基于 DoH 流量的 DGA 识别方法

张千帆,郭晓军,周鹏举
(西藏民族大学 信息工程学院,陕西 咸阳 712000)

摘 要:现有研究表明,域名生成算法(domain generation algorithm,DGA)已成为僵尸网络建立命令和控制服务通信的关键技术之一。由于利用 DGA 域名随机性的检测方法已趋于成熟,为逃避检测,DGA 算法可能采用加密流量形式进行传输。针对基于域名随机性的检测模型缺乏对加密 DGA 流量的识别等问题,该文基于 DoH(DNS-over-HTTPS)协议验证了 DGA 流量进行加密传输的可能性,分析了命令控制服务过程所产生的 HTTP 报文内容、HTTP 流量及对应的 TCP 流量。因利用 DoH 协议进行传输的数据包中不再包含 DNS 报文解析过程,最终选取了 DoH 流量数据包的长度和时序信息等特征进行识别。在 DoH 网络中 DGA 流量特征分析的基础上结合 KNN 分类算法识别 DGA 域名,设计了一种基于特征工程与机器学习结合的识别方法,提供了 DoH 网络中 DGA 流量的检测方法。实验结果表明,基于 DoH 流量的 DGA 分类模型在人工数据集上的准确率达到了 79%,表现出良好的分类精度,为 DoH 网络安全提供了保障。

关键词:僵尸网络;命令控制服务;域名生成算法;DNS-over-HTTPS/DoH 协议;网络流量分析
中图分类号:TP309.5 **文献标识码**:A **文章编号**:1673-629X(2021)12-0122-06
doi:10.3969/j.issn.1673-629X.2021.12.021

DGA Identification Method Based on DoH Traffic

ZHANG Qian-fan,GUO Xiao-jun,ZHOU Peng-ju
(School of Information Engineering,Xizang Minzu University,Xianyang 712000,China)

Abstract:Current research reveals that domain generation algorithm (DGA) has become one of the key technologies for Botnets to connect to C&C (command and control) servers. Since the detection method for the randomness of DGA domain name has become mature,the DGA algorithm may adopt the form of encrypted traffic transmission bypassing the detection mechanisms. In view of the lack of recognition of encrypted DGA traffic based on the randomness of the domain name detection model,we verify the possibility of encrypted transmission of DGA traffic based on the DoH (DNS-over-HTTPS) protocol,analyze HTTP message content,HTTP traffic and corresponding TCP traffic generated during the command and control server transmission process. Because the data packets transmission with the DoH protocol no longer contains the DNS message parsing process,the length and timing information of the DoH traffic data packets are finally selected for identification. Based on the analysis of DGA traffic characteristics in the DoH network,the KNN classification algorithm is used to identify DGA domain names,a recognition method based on the combination of feature engineering and machine learning is designed to provide a detection method for DGA traffic in the DoH network. Experiment shows that the accuracy of DGA recognition model based on DoH traffic on artificial data sets reaches 79%,showing ideal classification accuracy,which provides a guarantee for DoH network.

Key words:Botnet;command & control/C&C server;domain generation algorithm;DNS-over-HTTPS/DoH protocol;network traffic analysis

0 引 言

近日,腾讯安全应急响应中心报道了一款专门针对容器虚拟化服务的僵尸网络:BORG 僵尸网络,证实该攻击者在入侵内网后开始对内网中其他机器进行探测和入侵,并批量控制这些机器[1]。由此可见,未来僵尸网络的研究与检测工作仍然严峻。僵尸网络作为一个高度可控的网络攻击平台已对互联网安全造成了极大的威胁[2]。

现有研究表明,使用 DGA 算法已成为僵尸网络逃避检测的主流方法之一,如 Cryptolocker[3]、

收稿日期:2021-01-17 **修回日期**:2021-05-18
基金项目:西藏自治区自然科学基金项目(XZ2019ZRG-36(Z));西藏民族大学“藏秦喜马拉雅人才发展支持计划-杰出青年学者”项目(324011810216);西藏民族大学“涉藏网络信息内容与数据安全团队”项目(324042000709)
作者简介:张千帆(1997-),女,硕士研究生,CCF 会员(F6251G),研究方向为网络空间安全;郭晓军,博士,副教授,CCF 会员(17584M),研究方向为网络安全、网络测量;通信作者;周鹏举(1995-),男,硕士研究生,研究方向为 Web 安全。

Hesperbot[4]、Ramnit[5]等。目前针对 DGA 域名的识别方法有:逆向工程技术[6-8],即根据 DGA 表征或 DNS 行为找到 DGA 背后的家族和变种。Yadav S. 等人[9]则通过测量了同一 IP 下域名的 K-L 距离、编辑距离和 Jaccard 系数,根据差异构建分类器,提出了基于域名距离度量的检测方法。Zhao H 等人[10]利用 N-Gram 模型,将二级域名和三级域名字符串的长度分成 3、4、5、6 和 7,根据域名权重和阈值进行判断。Zhang Y 等人[11]结合 CNN 和 BLSTM 的混合深度神经网络来提取域名的语义特征和上下文依赖特征。Hao S 等人[12]分析合法域名与恶意域名注册时 3 大类共 22 种行为特征,根据 CPM(凸多面体机)算法学习后进行检测,对是否存在恶意域名进行预判。Vissers T 等人[13]对 10 000 个域名服务器域的配置问题和硬件错误进行分析,找出了现有滥用和脆弱的域名服务器。

但上述方法大多数是面向 DGA 算法所产生明文形式流量的检测方法,僵尸网络控制者为逃避检测,极有可能对 DGA 算法产生的流量进行加密传输,例如使用 DoH 加密协议[14]。因此,该文利用 doh-proxy 代理方式[15],将 DGA 算法产生的流量进行加密传输,并分析 DoH 网络中请求传输特征和加密 DGA 流量特征,最终根据 DoH 流量数据包的长度和时序信息等特征进行识别。由此构建基于 KNN 算法的加密 DGA 流量识别方法 DGA-DoH(DGA identification method based on DoH traffic)。

首先,阐述了 DoH 加密协议工作机制,验证了 DGA 流量利用 DoH 协议建立命令控制服务传输的可能性并详细描述了 DoH 网络中恶意流量与正常流量的差异性;其次,主要结合特征工程和 KNN 机器学习算法设计了一种在 DoH 网络中识别 DGA 流量的分类模型;再次,利用实验验证了模型的识别能力并通过比较不同模型参数,优化了模型的检测率,增加了模型的泛化能力;最后,对全文内容进行总结。

1　DGA-DoH 特征分析

1.1　DoH 加密协议

DoH 加密协议是一种通过 HTTPS 运行加密 DNS 的方法,DNS 交互信息采用 Base32/Base64 等编码方式进行传输,这种协议可以在其他加密机制可能被阻止时自由地穿越防火墙[16]。DoH 网络数据查询流程见图 1,工作站通过 TLS/TCP 443 端口将数据传送给内部 Web 代理服务器,Web 代理服务器接受数据并通过防火墙传输给公共 DoH 服务器,公共 DoH 服务器通过 TCP 443 端口返回应答,同时公共 DoH 服务器受域名的权威名称服务器组件 UDP 53 端口监听。

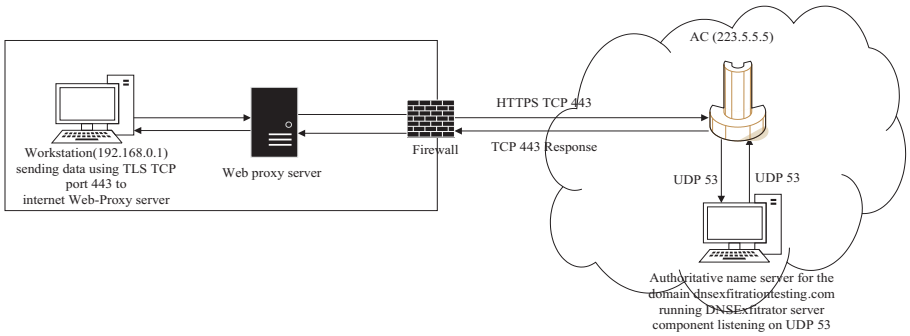
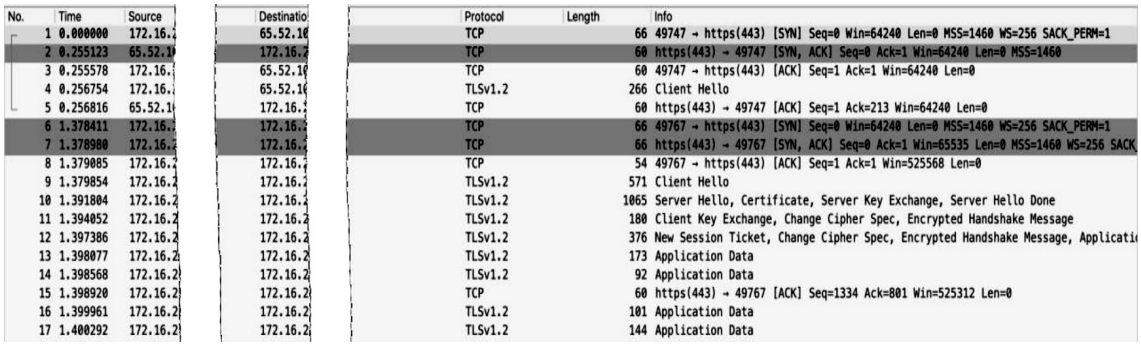


图 1　DoH 网络数据请求流程

该文采用 doh-proxy 代理方式[15],创建 DoH 网络环境。在 HTTPS 网络中发送 Web 请求后,用 Wireshark 软件抓取底层网关流量,结果如图 2 所示;同理用 Wireshark 软件记录 DoH 网络中数据请求过程,结果如图 3 所示。结果对比可得到,在图 2 基于 HTTPS 的 Web 请求过程中包含图中方框中的 DNS 报文解析过程,而图 3 中的 DoH 网络数据请求过程中因 DoH 协议加密传输的特性,传输过程中不再包含 DNS 报文解析过程。

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.17.151	211.137.13	DNS	73	Standard query 0x98d2 A www.baidu.com
2	0.000549	172.17.151	www.a.shif	TCP	78	63843 → https(443) [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1423
3	0.010244	211.137.13	172.17.151	DNS	132	Standard query response 0x98d2 A www.baidu.com CNAME www.a.shifen.com A
4	0.033955	www.a.shi	172.17.151	TCP	78	https(443) → 63843 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1440 WS=32
5	0.034013	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=1 Ack=1 Win=262144 Len=0
6	0.034352	172.17.15	www.a.shif	TLSv1.2	571	Client Hello
7	0.061185	www.a.shi	172.17.151	TCP	60	https(443) → 63843 [ACK] Seq=1 Ack=518 Win=30208 Len=0
8	0.061972	www.a.shi	172.17.151	TLSv1.2	150	Server Hello
9	0.062037	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=518 Ack=97 Win=262016 Len=0
10	0.062168	www.a.shi	172.17.151	TCP	1494	https(443) → 63843 [ACK] Seq=97 Ack=518 Win=30208 Len=1440 [TCP segment
11	0.062441	www.a.shi	172.17.151	TCP	1494	https(443) → 63843 [ACK] Seq=1537 Ack=518 Win=30208 Len=1440 [TCP segmen
12	0.062510	www.a.shi	172.17.151	TLSv1.2	935	Certificate
13	0.062520	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=518 Ack=2977 Win=260672 Len=0
14	0.062551	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=518 Ack=3858 Win=259776 Len=0
15	0.068705	www.a.shi	172.17.151	TLSv1.2	392	Server Key Exchange
16	0.068749	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=518 Ack=4196 Win=261760 Len=0
17	0.068869	www.a.shi	172.17.151	TLSv1.2	63	Server Hello Done
18	0.068896	172.17.15	www.a.shif	TCP	54	63843 → https(443) [ACK] Seq=518 Ack=4205 Win=262080 Len=0
19	0.069283	172.17.15	www.a.shif	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message

图 2　基于 HTTPS 的 Web 请求过程



No.	Time	Source	Destinatio	Protocol	Length	Info
1	0.000000	172.16.2	65.52.10	TCP	66	49747 → https(443) [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
2	0.255123	65.52.1	172.16.2	TCP	60	https(443) → 49747 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
3	0.255578	172.16.	65.52.10	TCP	60	49747 → https(443) [ACK] Seq=1 Ack=1 Win=64240 Len=0
4	0.256754	172.16.	65.52.16	TLSv1.2	266	Client Hello
5	0.256816	65.52.1	172.16.2	TCP	60	https(443) → 49747 [ACK] Seq=1 Ack=213 Win=64240 Len=0
6	1.378411	172.16.2	172.16.2	TCP	66	49767 → https(443) [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
7	1.378980	172.16.2	172.16.2	TCP	66	https(443) → 49767 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK
8	1.379085	172.16.2	172.16.2	TCP	54	49767 → https(443) [ACK] Seq=1 Ack=1 Win=525568 Len=0
9	1.379854	172.16.2	172.16.2	TLSv1.2	571	Client Hello
10	1.391804	172.16.2	172.16.2	TLSv1.2	1065	Server Hello, Certificate, Server Key Exchange, Server Hello Done
11	1.394052	172.16.2	172.16.2	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
12	1.397386	172.16.2	172.16.2	TLSv1.2	376	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message, Applicati
13	1.398077	172.16.2	172.16.2	TLSv1.2	173	Application Data
14	1.398568	172.16.2	172.16.2	TLSv1.2	92	Application Data
15	1.398920	172.16.2	172.16.2	TCP	60	https(443) → 49767 [ACK] Seq=1334 Ack=801 Win=525312 Len=0
16	1.399961	172.16.2	172.16.2	TLSv1.2	101	Application Data
17	1.400292	172.16.2	172.16.2	TLSv1.2	144	Application Data

图 3 DoH 数据请求过程

这种数据传输方式无疑为 DGA 流量逃脱检测提供了便利。现有加密传输工具所采用的编码方式各有异同,考虑 DoH 协议对数据编码方式及加密特性,对数据包解密后分析难度较大,同时为了保护数据包的完整性及用户的隐私性,该文不采用解密流量的方法[17]识别加密的 DGA 流量。

经分析建立命令控制服务所产生的 HTTP 报文内容、HTTP 流量及对应的 TCP 流量等信息后,该文将根据 DoH 流量数据包的长度和时序信息等特征识别加密的 DGA 流量。

1.2 DGA-DoH 特征分析

由于网络流特征分布会随时间和网络变化而变化,单一特征选择方法在给定数据集获得的特征子集无法在未来长时间维持稳定的分类精度[18]。因此,在已得到加密流量数据包中对所有流量进行分析。经分析与研究流量数据包中高维数据后,为减少建立与优化模型的时间等成本,仅选择表 1 中的三个最优特征子集,包括 Init_win_bytes(C->S)7th packet、Init_win_bytes(S->C)7th packet、Data_xmit_time,真实域名流量和 DGA 流量在该特征子集中存在明显差异。

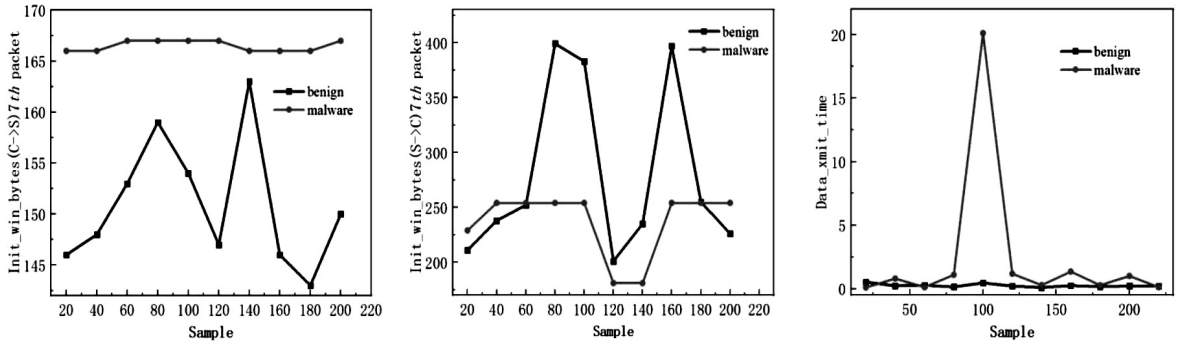
表 1 数据集的特征子集

	简称	特征描述
F_1	Init_win_bytes(C->S)7th packet	客户端到服务器端方向第七个初始窗口发送的字节数
F_2	Init_win_bytes(S->C)7th packet	服务器端到客户端方向第七个初始窗口发送的字节数
F_3	Data_xmit_time	从第一个包到最后一个非空包的传输时间

从已收集的真实域名集与 DGA 域名集生成的流量数据包中各随机选取 200 个数据包,特征对比结果如图 4 所示。

图 4(a)表示 F_1 特征对比,DGA 流量域名因有随机性、名称长度较长等特征,客户端发送给服务器端的第七个包的数据包的大小明显大于良性流量。图 4(b)表示 F_2 特征对比,因 DGA 算法生成的域名多数为无效域名,服务器难以解析,所以服务器端返回的第七个包的数据包大小明显小于良性流量数据包[19]。图 4(c)表示 F_3 特征对比,DGA 流量查询时间明显长于良性流量查询时间。

该文从初始 pacpng 数据包等高维数据中选择包含数据包长度、传输时间等最优特征子集,提高了分类模型的鲁棒性,减少了模型的生成和分类时间等成本,从而生成的模型具有更高分类效率和泛化能力。



(a) F_1: Init_win_bytes(C->S)7th packet (b) F_2: Init_win_bytes(S->C)7th packet (c) F_3: Data_xmit_time

图 4 DGA 域名流量与真实域名流量对比

2　基于 KNN 的 DGA-DoH 识别方法

2.1　算法总体设计

目前,基于 DGA 域名的检测方法已趋于成熟,但这些方法无法识别加密后流量,因此该文提出基于 KNN 模型的 DGA-DoH 识别方法,分类框图如图 5 所示,由 DoH 流量收集、特征分析、KNN 分类器部分组成。通过特征工程和机器学习紧密结合的办法解决加密恶意流量分类中的分析与检测精度不足的问题。

(1)DoH 流量收集。该文采用了 doh-proxy 代理[15],用于实现通过 HTTP 发送 DNS 查询并获得 DNS 响应时使用 HTTPS 的 URI[14]。对已收集的所有域名进行查询,得到域名数据集中每个域名在 DoH 网络中的流量,生成 DoH 数据集,并做好筛选标注及分类。为保证实验数据真实性,随机选择部分数据为实验数据。

(2)特征工程。观察和分析命令控制服务传输过程所产生的 HTTP 报文内容、HTTP 流量及对应的 TCP 流量后,根据图 4 中恶意流量与真实域名流量特征分析对比结果,该文选取 DoH 数据集中数据包的长度和时序信息等特征数据。

(3)训练模型。选取数据集中 90% 数据包为训练集,并根据提取特征进行训练。

(4)KNN 分类器。选取数据集中剩余 10% 的数据包为分类对象,对生成的分类器进行验证。

(5)模型评估。对完成分类的数据包进行检测,调整分类模型的参数,得到最优分类模型。

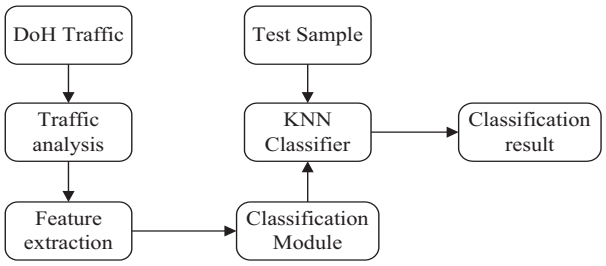


图 5　DGA-DoH 流量分类框图

2.2　DGA-DoH 数据获取

该文通过 doh-proxy 代理方式搭建 DoH 网络[15],流量传递方式如图 6 所示。DoH 网络中主要包括 DoH 客户端、Web 服务器、DoH 中转服务器、DoH 服务器等部件。该文通过在不同主机分别安装 DoH 客户端和 DoH 中转服务器,在查询过程中使用 Wireshark 软件从客户端网关监测。DoH 流量采用加密方式进行传输,因此根据 DoH 中转服务器 IP 地址对流量进行过滤,抓取客户端与中转服务器端之间的流量数据。

将截取的流量数据进行保存,并使用 Wireshark 的命令行版本及 Linux 命令行等工具对生成的 pcapng 文件进行解析,提取数据包中源 IP、目标 IP、数据包长度及传输时间等特征信息,获得所需的 DGA-DoH 数据包。

基于以上方法,保证了抓取数据是完成了 DoH 请求所产生的完整流量,增加了实验数据的真实性,有效降低了因噪声带来的实验误差。

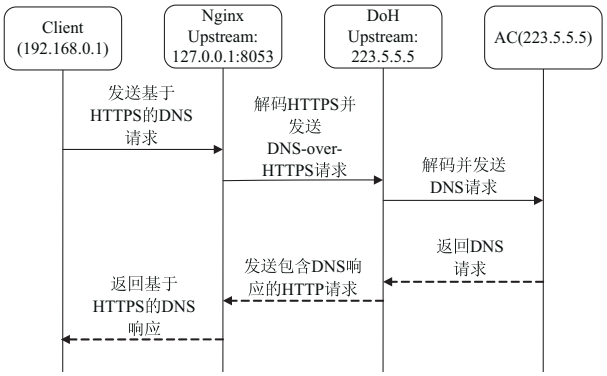


图 6　DoH 网络流量传输

2.3　基于 KNN 的 DGA-DoH 识别方法

2.3.1　数据预处理

选取表 1 中数据集的特征子集作为本次实验的数据集。将真实域名流量和 DGA 域名流量分类整理。对 DGA-DoH 流量识别,采用 KNN 算法对加密流量进行分类,对数据集中流量标签进行二值化处理,处理后标签只有 0 和 1 两种标签类型[20]。待处理的数据格式调整为分类器输入格式,并将数据分成特征向量和对应的分类标签向量两部分。

根据图 4 部分特征数值对比可知,表 1 中 F_1、F_2、F_3 三种特征的权重没有较大差异。因此该文采用数值归一化方法将任意取值范围的特征值归一化的相关公式为:

$$V_{new} = \frac{V_{old} - \mathrm{Min}}{\mathrm{Max} - \mathrm{Min}}$$

其中,Min 和 Max 分别是数据集中的最小特征值和最大特征值。

数据处理后的格式如表 2 所示。

表 2　数据归一化后的格式

Init_win_bytes (C->S)7th packet	Init_win_bytes (S->C)7th packet	Data_xmit_time
0.028 985 51	0.159 874 61	0.023 177 82
0.043 478 26	0.244 514 11	0.008 114 5
0.079 710 14	0.288 401 25	0.010 032 66

2.3.2　KNN 分类器

在训练集中,将数据标记成 DGA 流量或真实域名流量,输入测试集中未知流量后,将测试数据与训练数据集中对应的特征数据进行比较,提取训练样本集中最相似数据的分类标签,作为未知流量的分类[21]。

随机选取数据集中 90% 的数据为训练数据集,根

据欧氏距离公式在二维实数向量空间中计算当前点与已知类别点之间的距离。

$$|AB| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

将当前点与已知类别点的距离按从小到大排列，选取前 K 个点的类别，计算类别次数，返回次数最多的类别为当前点的类别。

3 实验验证

3.1 实验数据集

实验数据为 DoH 流量数据包，为保证含有更多种类的 DGA 家族，提高分类模型验证效果，DGA 域名集利用 Python 实现的 DGA 生成算法[22]；由于 DGA 域名通常具有生存周期低、访问量低等特点，难以在 Alexa 网站排名中处于 Top 位，因此真实域名采用的是 Alexa 前 100 万条域名[23]。将已收集的域名经查询、处理、解析后得到实验所需 DoH 流量数据集。表 3 包括 DoH 流量数据集中的域名种类、数据包数量等信息，数据集的 90% DoH 流量用于训练模型，剩余 10% DoH 流量用于测试生成的分类器。

表 3 数据集

类别	Packets
DGA 域名	835
真实域名	229

3.2 实验环境

该文基于 doh-proxy[15] 工具构建 DoH 网络环境，提取数据包特征后生成 DGA-DoH 数据包。将已收集的数据包分类标注后，采用 Python 语言编写 KNN 分类模型，具体实验环境如表 4 所示。

表 4 实验环境

软、硬件组成	参数
处理器	2.4 GHz 四核 Intel Core i5
内存	2 GB
操作系统	Windows 10
IDE	Jupyter
开发语言	Python 3.6

3.3 结果与分析

为了评估模型的准确率，该文参考了模型的交叉熵，交叉熵用于度量某一域名真实标签与模型分类后标签分布间的差异性信息，已知交叉熵越大，模型分类结果与真实标签越接近。随机选择数据集中 90% 数据作为训练样本来训练分类器，使用剩下的 10% 数据为实验的测试数据，检测分类器的准确率。

已知目前检测 DGA 流量的方法对 DGA-DoH 模型不具有参考价值，因此该文仅对比当前模型的检测精度，选取最优模型。为了优化模型，比较不同 K 值时模型的准确率和交叉熵，结果如表 5 所示。由表 5 可知，随 K 值的增加，模型的准确率及交叉熵呈递减趋势。因此选取与当前点距离较近的前 2 个点的类别（即 K 值为 2），返回较多点的类别为未知流量的类别，此时分类器的准确率为 79%，最大交叉熵为 9.047。

表 5 模型的交叉熵与准确率

K 值	准确率/%	交叉熵
2	79	9.047
4	74	8.696
6	72	8.556
8	64	7.995
10	61	7.785
12	62	7.855
14	61	7.785
16	61	7.785
18	60	7.714
20	58	7.574

4 结束语

该文围绕僵尸网络在新型 DoH 协议中工作机制的问题，对其构建方式及检测技术进行了相关研究，观察和分析了命令控制服务传输过程所产生的 HTTP 报文内容、HTTP 流量及对应的 TCP 流量，提取相应的特征，利用 KNN 算法对训练数据进行学习后创建分类器并采用验证数据进行验证。

通过实验分析总结了 DoH 协议中 DGA 流量特征，在未来的工作中，会继续观察 DNSCrypt、DoT 等 DNS 加密协议中恶意流量特征，提高检测精度，为 DoH、DoT 等加密 DNS 协议网络安全提供更多的保障。

参考文献：

[1] 宙斯盾流量安全分析团队 Pav1. BORG：一个快速进化的僵尸网络[EB/OL]. (2021-01-11)[2021-01-30]. https://security.tencent.com/index.php/blog/msg/175.

[2] 郭晓军. 僵尸网络新型命令控制机制及检测关键技术研究[D]. 南京：东南大学，2017.

[3] LIAO K, ZHAO Ziming, DOUPÉ A, et al. Behind closed doors: measurement and analysis of CryptoLocker ransoms in Bitcoin[C]//Electronic crime research. Toronto, ON, Canada: IEEE, 2016: 1-13.

[4] 胡鹏程，刁力力，叶 桦，等. 基于人工特征与深度特征的 DGA 域名检测算法[J]. 计算机科学，2020，47(9)：311-317.

[5] 陈立国，张跃冬，耿光刚，等. 基于 GRU 型循环神经网络的随机域名检测[J]. 计算机系统应用，2018，27(8)：198-202.

[6] SCHIAVONI S, MAGGI F, CAVALLARO L, et al. Phoenix: DGA-based botnet tracking and intelligence[C]//Detection of intrusions and malware, and vulnerability assessment. Egham, UK: Springer, 2014: 192-211.

[7] PLOHMANN D, YAKDAN K, KLATT M, et al. A comprehensive measurement study of domain generating malware[C]// 25th USENIX security symposium. Austin, TX, USA: USENIX, 2016: 263-278.

[8] BILGE L, KIRDA E, KRUEGEL C, et al. EXPOSURE: finding malicious domains using passive DNS analysis[C]//Proceedings of the network and distributed system security symposium. San Diego, California, USA: NDSS, 2011: 1-17.

[9] YADAV S, REDDY A K K, REDDY A L N, et al. Detecting algorithmically generated domain-flux attacks with DNS traffic analysis[J]. IEEE/ACM Transactions on Networking, 2012, 20(5): 1663-1677.

[10] ZHAO H, CHANG Z, BAO G, et al. Malicious domain names detection algorithm based on N-gram[J]. Journal of Computer Networks and Communications, 2019, 2019(2): 1-9.

[11] ZHANG Y, CHEN Y, LIN Y, et al. Detection of algorithmically generated domain names using SMOTE and hybrid neural network[C]//CCF conference on computer supported cooperative work and social computing. Kunming, China: Springer, 2019: 738-751.

[12] HAO S, KANTCHELIAN A, MILLER B, et al. PREDATOR: proactive recognition and elimination of domain abuse at time-of-registration[C]//Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. Vienna, Austria: ACM, 2016: 1568-1579.

[13] VISSERS T, BARRON T, VAN GOETHEM T, et al. The wolf of name street: hijacking domains through their nameservers [C]//Proceedings of the 2017 ACM SIGSAC conference on computer and communications security. Dallas, TX, USA: ACM, 2017: 957-970.

[14] HOFFMAN P, MCMANUS P. DNS queries over HTTPS(DoH), RFC 8484[S]. [s.l.]: IETF, 2018.

[15] CHANTRA. A set of python 3 scripts that supports proxying DNS over HTTPS as specified in the IETF Draft draft-ietf-doh-dns-over-https[CP/DK]. (2019-04-04)[2020-10-31]. https://pypi.org/project/doh-proxy/#doh-client.

[16] HADDON D A E, ALKHATEEB H. Investigating data exfiltration in DNS over HTTPS queries[C]//2019 IEEE 12th international conference on global security, safety and sustainability (ICGS3). London, UK: IEEE, 2019: 212.

[17] ANDERSON B, MCGREW D. Identifying encrypted malware traffic with contextual flow data[C]//Proceedings of the 2016 ACM workshop on artificial intelligence and security. Vienna, Austria: Association for Computing Machinery, 2016: 35-46.

[18] 潘吴斌. 加密流量精细化分类技术研究[D]. 南京:东南大学, 2018.

[19] 朱迦南. 基于 DNS 日志数据的异常域名检测研究[D]. 成都:电子科技大学, 2018.

[20] 周琳娜, 吕欣一. 基于 SVM 的 DGA 家族分类方法研究[J]. 中国科技论文, 2020, 15(11): 1328-1333.

[21] 王　芳. 基于类划分和近邻选取的 k 近邻算法研究[D]. 西安:西安理工大学, 2020.

[22] JOHANNES B. Domain generation algorithms (DGAs) of malware reimplemented in Python[CP/DK]. (2020-07-21)[2020-10-31]. https://github.com/baderj/domain_generation_algorithm.

[23] ALEXA. The top 500 sites on the web[CP/DK]. (2020-07-21)[2020-10-31]. https://www.alexa.com/topsites.