUM 算法计算随机序列比正常情况高的累积值，通过累积值可以检测连续异常[18]。

设随机序列$\{H_n\}$表示时间窗口 m 内 Transaction ID 在 IP 地址给定条件下的经验条件熵。正常情况下，设 $E(H_n)=\alpha$。令 $Z_n=H_n-\alpha-\beta$，其中 β 是大于零的常数，使 Z_n 在正常情况下为负。攻击时，Z_n 突然增大且为正，当累积和超过阈值时判定为攻击。

递归公式为：

$$\begin{cases}y_i=(y_{i-1}+Z_i)^+, & i\leqslant N\\ y_i=(y_{i-1}+Z_i-(Z_{i-N})^+)^+, & i>N\end{cases} \tag{4}$$

判决函数为：

$$d(y_i)=\begin{cases}1, & y_i>D\\ 0, & y_i\leqslant D\end{cases} \tag{5}$$

CUSUM 算法能及时发现攻击开始的时间点，但由于累积和不断增大，即使攻击结束了也不能及时判断攻击结束，因此对 CUSUM 算法进行改进。

改进的 CUSUM 算法在触发异常后清空累积值，即触发事件后设置：

$$y_i=0,d(y_i)=1 \tag{6}$$

4.2.2　IP 溯源

投毒过程中需要向本地域名服务器发送大量 DNS 报文来猜解 Transaction ID，在单一源 IP 地址且 IP 地址真实的情况下，该 IP 地址关于 Transaction ID 的条件熵远大于正常值。因此，对检测出的可疑攻击时间进行回溯分析，选取触发事件时间内 IP 的条件熵，对该时间段内各 IP 条件熵求和，排名靠前的 IP 地址极有可能是投毒目标域名权威域名服务器的 IP 地址。

4.3　检测评估参数

用检测率、误检率、漏报率作为检测效果的评估指标。

检测率(Detection Rate，DR)：

$$R_d=\frac{TP}{FN+TP} \tag{7}$$

误检率(False Positive Rate，FPR)：

$$R_{f+}=\frac{FP}{FP+TN} \tag{8}$$

漏报率(False Negative Rate，FNR)：

$$R_{f-}=\frac{FN}{FN+TP} \tag{9}$$

4.4　时间复杂度分析

假设滑动窗口$[t,t+m]$内有 n 条样本，去重的时间复杂度为 $O(n)$。设去重样本数为 d，则去重后的样本数为 $n-d$。对去重后的样本计算条件熵，设按照一定条件分别计算条件熵，依照条件分为 I 类，第 i 类条件有j_i条样本，j_i 条样本中 Transaction ID 有J_i类。计算条件熵执行 $n-d+I+\sum_{i=1}^{I}(j_i+J_i)$次，时间复杂度为 $O(2n-2d+I+\sum_{i=1}^{I}J_i)$。滑动窗口$[t,t+m]$内 n 条样本的总体时间复杂度为 $O(n)+O(2n-2d+I+\sum_{i=1}^{I}J_i)$。

如果不去重而直接计算条件熵，时间复杂度为 $O(2n+I+\sum_{i=1}^{I}J_i)$。当 $d>\frac{n}{2}$时，去重再计算条件熵的时间复杂度较小；当 $d<\frac{n}{2}$时，直接计算条件熵的时间复杂度较小。在流量数据中，一段时间内的请求应答报文几乎是成对出现的，多数符合$d>\frac{n}{2}$的情况，所以去重再计算条件熵的时间复杂度较小。

5　实验及结果分析

5.1　实验环境

为了得到攻击流量数据，搭建了本地域名服务器和两个客户端，两个客户端分别是正常用户和攻击者。假设攻击者通过前期信息收集等手段猜测到 IP 地址和端口号，攻击仅需要猜测 Transaction ID。根据攻击原理，用 C 语言实现 DNS 远程投毒攻击程序[19-20]，攻击者冒充正常用户向本地域名服务器请求解析主机名不存在的域名(如 abddc. example. com)，随后攻击者假冒成权威域名服务器(如 example. com)向本地域名服务器发送构造了 NS 记录的伪造应答报文。在搭建好的环境内模拟 Kaminsky 攻击和正常 DNS 请求，用 Wireshark 抓取本地域名服务器数据包。

为了最大限度地增加攻击成功率，统计攻击报文中的真实响应报文回复所用时间，如表 1 所列。当设置时间间隔为

0.20 s 时，样本中超过 75% 的真实请求响应报文时间间隔等 0.02 s 长。假设正常用户请求本地域名服务器报文发出到本地域名服务器请求权威域名服务器报文发出时间忽略不计，则相同网络情况下时间间隔设置为 0.20 s 时只有 25% 的真实响应报文比伪造报文更早到达本地域名服务器，取攻击的请求和响应时间间隔为 0.20 s 以内。

表 1 真实请求响应报文时间间隔统计

Table 1 Interval statistics of real request response packets

（单位：s）

mean	*std*	*min*	25%	50%	75%	max
0.21	0.03	0.19	0.20	0.21	0.21	1.94

5.2 检测步骤

(1)数据预处理

由于实验中会混入其他协议数据包，我们过滤出 DNS 数据包。对 DNS 数据包提取特征，提取的特征有时间、源 IP 地址、目的 IP 地址、源端口号、目的端口号、DNS Flags 和 DNS Transaction ID 等。

(2)制作样本

将正常流量和攻击流量混合并标记攻击时间段。

(3)滑动时间窗口

在混合流量上选取滑动时间窗口 m 内的流量进行分析。

(4)Transaction ID 去重

选取固定长短的时间段 m，如[0，6 s]。在时间段 m 内进行数据包特征分析。之后将该时间段 m 向后移，继续滑动时间窗口。如向后移的步长为 1 s，则下一个时间窗口为[1 s，7 s]。对有相同 Transaction ID 和源目 IP 地址的样本进行去重。

(5)计算条件熵

计算 Transaction ID 去重后滑动时间窗口 m 内 IP 地址给定的条件下 Transaction ID 的条件熵。

(6)改进的 CUSUM 算法检测

对计算出的条件熵序列，用改进的 CUSUM 算法判定攻击时间。

(7)IP 地址溯源

对检测出的攻击时间内的数据溯源，取条件熵前 10 的 IP 地址进行分析。

(8)用检测率、误检率、漏报率评估检测效果。

5.3 检测效果

(1)确定正常行为基线

采用出口流量数据作为正常流量数据，日数据量在 2 TB 左右。取近 10 min 的 DNS 数据包，过滤出 DNS 报文。设定时间窗口大小为 6 s，每次向后滑动 1 s。正常流量的条件熵统计如表 2 所列，异常流量的条件熵统计如表 3 所列。CUSUM 算法中，$E(H_n)=\alpha$，正常流量条件熵均值为 0.05，故取 $\alpha=0.05$。正常情况下 $Z_n=H_n-\alpha-\beta$ 为负，即正常情况下 $\beta>H_n-\alpha$，取 $\beta=0.35$。

表 2 正常流量条件熵统计

Table 2 Normal flow conditional entropy statistics

（单位：bit）

mean	*std*	*min*	25%	50%	75%	max
0.05	0.10	0	0	0	0.00	0.50

表 3 异常流量条件熵统计

Table 3 Abnormal flow conditional entropy statistics

（单位：bit）

mean	*std*	*min*	25%	50%	75%	max
4.80	0.41	3.28	4.55	4.91	5.09	6.35

(2)确定参数

为了确定 CUSUM 中的 β 和 D，统计正常流量和异常流量的条件熵，如表 2 和表 3 所列，可以看出 β 和 D 的合理取值范围为[0，4]。

选取 AUC 为衡量分类器好坏的指标，实验结果如图 4 所示。通过比较不同 β 和 D 取值的分类效果，取 $\beta=0.35$，$D=1$。

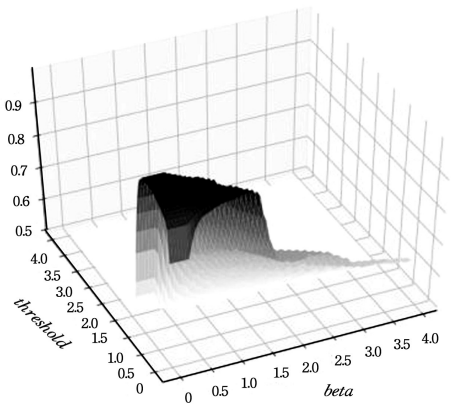

图 4 不同 β 和 D 取值下 AUC 值

Fig. 4 AUC values with different beta and threshold values

(3)对比实验

本文算法有几个关键步骤：

1)相同 Transaction ID 去重；

2)条件熵计算；

3)CUSUM 算法触发异常后清空累计值。

记仅没有步骤 1)的算法为 Transaction ID 不去重算法。记仅没有步骤 2)，计算 Transaction ID 信息熵的算法为普通熵算法。记仅没有步骤 3)，用普通 CUSUM 的算法为普通 CUSUM 算法。记文献[5]的算法为 LD 算法，即计算 FQDN 的 Levenshtein 距离并设定阈值检测攻击时间。分别用本文算法、Transaction ID 不去重算法、普通熵算法、普通 CUSUM 和 LD 算法检测攻击时间。

攻击代码参数 *count* 表示每轮伪造的响应报文个数，*sleep* 表示每轮伪造的请求报文和响应报文的时间间隔。*count*=100，*sleep*=0.01 代表高强度攻击模式；*count*=50，*sleep*=0.1 代表中低强度攻击模式。

通过设置不同的 *count* 和 *sleep* 参数模拟不同的攻击模式，检测效果如表 4 和表 5 所列。

表 4 *count*=100，*sleep*=0.01

Table 4 *count*=100，*sleep*=0.01

	time/s	R_{f+}/%	R_{f-}/%	R_d/%
our method	26.87	3.39	0.00	100.00
Transaction ID not deduplicated	1 706.47	31.02	0.00	100.00
entropy	14.91	97.29	0.00	100.00
CUSUM	23.85	54.07	0.00	100.00
LD	4 665.68	40.31	0.00	100.00

表 5　$count=50, sleep=0.1$

Table 5　$count=50, sleep=0.1$

	$time$/s	R_{f+}/%	R_{f-}/%	R_d/%
our method	30.13	19.81	5.26	94.74
Transaction ID not deduplicated	2115.04	42.53	5.26	94.74
entropy	24.26	99.54	0.24	99.76
CUSUM	31.65	99.76	0.24	99.76
LD	6620.52	52.76	0.00	100.00

可以发现，如果 Transaction ID 不去重，算法用时较多，因为时间窗口内 Transaction ID 去重可以极大地减少后续计算量。如果不使用条件熵，用普通熵做检测，误检率较高，因为正常流量中可能出现短时间内发起多个会话请求多个域名的情况，这类情况会误报。如果用普通 CUSUM 算法，触发事件后可能因为累计值过高持续超过阈值，即攻击结束后可能持续触发报警，增加了误检率。而 LD 算法需要对相邻 FQDN 计算 Levenshtein 距离，耗时较多且误检率高。

综合比较本文算法和其他 4 种算法在不同攻击模式下的检测效果，本文算法用时较少且具有较低的误检率、漏报率和较高的检测率。

(4)IP 地址溯源

选取触发事件时间内 IP 的条件熵，对该时间段内各个 IP 条件熵求和，取排名前 10 的 IP 地址。通过分析发现，排名第一的 IP 地址即为投毒目标域名的权威域名服务器 IP 地址。

结束语　Kaminsky 攻击是远程 DNS 缓存投毒攻击的一种，相比传统远程 DNS 投毒攻击，Kaminsky 攻击能在更短的时间内攻击成功，攻击成功后污染权威域名服务器的记录，导致对该域名服务器接管的所有主机的请求都被发送到攻击者控制的域名服务器上，因此及时准确地检测出攻击时间具有重要意义。本文详细介绍了 Kaminsky 攻击的攻击原理，通过提取攻击数据包特征，在滑动时间窗口上去重并计算条件熵得到观测序列，用改进的 CUSUM 算法判定攻击发生时间并追溯到投毒目标权威服务器地址。通过实验对比验证了本文方法的检测效果，其能检测出淹没在海量正常流量中的攻击流量并找出投毒目标域名的权威服务器地址。

本文提出的方法也具有一定的局限性：算法的参数需要根据网络情况调整；仅仅考虑了单一源 IP 地址进行 DNS 投毒攻击的情况，未来可以考虑计算相同 DNS 请求关键字的 Transaction ID 条件信息熵，并对 DNS 请求关键字溯源。

要从根本上保证 DNS 的安全，需要部署并开启 DNSSEC。但现实情况下，由于需要更多的系统、网络资源和后期的升级维护没有全面部署，因此 DNSSEC 的安全性和实用性研究将是下一步的工作方向。

参 考 文 献

[1] JIN C, HAO Z Y, WU Z G. Principle and Defense Strategy of DNS Cache Poisoning Attack [J]. China Communications, 2009, 6(4): 17-22, 75-81.

[2] LARSEN M, GONT F. Transport Protocol Port Randomization Recommendations: RFC 6056[S]. 2010.

[3] DAGON D, ANTONAKAKIS M, VIXIE P. Increased DNS Forgery Resistance Through 0x20-Bit Encoding[C]// Proceedings of ACM CCS'08. ACM Press, 2008.

[4] JU Y W, SONG K H, LEE E J, et al. Cache Poisoning Detection Method for Improving Security of Recursive DNS[C]// The 9th International Conference on Advanced Communication Technology. Okamoto, Kobe, 2007: 1961-1965.

[5] MUSASHI Y, KUMAGAI M, KUBOTA S, et al. Detection of Kaminsky DNS Cache Poisoning Attack[C]// 2011 4th International Conference on Intelligent Networks and Intelligent Systems. Kunming, 2011: 121-124.

[6] JIN Y, TOMOISHI M, MATSUURA S. A Detection Method Against DNS Cache Poisoning Attacks Using Machine Learning Techniques: Work in Progress[C]// 2019 IEEE 18th International Symposium on Network Computing and Applications (NCA). Cambridge, MA, USA, 2019: 1-3.

[7] WANG P J. Design and implementation of a private DNS-oriented attack detection and response system [D]. Harbin: Harbin Institute of Technology, 2018.

[8] Internet Governance Landscape Background Paper[EB/OL]. (2010-08-11). http://www. intgovforum. org/cms/2010/Background/Chinese-IGF-Background-Paper. pdf.

[9] DU W L. Remote DNS Cache Poisoning Attack Lab[EB/OL]. (2016-12-11). https://seedsecuritylabs. org/Labs_16. 04/PDF/DNS_Remote. pdf.

[10] XU C X, HU R G, SHI F, et al. Research on defense strategy of cache poisoning in Kaminsky domain name system[J]. Computer Engineering, 2013, 39(1): 12-17.

[11] ZHANG W X, WU W Z, LIANG J Y, et al. Rough Set Theory and Method [M]. Beijing: Science Press, 2001.

[12] KANDA Y, FONTUGNE R, FUKUDA K, et al. ADMIRE: Anomaly detection method using entropy-based PCA with three-step sketches[J]. Computer Communications, 2013, 36(5): 575-588.

[13] TELLENBACH B, BURKHART M, SCHATZMANN D, et al. Accurate network anomaly classification with generalized entropy metrics[J]. Computer Networks, 2011, 55(15): 3485-3502.

[14] LEE W, DONG X. Information-theoretic measures for anomaly detection[C]// Proc. of IEEE Symposium on Security and Privacy (S&P). Oakland: CA, 2001.

[15] MANIKOPOULOS C, PAPAVASSILIOU S. Network intrusion and fault detection: a statistical anomaly approach [J]. IEEE Communications Magazine, 2002, 40(10): 76-82.

[16] LAKHINA A, CROVELLA M, DIOT C. Mining anomalies using traffic feature distributions [C]// Proc. of ACM SIGCOMM. Philadelphia: PA, 2005.

[17] SHU Y Z, MEI M Y, HUANG W Q, et al. Research on DDoS Attack Detection Based on Conditional Entropy in SDN Environment [J]. Wireless Internet Technology, 2016(5): 75-76.

[18] SUN Z X, LI Q D. DDoS Attack Prevention Strategies for Databases Based on Source and Destination IP Addresses [J]. Journal of Software, 2007(10): 2613-2623.

[19] PETR E. An Analysis of the DNS Cache Poisoning Attack[EB/OL]. (2009-11-02). http://labs. nic. cz/files/labs/ DNS-cache-poisoning-attack-analysis. pdf.

[20] WANG G. Research on Security of Domain Name System [D]. Harbin: Harbin Institute of Technology, 2007.

CHEN Xi, born in 1994, postgraduate. Her main research interests include network security, anomaly detection and behavior analysis.