计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2021.20200498
58(8):1773-1788,2021

针对 Tor 的网页指纹识别研究综述

孙学良[1,2]　黄安欣[1,2]　罗夏朴[3]　谢　怡[1,2]
[1](厦门大学信息学院　福建厦门　361005)
[2](福建省智慧城市感知与计算重点实验室(厦门大学)　福建厦门　361005)
[3](香港理工大学计算机系　香港　999077)
(sunxueliang@stu.xmu.edu.cn)

Webpage Fingerprinting Identification on Tor: A Survey

Sun Xueliang[1,2], Huang Anxin[1,2], Luo Xiapu[3], and Xie Yi[1,2]
[1](*School of Informatics*, *Xiamen University*, *Xiamen*, *Fujian* 361005)
[2](*Fujian Key Laboratory of Sensing and Computing for Smart City* (*Xiamen University*), *Xiamen*, *Fujian* 361005)
[3](*Department of Computing*, *The Hong Kong Polytechnic University*, *Hong Kong* 999077)

Abstract　With the prosperous development of Web services, protecting Web-surfing privacy has become a significant concern to society. Various protection techniques (e.g., anonymous communication networks) have been proposed to help users hide the real access targets and anonymously browse the Internet. However, Webpage fingerprinting (WF) identifications, through monitoring and analyzing network traffic, can still determine whether a Web page is visited by exploiting network traffic features, thus jeopardizing the anonymity. On the other hand, law enforcement agencies can leverage the methods of WF identification to monitor anonymous networks to prevent abusing them for carrying out illegal activities or covering up crimes. Therefore, WF identification is a significant and noteworthy technique for privacy protection and network supervision. In this survey, we first introduce the concept and development of WF identifications, and then focus on two kinds of WF identifications on Tor, a widely used anonymous network, including single-tag oriented identifications and multi-tag oriented identifications. In particular, the characteristics of these WF identifications are analyzed and these WF limitations are pointed out, such as simplistic assumptions and insufficient experiments for systematical evaluation. Finally, future research directions for WF identifications are concluded.

Key words　Webpage fingerprinting identification; Tor anonymous communication; privacy preserving; traffic analysis; machine learning; network supervision

摘　要　随着 Web 服务的发展,以匿名网络为代表的互联网隐私保护技术越来越受到重视.用户可以通过匿名网络隐藏真实的访问目标,在互联网上匿名浏览网页.然而网页指纹识别仍能通过监听和分析网络流量判断出用户真实的访问目标,从而破坏匿名性.因此,网页指纹识别的方法也能对匿名网络实施监管和审查,避免不法分子滥用隐私保护技术进行非法活动或掩盖罪行.无论从隐私保护还是网络监管

收稿日期:2020-06-16;**修回日期**:2020-10-12
基金项目:国家自然科学基金项目(61771017,61671397,61772438,61972313);香港创新及科技基金项目(GHP/052/19SZ)
This work was supported by the National Natural Science Foundation of China (61771017, 61671397, 61772438, 61972313) and Hong Kong Innovation and Technology Fund Project (GHP/052/19SZ).
通信作者:谢怡(csyxie@xmu.edu.cn)

的角度来说,网页指纹识别都是值得重点关注的技术手段.在介绍网页指纹识别的概念和发展基础上,针对最有代表性的匿名系统 Tor,重点阐述和分析面向单标签和面向多标签的 2 类网页指纹识别,并讨论其工作原理、性能特点和局限性(例如过于简化的研究假设和缺乏系统的实验评估).最后总结和展望网页指纹识别的未来发展方向.

关键词 网页指纹识别;Tor 匿名通信;隐私保护;流量分析;机器学习;网络监管

中图法分类号 TP393.08

Web 服务给人们的生活工作带来极大方便,但也使得私人信息大量地暴露于网络.例如,为提升体验,电子购物、银行服务和远程医疗网站跟踪和分析用户的消费习惯、经济和健康状况,可能会带来服务商信息泄露和不法分子窃取侵害的风险.因此互联网隐私保护越来越受到重视,主要措施包括加密通信和匿名访问.

加密通信协议从最初的 SSL1.0(secure socket layer),发展到 TLS1.3(transport layer security)[1],并成功应用于 HTTPS(hyper text transfer protocol over secure socket layer)[2-3].2018 年全站 HTTPS 化后,网页访问的明文传输数据越来越少,但访问过程产生的网络流量仍会暴露使用者的意图[4].例如分析未加密的 DNS(domain name system)请求能获知用户要访问的网站;从数据包的目标 IP 地址能反向解析出域名;从 TLS 连接建立过程中的 Client Hello 和 Server Certificate 能获知明文域名信息.目前,DoT[5-6](DNS over TLS)和 DoH[7](DNS over HTTPS)协议正致力于解决 DNS 加密问题;ESNI[8]和 TLS1.3 也将加密相关域名信息.相较于协议标准的缓慢升级,自由开源的隐私保护技术更受关注,Tor[9-10],AN.ON[11],FreeNet[12],I2P[13]等匿名访问网络应运而生.用户可以通过匿名网络在互联网上进行私密浏览、匿名访问和发布信息,不但保护隐私,还能防止追踪甚至避开网络审查.

尽管匿名网络加密了所有敏感信息,通过监听和分析网络流量仍能判断用户是否访问某个网页.因为构成每个网页的 CSS(cascading style sheets)和 JavaScript 代码、图片、视频、广告等元素不完全相同,访问过程中产生的网络流量就有区别,例如数据包的大小、顺序和时间等非敏感信息可以形成网页指纹(Webpage fingerprinting, WF).将搜集到的网络流量与已知网页访问流量的指纹进行对比,就能获知用户访问的隐私信息.匿名网络技术给网页指纹识别带来许多新的挑战,已成为网络安全研究的热点.

但 Tor 等匿名网络也常常被不法分子滥用以遮盖其网络犯罪行为[14-16],严重违背了保护用户隐私和匿名性的设计初衷.若缺乏有效的监管技术,政府难以识别追踪隐藏在匿名网络中的非法地下网站,无法准确打击相关犯罪行动.因此,提高匿名网络的监管水平势在必行,而网页指纹识别的方法恰好能对匿名网络实施监管和审查.可见无论从隐私保护还是网络监管的角度来说,网页指纹识别都是需要重点关注的技术手段.为方便阐述,本综述将网页指纹的使用者①统称为攻击者.

1 简 介

在网页指纹识别中,用户的身份可由 IP 或 MAC(media access control address)地址、NAT(network address translation)记录等信息来确定;而攻击者通过监听用户访问页面的流量并分析其模式与特征,来判断用户当前访问的页面.该攻击者可以处于多个网络位置,例如与互联网服务提供商、Tor 入口的控制者以及与用户处于同一 AP 接入点的恶意攻击者.早期研究将网页指纹识别视为匹配问题[17-19],对捕获的用户流量构造特征生成网页指纹,与已知网页指纹进行匹配从而判断其归属.如今,网页指纹识别以机器学习的视角转换为分类问题[20-21].攻击者首先使用提前收集的若干目标网页(也称受监视页面)的访问流量训练分类器,然后输入被捕获的用户访问流量,可输出分类结果,即判断用户访问了某个受监视页面.根据不同划分标准,如目标网络协议、特征提取、分类器选择、协议数据单元和浏览器模式,网页指纹识别可分为不同类型,如图 1 所示.

由于加密程度较低,早期针对 HTTP 的识别[17]

① 网页指纹的实际使用者可能是审查网络环境的管理/执法人员,也可能是侵害用户隐私的不法分子.

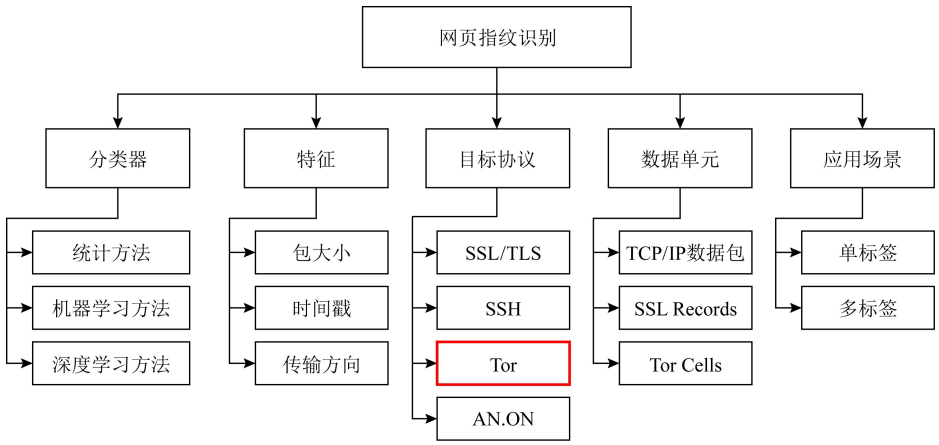

Fig. 1 The classification of Webpage fingerprinting identification

图 1 网页指纹识别的分类

比较容易从流量中提取有用信息,来确定用户访问的网页.随着 SSH(secure shell)和 TLS 加密网页的流行,研究者也有针对性地提出许多准确率高的网页指纹识别方法[21-25].由于 SSH 和 TLS 加密通信仅加密明文内容(例如通信内容),而不对流量本身的一些统计学特征进行加密(如数据包的大小和到达顺序等),因此攻击者可以将流量中独特的数据包大小与网页建立匹配关系[26],从而获知用户访问的页面.近年涌现了加密程度更高的网络,特别是以 Tor 为代表的匿名网络兴起给网页指纹识别带来新的挑战.Tor 不仅通过多个中间节点的跳转隐藏了通信双方的真实身份,还通过分割流量数据为固定长度 512 B 的 cell,并采用周期性切换 circuit 等匿名手段[9,27],消除了许多常用的流量特征,使早期的网页指纹识别失效[21].因此针对 Tor 加密流量的监管和识别成为亟待解决的重要问题.本综述将重点关注适用于 Tor 加密的匿名网页访问的网页指纹识别方法.

网页指纹识别需要从网络流量中提取特征,例如每个数据包的大小、传输时间、传输方向及到达顺序,以及一次完整的传输中所传输的字节数和数据包总数等.Panchenko 等人[28]针对 Tor 通信,分别在 Tor cell,TLS record,TCP packet 这 3 种协议数据单元中提取特征并比较识别性能.实验表明,针对 Tor 加密流量,从 Tor cell 中提取特征能达到最好的识别效果,这与文献[29]提出的数据处理优化方法结论吻合.3.4 节将对各种识别算法的特征提取和分类算法进行分析总结,然而如何提取流量特征实现准确的网页指纹识别,仍是一个开放问题.

单标签(single-tab)和多标签(multi-tab)浏览模式是以用户行为分类的基本应用场景.在单标签场景中,用户每次访问且仅访问一个页面;在多标签场景中,用户可以一次打开浏览器的多个标签,并同时访问多个页面.但单标签场景常被认为脱离实际[30],大多用户习惯一次性打开多个标签或者打开多个浏览器界面,而且在不关闭前一个页面时访问新页面[31-33].虽然现有的单标签网页指纹识别[34]已获得相当高的准确率,但往往无法直接用于多标签场景[30].因此,近期研究考虑更具实用性和挑战性的多标签情景.

现有解决思路是把多标签场景转换为多个单标签问题[35-38],然而转换过程并非简单的拆分.因为不同于单标签场景下追求高准确率,多标签网页指纹识别方法需要具备较强鲁棒性以抵抗标签间的复杂干扰.在用户使用浏览器时,多个标签页面并非同时打开(即存在时间间隔),找到这个时间间隔并对流量进行分割,就能获得多个单标签访问的流量记录,然后可以对分割好的网页访问流量实施网页指纹识别.解决多标签问题时,多标签判定、流量分割和流量分类的 3 个步骤是作为一个系统性问题来考虑,每一个前置步骤中的误差都会对后续步骤的准确率产生影响.因此多标签网页指纹的研究极具挑战,已成为相关研究的发展方向,具有重要的意义.

2 网页指纹识别的基础概念

2.1 针对 Tor 的网页指纹识别工作原理

用户通过 Tor 网络匿名浏览网页时,攻击者能够监听、标记且分离用户网页访问流量,也可以在多个网络位置进行网页指纹的识别.例如使攻击者和

用户接入同一个 AP 或位于同一段通信链路,也可以由路由器管理者或互联网服务提供商(ISP)进行部署,其工作原理如图 2 所示.攻击者不会做出丢弃、修改或插入数据包等主动行为,只是被动地监听由用户发出的网络流.由于采用了 Tor 匿名技术,攻击者无法获取接收者的信息和通信内容,因为这些敏感信息已经被加密并且无法解密获得明文.于是攻击者分析网络流量中那些不受保护且可以轻松捕获的非敏感信息,例如一系列数据包的大小、顺序和数据包之间的时间差等.

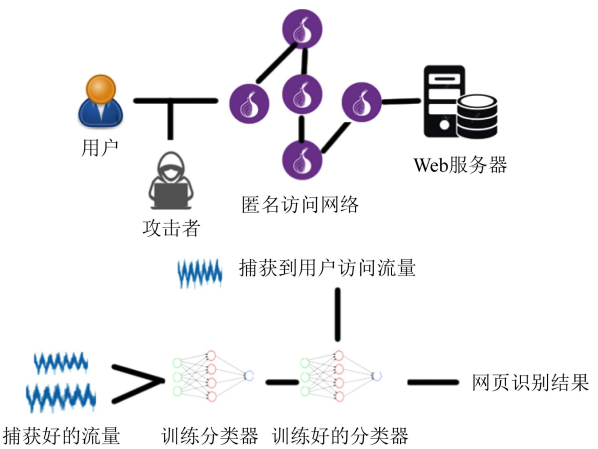

Fig. 2 The workflow of WF attack for Tor traffic

图 2 针对 Tor 的网页指纹识别工作流程

网页指纹识别假设攻击者对于用户的访问环境已经具备一定了解.因此在准备阶段,攻击者能尽可能模仿与用户一致的访问环境,例如使用相同的终端设备、操作系统、桌面环境、网络环境和 TBB(Tor browser bundle)版本配置.然后,用模仿的终端设备发起对一系列受监视页面的访问,分别记录访问流量,并从流量的非敏感信息中提取特征,训练分类器或指定匹配规则.在识别阶段,攻击者从监听到的用户网络流量中分割出网页访问流量,分析其中非敏感信息所形成的特征,并通过分类或匹配方法判断出用户所访问的网页是否属于受监视页面,以及属于哪一个受监视页面.

在 Tor 匿名网络中实现网页指纹识别是一项有挑战性的工作.首要任务是要从用户设备不断发出的网络流量中分割出网页访问流量,即流量分类.由于采用固定长度 cell 封装数据,Tor 产生的流量中数据包大小分布与众不同.针对加密流量涌现了大量优秀的流量分类算法[39],网页访问流量的分割也愈加精确,还可以判断用户是否使用了 Tor,为网页指纹识别打好了基础.接着从非敏感信息中提取有效特征,制定匹配规则或训练高效的分类器,开展网页指纹识别.本综述将分类介绍利用匹配方法、朴素贝叶斯(naive Bayes, NB)、支持向量机(support vector machine, SVM)、随机森林(random frost, RF)、最邻近算法(k-nearest neighbor, kNN),以及深度学习(deep learning, DL)方法对加密网页进行网页指纹识别的工作.

2.2 威胁场景

网页指纹识别的研究必须考虑 2 种公认的威胁场景:closed-world 和 open-world.在不同的场景,用户访问的网站列表具有不同的构成方式.

在 closed-world 场景,用户访问有限的 N 个受监视网页,而攻击者事先获知这些网站的集合.在准备阶段,攻击者对 N 个网页分别进行 X 次访问,搜集、分析并标注这些网页访问流量,用于设计匹配原则和训练分类器,一个受监视网页被视为一个类别.然后,攻击者将捕获的用户网络流量输入分类器,若能成功识别出用户访问的受监视页面(真实所属的类别),即可判定识别成功.此场景下的网页指纹识别,实际是一个经典的多分类问题,随机猜测也有 $1/N$ 的成功率.但仅考虑 closed-world 场景的工作[17-18,21-22,40]往往被认为脱离实际[30,41],因为一般用户访问的页面数量大也难以预测,无法提前简化为固定集合.

在 open-world 场景[23-29,42-45],攻击者首先选取 N 个受监视页面,并对每个网页分别进行 X 次访问;同时选取 M 个普通网页,并对每个网页进行 1 次访问(通常 $N<M$).然后将收集到的 $N\times X+M$ 组网络流量进行训练,每个受监视网页被标注为一个类别,所有普通页面被标注为同一个类别,即攻击者需要考虑 $N+1$ 个分类标签.因此,分类器不仅需要识别在准备阶段训练过的受监视页面和普通网页,还需要在识别阶段,能够将未经过训练的普通网页正确分类到普通页面类别中.

2.3 性能评估

由于网页指纹识别可以视作一个分类问题,因此机器学习的通用性能指标也适用,例如准确率(accuracy, Acc)、召回率(recall)、真阳率(true positive rate, TPR)、假阳率(false positive rate, FPR)和混淆矩阵(confusion matrix).针对 closed-world 和 open-world 威胁场景,还定义了更细致的评估方案.在 closed-world,攻击者假设用户仅会访问 N 个受监视页面,属于多分类问题.因此使用 Acc,TPR/FPR 和混淆矩阵均能有效描述识别性能.而在 open-world

中,网页指纹识别分为 2 个阶段.第 1 阶段被视作 2 分类问题,即判断一段网页访问流量是否属于普通页面类别.如果是,则判断结束;否则进入第 2 阶段,对 N 个受监视页面进行多分类处理(相当于 closed-world 问题),如图 3 所示:

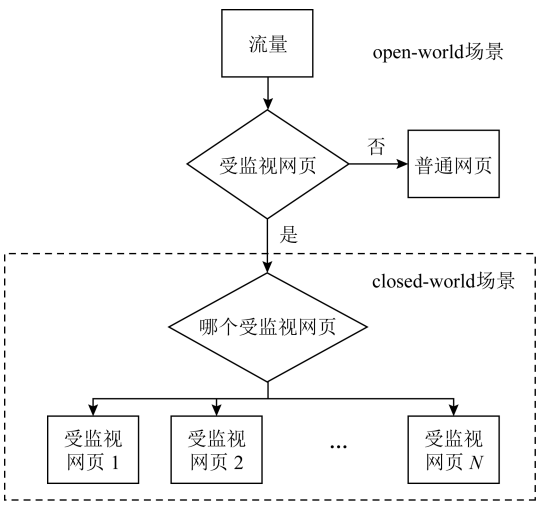

Fig. 3 The workflow of open-world problem

图 3 open-world 场景识别的流程图

因此,open-world 可以选择 2 种评价体系:Two-*[17-21,37,42,44-45]和 Multi-*[28,43],* 代表着某种具体评价指标,例如准确率.在 Two-* 指标中,如果一个受监视页面被正确分类为受监视页面,则认为是 TP(true positives),被错误分类为普通页面则认为是 FN(false negatives).如果一个普通页面被正确分类为普通页面,则认为是 TN(true negatives),否则被认为是 FP(false positives).Multi-* 指标则严格评估网页访问流量是否准确地被分成 $N+1$ 个类别.当且仅当一组访问流量被正确地判断为其真实所属的受监视页面类别时,才会被认为是分类准确,识别成功.其后出现的实验数据,默认采用 Two-* 指标;若以 Multi-* 指标做参考,将特别注明.

有些学者还指出 open-world 的网页指纹识别不能简单地视为二分类问题,而是一个介于二分类和多分类之间的问题,需要定义新的性能指标.例如,Wang[46]在 Two-* 的基础上引入了 WP(wrong positive)概念,代表一段网络流量被分类器成功判断为受监视页面,但是并不能正确找到其真实属于哪个受监视页面的情况,相关混淆矩阵定义如表 1 所示.同时提出应提高网页指纹识别的精确度来避免基率谬误(base-rate fallacy).基率谬误是指当受监视页面数量远少于普通页面时,即使分类器具有很好的性能表现,也有很大概率得到错误结论.假设受监视页面数为全部页面数的 1/1 000,分类器拥有 99% 的 TPR,1% 的 FPR,1% 的 FNR 和 99% 的 TNR.若一段网络流量被该分类器识别为受监视网页,那么根据贝叶斯定理,该判断的正确概率仅为 9%.因此本综述在评价识别效果时(见第 3 节和第 4 节),必须考虑数据集的组成.

Table 1 The Confusion Matrix Using WP[46]

表 1 引入 WP 的混淆矩阵

混淆矩阵		分类结果		
		正确分类的受监视网页	错误分类的受监视网页	普通网页
正确分类	受监视网页	TP	WP	FN
	普通网页		FP	TN

3 面向单标签的网页指纹识别

面向单标签的网页指纹识别,认为用户使用浏览器每次仅会访问一个页面,即只打开单个标签页.这类识别方法很多,本节根据所选取的匹配或分类算法进行分类介绍.由于没有公认的公开数据集,每个算法研究工作使用的数据集组成都有区别,然而无论是页面数量、流量样本数量,还是采集降噪方式都会对识别结果产生较大影响.为了保证算法分析的客观性,讨论识别效果时会说明所用数据集的参数,例如受监视页面数量 N 和普通页面数量 M、对每个页面采集流量样本的次数 X 等.例如在识别准备过程,搜集 $N=5$ 个真实受监视网页的访问流量,每个网页采集 $X=35$ 次;收集 $M=4\,000$ 个普通网页的访问流量,每个网页采集 1 次,则数据集的规模(scale)记为 $5\times35+4\,000\times1$,即 $N\times X+M\times1$.

3.1 基于统计的方法

Su-Sim[24]是一种针对 SSL/TLS 加密网页的基于统计的识别方法.它从浏览器与代理的连接中分离出网络对象(Web objects),并使用网络对象的大小和数量计算 Jaccard 系数[47],以此衡量 2 段网络流量的相关性.通过相关性匹配,能在 open-world 场景(2 191 个受监视页面和 98 496 个普通页面),获得 75% 的 TPR.其局限性在于,当使用 WEP(wired equivalent privacy)/WPA(Wi-Fi protected access)、SSH、通道(tunnel)和 VPN(virtual private network)等方式访问网页时,网络对象无法从网络流量中分离出来.

Bi-XCor[17]则根据访问流量之间的互相关系数(cross correlation)进行匹配,对每个受监视页面进行一次流量采集即可完成识别准备.在 $N=100$ 的 closed-world 中,对经过加密或通过代理的流量,仅得到 23%的准确率.由于没有考虑匿名通信网络填充数据包的情况,它将数据包大小作为特征之一,而 Tor 浏览器已经将 cell 的大小固定为 512 B.可见,基于统计的方法大多无法适用于匿名网络.

3.2 基于传统机器学习的方法

3.2.1 基于朴素贝叶斯

不同于 Bi-XCor[17],Liberatore 等人[18]不再使用基于时间信息的特征,而是使用数据包的大小和方向设计了基于 Jaccard 系数和朴素贝叶斯分类器的网页指纹识别方法.在包括 1 000 个受监视页面的 closed-world 中,一次识别可获得 73%的准确率;若允许多次识别,其准确率能提升到 90%.同时提出识别准确率与受监视网站总数对数值($\mathrm{lb}\,N$)呈线性关系,并讨论了 4 种匿名化的流量填充对识别性能的影响.例如,将数据包全部填充至最大传输单元(maximum transmission unit, MTU)时,数据包大小的特征完全消失,识别准确率骤降至 7.7%,而且开销极大.

He-MNB[21]改进了 Liberatore 等人[18]的工作.虽然同样使用数据包的大小和方向信息作为特征,但它采用多项式朴素贝叶斯算法(multinomial naive Bayes, MNB)做分类.与朴素贝叶斯算法不同,MNB 不直接计算一个特征向量归属每个类别的概率,而是使用在所有训练集上的聚合特征.在 775 个受监视页面的 closed-world 中,He-MNB 针对 Stunnel, OpenSSH, CiscoVPN, OpenVPN 的网页访问,均达到了 95%以上的识别准确率;但对匿名网络 Tor 和 AN.ON 不适用,准确率仅为 2.96%和 19.97%.

3.2.2 基于支持向量机

Pa-SVM[42]在 He-MNB[21]的基础上增加了衡量流量突发性的多个特征(如流量方向反转次数、HTML 文档大小、总传送字节数、流量方向反转期间的数据包数量、不同大小数据包出现的次数、流入的数据包比例、数据包总数和 TLS/SSL 记录的大小等),并使用 SVM 做分类器.在 775×20 的 closed-world 中,Pa-SVM 对匿名网络的性能大大提高,针对 Tor 和 AN.ON 流量,识别准确率分别提升到 55%和 80%.在 3 个模拟审查类别中(sexually explicit, Alexa top, ranked Alexa random),它最高取得了 73%的 TPR 和 0.05%的 FPR,也是在 open-world 中第一个成功实施网页指纹识别的方法.

基于 Pa-SVM[42]的手工构造的特征,Ca-OSAD[22]利用 Damerau-Levenshtein 距离[48]构建内核,建立了 SVM 分类器,进一步提升了针对 Tor 加密流量的识别效果.在 100×40 的 closed-world 下,针对 Tor 流量的识别准确率达到了 87.3%.Wang 等人[29]从数据搜集、数据处理和网页指纹分离 3 个方面对 Ca-OSAD[22]进行改进.首先提取 Tor cells 而不是 TCP/IP packets 的特征信息,然后同时使用 Damerau-Levenshtein 和 Optimal-String-Alignment 这 2 种距离表征流量之间的差异.在 100×40 的 closed-world 中,针对 Tor 流量的识别可以达到 91%的准确率;在 4×40+860×1 的 open-world 中,可获得 96.9%的 TPR,而相同数据集下 Ca-OSAD 的 TPR 为 86.9%.

Pa-CUMUL[28]更细致地分析 Tor 网页访问流量,通过统计 Tor cells 的长度和数量提取了 104 个有效特征,并使用 RBF(radial basis function)内核构建了 SVM 分类器.针对 Tor 流量,Pa-CUMUL 在 100×90 的 closed-world 中,最高取得了 92.22%的准确率.在 100×90+9 000×1 的 open-world 中,达到了 96.92%的 TPR 及 1.98%的 FPR.

3.2.3 基于随机森林

Hayes 等人分析比较已有的特征选取方案,提出 Ha-kFP[23]识别方法.它使用随机森林算法从流量记录的特征中生成具有强鲁棒性的叶子向量,最有效的特征包括:收到的 TCP/IP 数据包总数、收到和发出的数据包比例、数据包顺序统计等.这些向量被视为经过编码的网页指纹信息,然后使用 kNN 分类器计算网页指纹信息之间的距离并据此分类.针对 Tor 流量,在 100×90 的 closed-world 中,Ha-kFP 可获得了 91%的识别准确率;在 30×80+16 000×1 的 open-world 中,最高可取得了 81%的 TPR 和 0.02%的 FPR.

3.2.4 基于最邻近算法

Wa-kNN[26]使用最邻近算法作为分类器,提取了大量的特征,例如数据包的顺序、收到和发出的数据包数量及流量突发的次数等,其特征总数量接近 4 000.其中 3 000 多个特征是某个页面访问过程中传输的独特数据包大小,最终通过计算权重的方法进行了精简,因此未导致过长的处理时延.它判断一段流量样本所需的时间仅为 Ca-OSAD[22]的 1/4 500.在提取特征时,若抛弃独特数据包大小的特征,耗时

还能缩短至 1/4.Wa-kNN[26] 在 100×90 的 closed-world 中,达到了 91%的准确率,在 100×90+5 000×1 的 open-world 下,取得了 85%的 TPR 和 0.6%的 FPR.

3.3 基于深度学习的方法

Ri-DF[34] 使用深度学习领域中的 3 个经典模型:堆叠降噪自动编码器(stacked denoising autoencoder,SDAE)、卷积神经网络(convolutional neural network,CNN)及长短期记忆网络(long short term memory,LSTM)完成了对网页指纹识别的测试,其使用特征提取网络自动地提取特征,而不再通过特征工程手动构建特征.其模型输入是表示 Tor cells 方向的序列.在 100×2 500 的 closed-world 中,由 CNN 构建的模型达到了最高的识别准确率 96.26%.在 200×2 000+400 000×1 的 open-world 中,CNN 仍取得了最好的效果:80.11%的 TPR 和 10.53%的 FPR.

Si-DF[44] 使用 CNN 进行特征提取并使用全连接层作为分类网络构建了分类器.与 Ri-DF[34] 相同,Si-DF 仅使用 Tor cells 的方向作为特征.在95×800 的 closed-world 中,Si-DF 达到 98.3%的识别准确率;在 95×900+20 000×1 的 open-world 中,通过调整置信度可获得最高精确度 96%(TPR 为 68%),或最高 TPR 为 96%(准确率为 67%).这也验证了 CNN 模型的性能优势.Abe 等人[49] 也使用 SDAE 设计网页指纹识别方法,并在 Wa-kNN[26] 相同的数据集上测试.然而由于数据量不足,识别准确率为 88%,弱于 Wa-kNN.由此可见,利用深度学习进行网页指纹识别,需要依赖大量样本数据.

深度学习可以提高网页指纹识别效果,但巨大的数据量也增加了识别成本.例如 Ri-DF 和 Si-DF 识别的准备就需要花费数天来收集和训练数据[45].为此,Si-Triplet[45] 使用(n-shot learning, NSL)来减少网页指纹识别对大数据样本的依赖性.Si-Triplet 首先使用 CNN 构建特征提取网络提取特征,对流量样本记录编码,再使用 kNN 算法计算编码之间的距离,并据此分类结果完成网页指纹识别.在 closed-world 中,Si-Triplet 使用 100×25 的数据集来训练特征提取网络,并对 100 个受监视页面进行 CNN 特征提取和 kNN 分类测试,可达到 79.4%的识别准确率.而在同样条件下,基于传统机器学习的 Pa-CUMUL[28] 和 Ha-kFP[23] 识别准确率分别为 42.1%和 36.3%.在 100×5+9 000×1 的 open-world 中,使用同样的数据集进行特征提取网络的训练,再使用不同的数据集进行测试,Si-Triplet 最大精确度为 87.1%(TPR 为 80.8%).

Bh-VarCNN[43] 引入空洞卷积及残差网络构建了神经网络分类器.与 Ri-DF[34],Si-DF[44],Si-Triplet[45] 不同,Bh-VarCNN 不仅通过特征提取网络从 Tor cells 的时间和方向序列中自动提取特征,还使用一些手工构建的特征(例如,数据包总数、接收和发出的数据包总数、结束和发出的数据包比例、总传输时间、数据包的平均传输时间).同时证明在较少样本量的情况下,Bh-VarCNN 仍然能保持较高的准确率.例如,在 100×100 的 closed-world 中,Bh-VarCNN 可以达到 97.8%的识别准确率,而 Si-DF[44] 的准确率为 93.6%.当样本充足时,在 900×2500 的 closed-world 中,Bh-VarCNN 达到了 98.8%的识别准确率,而 Si-DF 为 96.5%.在 100×100+90 000×1 的 open-world 中,Bh-VarCNN 达到 89.2%的 Multi-TPR 和 1.1%的 FPR.实验还指出,受监视页面和普通页面比例悬殊会导致大量误报,这与 Wang[46] 对基率谬误的讨论一致.

Tik-Tok[50] 还手动构建了 8 个基于 Burst 的时间特征,通过增强网页指纹识别方法对时间信息的利用能力,提升了现有网页指纹识别算法的可解释性以及抵抗干扰的能力.Burst 是指流量中的小段同方向序列,如图 4 所示,每一段同方向的数据包被视为是一个 Burst.受此启发,Ma 等人[51] 提出了 Ma-Burst-DF,通过设计多尺度的 Burst 特征提取网络改进了 Si-DF[44] 的网络结构,并且创造性地提出了要将神经网络的输出作为指纹向量,接着使用随机森林算法对指纹向量进行分析最终输出分类结果,其模型性能和抵抗干扰的能力较 Si-DF[44] 有所提升.这种基于 Burst 的特征提取结构在 Dy-VNG++[40] 和 Wa-kNN[26] 中也有应用.

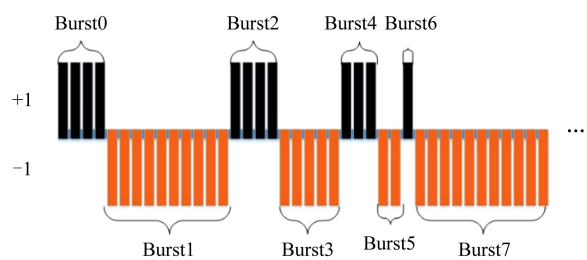

Fig. 4 Structure of Burst

图 4 Burst 结构

3.4 算法和特征的选择

早期的网页指纹识别利用数据包的大小、时间等信息手动构建特征,并计算流量之间的相关度做

匹配依据,例如 Bi-XCor[17]和 Su-Sim[24].接着,机器学习的经典算法被引入,在手动构造流量特征的情况下提升网页指纹识别的效果.首先被引入的是朴素贝叶斯,例如 Li-NB[18],He-MNB[21],Dy-VNG++[40].由于在解决小样本的非线性问题中的独特优势,SVM 算法也被广泛用于网页指纹识别.例如 Pa-SVM[42],Ca-OSAD[22],Wa-cOSAD[29],Pa-CUMUL[28]通过自定义内核,有效提升了流量分类的能力.基于 kNN 算法的 Wa-kNN[26]识别也达到了不错的效果.机器学习算法还可以用来提取特征,例如 Ha-kFP[23]识别使用随机森林作为特征提取的工具,把最后一层叶子节点的输出作为流量的指纹编码,然后计算指纹编码之间的距离进行分类.进入深度学习时代后,网页指纹识别也逐渐地脱离了对手动构建特征的依赖,转而使用特征提取网络.例如,Ri-DF[34],Si-DF[44],Si-Triplet[45]均是由特征提取网络从[1,−1]的方向序列中自动生成的特征,并取得很好的识别效果;而 Bh-VarCNN[43]则表明结合手动构建特征可以进一步提高识别性能.

根据表 2 的统计,数据包的时间和方向序列是 2 个最重要特征;数据包的数量、发出和收到数据包的比例、总传送时间、特定文件的大小和特殊数据包的大小等特征也常被网页指纹识别方法所选用.并且 Li 等人[52]分析了网页指纹的信息泄露(information leakage)情况,计算了网页指纹识别方法中常用特征所泄露的信息量,为手动构建特征的可解释性提供了理论支持.在基于深度学习的识别方法中,总字节数、流量反转次数等手动构建的特征往往能够在特征自动提取的基础上增强识别效果.但匿名网络中的识别一般不选取数据包大小,因为 Tor 的强制填充规则使其失去了特征表征能力.相比于 TCP/IP 层的数据包信息,从 Tor cells 层解析数据包的信息特征,能使分类器达到更好的效果[28].大量针对 Tor 流量的识别方法如 Wa-cOSAD[29],Wa-kNN[26],Ri-DF[34],Si-DF[44],Si-Triplet[45],Bh-VarCNN[43],都选择 Tor cells 作为特征提取的对象.

Table 2 The TypicalFeatures Used in WF Identification

表 2 网页指纹识别所使用的典型特征

识别方法	核心算法	数据单元	用于生成网页指纹的典型特征
Bi-XCor[17]	互相关系数	TCP/IP 数据包	数据包的大小、时间戳.
Li-NB[18]	朴素贝叶斯	TCP/IP 数据包	数据包的大小、方向.
He-MNB[21]	多项式朴素贝叶斯	TCP/IP 数据包	数据包的大小、方向.
Dy-VNG++[40]	朴素贝叶斯	TCP/IP 数据包	网页加载时间,上传下载的总数据量,访问期间流量 Burst 的数据量.
Su-Sim[24]	Jaccard 系数	网络对象	网络对象的大小、数量.
Pa-SVM[42]	支持向量机	TCP/IP 数据包	数据包的大小、顺序、数量,访问网页的总数据量,流量传输方向变化次数,HTML 的大小.
Ca-OSAD[22]	支持向量机	TCP/IP 数据包	数据包的大小、方向、顺序.
Wa-cOSAD[29]	支持向量机	Tor cells	Cell 的方向.
Pa-CUMUL[28]	支持向量机	Tor cells	Cell 的大小、方向、顺序.
Ha-kFP[23]	随机森林	TCP/IP 数据包	数据包的数量,发出和接收的数据包比例,接收数据包的顺序.
Wa-kNN[26]	最邻近算法	Tor cells	数据包的大小、数量、到达顺序,流量 Burst 的数量.
Ri-DF[34]	深度学习	Tor cells	Cell 的方向(自动提取).
Si-DF[44]	深度学习	Tor cells	Cell 的方向(自动提取).
Si-Triplet[45]	深度学习	Tor cells	Cell 的方向(自动提取).
Bh-VarCNN[43]	深度学习	Tor cells	Cell 的方向、时间戳(自动提取),接收和发送 Cell 的数量.

3.5 识别效果比较

表 3 对比了典型网页指纹识别方法的识别效果(为便于横向比较,仅列出实施一次识别的准确率).如果一项算法研究讨论了针对多种加密方式(如 SSH,VPN,Tor)的网页指纹识别,主要比较针对 Tor 流量的识别准确率.在 open-world 下,默认使用 Two-* 指标;否则指定 Multi-* 指标(相关定义见 2.3 节).本节还描述了每种方法评估所依托的数据集 $N\times X+M\times 1$,并按照受监视页面数量 N、普通页面数量 M 和训练样本采集数目 X 来讨论网页指纹识别的效果.

在 $N\times X$ 的 closed-world 中,考虑的受监视页面

越多则分类难度越大,对每个页面采集的训练样本越多则完成识别的开销越大.非深度学习算法的识别方法,对训练样本要求不高,X 通常设为 20 和 40.而基于深度学习的识别方法虽然能获得很高的识别准确率,但所需训练样本数目巨大.例如在 95×800 的 closed-world 中,Si-DF 可以获得 98.3%的准确率;在 900×2 500 的 closed-world 中,Bh-VarCNN 的准确率高达 98.8%,但它们的识别准备都非常耗时.

在 $N\times X+M\times 1$ 的 open-world 中,对网页指纹识别的评估往往受到普通页面与受监视页面比例的影响.普通页面越多,准确识别受监视页面的难度也就越大.表 3 中,Ri-DF 的比例最高 $M/N=2\,000/1$,可获得 80.11%的 TPR.比例次高的是 Bh-VarCNN($M/N=900/1$),在严格的多分类指标下,准确率达到了 89.2%的 Multi-TPR.若结合 closed-world 中的表现,Bh-VarCNN 方法的识别效果最佳.

基于深度学习的网页指纹识别方法需要依赖大量训练数据的支撑.在现实场景中,如果采集一个受监视页面流量需要花费 20 s(考虑 Tor 缓慢的响应速度,实际上还会更多),一台设备 24 h 仅采集 4 320 个样本.若受监视页面增加,收集和训练数据的识别准备过程将更长.而网页指纹识别的效果往往会随着搜集数据和发起识别的时间间隔加大而衰减.因此少样本学习(few shot learning, FSL)的表现和实践性更值得关注.例如,Si-Triplet 在 $M/N=90/1$ 的 open-world 中取得 89.3%的 TPR,仅使用少量训练数据获得较好识别效果.

3.6 本节小结

单标签页的网页指纹识别能达到很高的准确率,即使在 open-world 中的用户访问上万个网页,并且采用 SSH 及 SSL 等流量加密方法,攻击者也能通过训练网页指纹的识别算法准确地判断是否访问了某个受监视的页面.尽管用户使用 Tor 服务消除了数据包大小等信息以抵抗流量分析,但是随着相关研究的不断推进,针对 Tor 的网页指纹识别方法的效果也越来越好,多数防御手段被逐一打破.因此,基于网页指纹识别的攻击行为对用户隐私产生巨大威胁.早期的研究虽被诟病于脱离实际,但是随着识别算法从统计方法、传统机器学习,发展到最新的深度学习方法,针对 Tor 网页指纹识别在接近真实识别场景的实验环境中取得越来越好的效果.

Table 3 Performance of Single-tag WF Identification

表 3 单标签网页指纹识别的比较

识别方法	加密协议	closed-world		open-world	
		规模	准确率/%	规模	准确率/%
Bi-XCor[17]	SSH	100×1	23		
Li-NB[18]	SSH	1 000×4	73		
He-MNB[21]	Tor, Stunnel, VPN, SSH, AN.ON	775×4	2.96		
Dy-VNG++[40]	SSH	775×4	93.9		
Ca-OSAD[22]	Tor, SSH	100×40	87.3		
Su-Sim[24]	SSL/TLS			2191×1+98 496×1	75
Pa-SVM[42]	Tor, AN.ON	775×20	54.61	5×35+4 000×1	73
Wa-cOSAD[29]	Tor	100×40	91.0	4×40+860×1	96.9
Pa-CUMUL[28]	Tor	100×40	92.22	100×90+9 000×1	96.92
Ha-kFP[23]	Tor, SSL/TLS	100×90	91	30×80+16 000×1	81
Wa-kNN[26]	Tor	100×90	91	100×90+5 000×1	85
Ri-DF[34]	Tor	100×2 500	96.26	200×2 000+400 000×1	80.11
Si-DF[44]	Tor	95×800	98.3	95×900+20 000×1	96
Si-Triplet[45]	Tor	100×1	79.4	100×5+9 000×1	89.3
Bh-VarCNN[43]	Tor	900×2 500	98.8	100×100+90 000×1	89.2(Multi)

注:1) $N\times X$ 是 closed-world 数据集的规模,$N\times X+M\times 1$ 是 open-world 数据集的规模.其中,N 是受监视网页的数量,每个受监视网页有 X 个样本;M 是普通网页的数量,每个普通网页只有 1 个样本.2) 标记"Multi"的值为 Multi-* 评估指标体系中的结果.

4 面向多标签的网页指纹识别

多标签的网页指纹识别,认为用户在进行网页浏览时,会打开多个标签页同步访问多个网页.因为多个网页访问流量的相互叠加,面向单标签的网页指纹识别方法经常失效.解决这个问题的关键是准确判断一段网络流量是否包含多个页面的信息,如何准确分割各个网页流量,并设计分类算法对分割后的流量进行准确分类.本节将介绍多标签网页指纹识别的难点挑战与相关工作,然后比较这些研究的识别效果.

4.1 多标签网页指纹识别的特点

在多标签场景下,攻击者可以通过多标签流量判定以及流量分割 2 个步骤将问题转换为单标签网页指纹识别.如图 5 所示,如果一段流量记录被判定仅含有一个页面,则直接使用单标签识别方法进行特征提取和网页分类识别;如果判定含有多个页面,则进行流量分割后再分别转换为单标签识别问题.目前的多标签识别的研究[35-38]常采用这种思路.

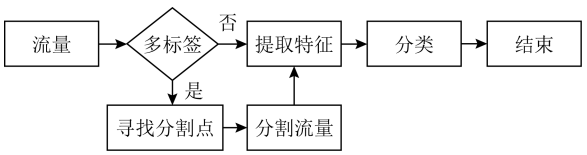

Fig. 5 The workflow of multi-tag WF attack

图 5 多标签页场景的识别流程

但是多标签场景并不是单标签网页指纹识别的一种简单扩展.首先单标签网页指纹识别方法无法直接运用于多标签场景的流量分类任务.Juarez 等人[30]测试了 5 种单标签网页指纹识别方法在多标签场景下的表现:He-MNB[21]、Wa-cOSAD[29]、Dy-VNG++[40]、Pa-SVM[42]、决策树算法.实验表明这些方法在多标签场景的网页识别准确率不足 20%,即识别无效.多标签下的网页指纹识别方法必须强调鲁棒性,有效抵抗多标签浏览行为带来的诸多干扰,例如多个页面访问的流量互相重叠、多标签判定算法的误差以及流量分割算法的误差.只有把多标签判定、流量分割及流量分类 3 个步骤进行整体化设计才能达到好的网页指纹识别效果.

4.2 多标签网页指纹识别的相关方法

基于数据包时间差的最邻近算法 Wa-TkNN[35],可以准确判断一段 Tor 流量是否属于多标签浏览,准确率高达 97%.它使用的特征包括 cell 间的最小和最大时间差,以及时间差的均值和方差等.如果一段流量被判定为多标签浏览,则使用 kNN 分类器来寻找不同标签页之间的流量分割点,所使用的特征包括分割点周围 5 个 cell 之间的时间差,以及分割点周围 50 个 cell 的时间差中的最大值、最小值和方差等.实验证明在 3 种多标签浏览情况下,网页流量的分割准确率分别为 88%,63%,32%.这 3 种典型情况是:在第 1 个页面完全加载后有一段间隔再访问第 2 个页面(class1);在第 1 个页面完全加载后立刻访问第 2 个页面(class2);在第 1 个页面未完全加载完时就访问第 2 个页面(class3).可惜的是,Wa-TkNN[35]工作仅提出了判定和分割多标签流量的方法,并没有进一步设计基于此的网页指纹识别.

Xu 等人选取与 Wa-TkNN[35]相同的流量特征,融合 BalanceCascade[53]和 XGBoost[54]分类器构建了一个二分类的分类器 Xu-Boost[37].它通过独立地计算用户每一个发出的数据包是真实分割点的概率来寻找分割点,并且构建了随机森林分类器对分割后的流量进行分类.随机森林分类器使用数据流量总大小、cells 数量等手动构建的特征,对第 1 个页面的分类达到了 86.56%的 Multi-TPR,及 0.52%的 FPR.可惜的是,Xu-Boost 抛弃了分割点后的所有流量,仅对第 1 个页面进行分类.

Gu-NB[38]假设第 2 个页面在固定时间差后打开(例如 2s),因此省略网页间的流量分割过程,直接讨论如何分类 2 段相互干扰的流量.通过提取一段流量的 TCP 连接数量、流出/流入总带宽和数据包间时间差等特征,利用随机森林算法对 2 个页面进行了分类.第 1 个页面的分类识别达到了 75.9%的准确率和 22%的 FPR;第 2 个页面则是 40.5%的准确率及 17.4%的 FPR.然而 Gu-NB[38]假设过于理想化,不能解决多标签网页指纹识别的实际问题;并且实验仅针对 SSH 流量,未证明其在 Tor 匿名网络中的作用.

Cui 等人[36]指出多标签场景下的网页指纹识别主要面对 2 种干扰:流量的缺失(missing)和重叠(overlapping).当用户进行多标签浏览时,攻击者将整段流量记录进行分割后,第 1 个页面容易缺失尾部的信息,而第 2 个页面的头部会受到第 1 个页面的尾部信息干扰.基于 Wa-TkNN[35]判断多标签浏览的流量的能力,他们提出了相当完整的多标签网页指纹识别方法 Cu-HMM[36].它首先使用隐马尔可夫模型对流量进行分割,再使用 Sectioning 算法对

2 段流量进行分类.Sectioning 算法将每个页面的流量拆分为一些固定长度的小段,分别对每一小段进行分类,统计小段的分类结果即得到了整段流量的分类结果,其中使用的特征来源于 Pa-CUMUL[28].closed-world 实验表明,Cu-HMM 识别对第 1 个页面和第 2 个页面分别达到了 70.2% 和 69.2% 的 Top3 准确率(分类器输出的概率最大 3 个类别中若包括正确类别,则认为分类正确).可惜它没有测试 open-world 的场景下网页指纹识别的效果,所以仍然存在一定改进的空间.

4.3 识别效果比较

表 4 与表 5 对比了多标签网页指纹识别方法的分割与识别效果,时间间隔指的是开始访问 2 个网页之间的时间差,E_{split} 表示流量分割误差,Acc 和 TPR 表示流量分割效果,规模记录了数据集组成方式.由于不同研究中采用的性能评价指标不同,并没有提供所使用的数据集以及实现代码.难以使用统一的单位来对比其性能和表现,因此表 4 与表 5 中仅采用文献中提供的评价指标与数值以供参考.另外由于由 Gu-NB[38]假设固定的时间差,并跳过了流量分割过程,所以在表 4 中 Gu-NB[38]分割指标的 3 项为空.而 Wa-TkNN[35]仅考虑了多标签页场景下的流量分割问题,因此在表 5 中,Wa-TkNN[35]分类指标皆为空.

显然,流量分割算法是多标签网页指纹识别成功的关键,精准的流量分割才能保证后续的分类算法的准确性.为了达到最好的识别效果,研究者需要提前分析流量分割算法的性能.然而不同分割算法猜测网页流量分割点的方法也不同,主要有 2 种:第 1 种,尽量让猜测的流量分割点与真实的距离不超过规定的数据包数目 Y,例如 Xu-Boost[37]和 Wa-TkNN[35];第 2 种是把长为 L_s 的流量纪录为许多长度为 L_{sub} 的小段,如果猜测的流量分割点与真实的位于同一 L_{sub},则认为分割正确,例如 Cu-HMM[36].

为了方便描述和对比,需要定义一个指标用于衡量分割误差 E_{split},即使一个分割点被判定为正确,仍然可能存在的误差.2 种情况的误差计算公式分别为式(1)和式(2).

$$E_{split}=(Y/L_s)\times 100\%, \tag{1}$$

$$E_{split}=(L_{sub}/L_s)\times 100\%. \tag{2}$$

Xu-Boost[37]和 Cu-HMM[36]均达到 80% 以上的流量分割准确率,是目前效果最好的分割算法.而 Wa-TkNN[35]由于过度依赖流量中的时间特征,虽然对 2 个网页流量间有间隔的情况可以达到较高的分割准确率,但是对 2 个网页流量有部分重叠的情况不能做出很好的分割,因此最终的分割性能不佳.

多标签网页指纹识别的第 2 个关键技术是流量分类算法.由于分割后的流量往往会存在缺失和重叠 2 种干扰[36],多标签网页指纹识别场景下的分类算法往往要具备更强的抗干扰能力.

Table 4 Performance of Multi-tab WF Split

表 4 多标签网页指纹分割的比较 %

识别方法	时间间隔	加密协议	E_{split}	Acc	TPR
Xu-Boost[37]	随机	Tor	$25/L_s$	82	
Cu-HMM[36]	随机	Tor	6.67	80	
Wa-TkNN[35]	随机	Tor	$25/L_s$		55
Gu-NB[38]	固定	SSH			

Table 5 Performance of Multi-tab WF Identification

表 5 多标签网页指纹识别的比较

识别方法	规模	第 1 个页面的分类结果			第 2 个页面的分类结果		
		TPR/%	Acc/%	FPR/%	TPR/%	Acc/%	FPR/%
Xu-Boost[37]	50×50+2500	86.56(Multi)		0.52			
Cu-HMM[36]	100×40		70.2(Top3)			69.2(Top3)	
Wa-TkNN[35]							
Gu-NB[38]	50×50+50		75.9	22		40.5	17.4

注:1) 标记“Multi”的值为 Multi-* 评估指标体系中的结果.2) 若分类器输出的概率最大的 3 个分类结果包含真实类别,则认为分类正确,并据此计算性能指标,标记为(Top3).

在 $N\times X+M\times 1$ 的 open-world 中,Gu-NB[38]选取 SSH 而不是 Tor 流量,并省略流量分割的过程,因此 2 个页面 75.9% 和 40.5% 的分类准确率也难以证明其可用于真实的多标签情景.在受监视页面数量 $N=50$ 时,Xu-Boost[37]的普通页面数量 M 是 Gu-NB[38]的 5 倍.面对更困难的分类任务,Xu-Boost[37]把第 1 个页面的分类准确率提高到 86.56% 的 Multi-TPR.但由于忽略了第 2 个页面的流量,并采用不同

的数据集分别验证流量分割和分类性能,Xu-Boost的实验结果无法全面反映其真实性能.

在 $N \times X$ 的 closed-world 中,仅有 Cu-HMM[36],得益于准确率 80%的分割算法,并提出 Sectioning 算法有效抵抗了第 2 个页面流量头部所受到的干扰.Cu-HMM[36]对第 2 个页面的分类准确率达到了 69.2%的 Top3 准确率,也是对第 2 个页面的分类准确率最高的识别算法.

4.4 本节小结

多标签场景是网页指纹识别相关研究的未来发展方向,因为它更加贴近实际应用场景,对用户隐私的威胁也更大.攻击者可以用高准确率的分割算法削弱多标签浏览的影响,完成多标签场景的判定,并且进行流量分割,最后使用流量分类的方法识别网页,完成网页指纹识别.综合来看,Cu-HMM[36]的设计和实验最为合理,使用 Wa-TkNN 准确判定多标签网页流量,然后流量分割和流量分类的完整过程.其中除了 Gu-NB[38]是针对 SSH 加密流量进行的分析外,Xu-Boost[37],Cu-HMM[36],Wa-TkNN[35]均针对的是 Tor 加密的流量.在多标签场景中,即使用户在短时间内打开多个页面,Tor 也可能在此期间重新建立 Circuit 进行通信,而 Circuit 的切换会导致数据包数量及时间戳等信息发生变化,这些潜在的干扰都极大地增加了网页指纹识别的难度.对流量分割和分类算法的性能及抵抗干扰的能力的要求也就更加严格.

5 网页指纹识别的局限性

网页指纹识别的场景和假设越来越接近实际情况,从 closed-world 到 open-world 的进步显而易见.本节从实践角度出发,讨论威胁场景、条件假设、Tor 版本、背景流量、采集数据和发起识别的时间差等因素对网页指纹识别效果的影响,如表 6 所示.在设计识别方法时,必须重视这些因素的影响.

绝大部分识别方法假设已知 Tor 浏览器版本及操作系统等用户信息,因此攻击者可以模仿用户的网络环境来进行数据的收集并做好识别准备工作.然而当用户的真实网络环境与攻击者模拟环境不完全相同时,网页指纹识别效果会大打折扣[28].

Table 6 The Influencing Factors Considered in Typical WF Identification

表 6 典型 WF 识别方法考虑的影响因素列表

识别方法	closed-world	open-world	Tor 版本	背景流量	防御手段	时间差
Su-Sim[24]		√			√	
Bi-XCor[17]	√					
Li-NB[18]	√				√	√
He-MNB[21]	√				√	√
Dy-VNG++[40]	√				√	
Pa-SVM[42]	√	√		√	√	
Ca-OSAD[22]	√					
Wa-cOSAD[29]	√	√			√	
Pa-CUMUL[28]	√	√		√	√	
Ha-kFP[23]	√	√			√	
Wa-kNN[26]	√	√			√	
Ri-DF[34]	√	√			√	√
Si-DF[44]	√	√			√	
Si-Triplet[45]	√	√	√		√	√
Bh-VarCNN[43]	√	√			√	
Wa-Boost[37]	√	√		√	√	
Cu-HMM[36]	√	√		√		
Gu-NB[38]	√	√		√		
Wa-TkNN[35]	√	√		√		

目前仅有 Si-Triplet[45]考虑了 Tor 版本的影响.

网页指纹识别应当充分考虑背景流量对识别效果的影响以提高其可实践性.大部分识别方法假设攻击者在捕获用户的网络流量后,能够完整且无重叠地切分出网页浏览的流量.例如,Wa-kNN[26]把 Tor 浏览器的自带流量(circuit 建立和控制流量的 SEND ME 数据包等)都视为噪音进行过滤,以提高识别准确率.但这在现实网络环境中无法真正实现.正常的联网设备往往会持续不断地产生网络流量,即使攻击者仅采集用户的 TLS 流量也会被其他应用或者程序所产生的 TLS 流量干扰.只有 Pa-SVM[42],Pa-CUMUL[28], Wa-Boost[37], Cu-HMM[36], Gu-NB[38], Wa-TkNN[35]方法通过手动添加噪声的方式测试了自身算法抵抗背景流量噪声干扰的能力.文献[55]提出可以通过在训练分类器期间加入目标页面内超链接的统计来提升训练器抵抗噪声的能力.

由于大部分网页是非静态的,网页指纹识别的效果往往会随着搜集数据和发起识别的时间间隔增加而衰减.网页指纹识别的数据搜集和分类器训练需要花费大量的准备时间[45].当用户进行网页访问时,当前网站很可能已经发生变化,例如视频网站首页往往会根据热度变化来排列节目的链接,页面之间插入的广告也在更新.因此,从识别准备到真正发起识别的时间越短越好,网页指纹识别的准确率会随着时间间隔增大而下降[18,24,34,44].网页指纹识别的时效性也是一个现实的挑战.

6 研究总结与展望

本综述从网页指纹识别的概念、原理和威胁场景出发,分类讨论面向单标签和多标签的典型识别方法.特别研究他们针对 Tor 等匿名网络的识别效果,并从不同角度讨论其性能优点、局限性及可实践性.5 个问题值得进一步研究和讨论.

1) 设计更具实践性的网页指纹识别.现有的网页指纹识别方法研究普遍采用了较多的实验假设,例如 closed-world、单标签浏览、无背景噪声及训练数据采集源与用户设备操作环境完全一致等,每一个假设因素都会对网页指纹识别效果产生较大影响.这些假设使现有识别方法缺乏实践性,虽然有研究者提出了一些针对多标签场景的网页指纹识别方法[35,38],也有研究者针对网页指纹的可实践性进行了讨论[30,35],但是网页指纹识别距离真正实用仍有不小差距.Tor project 早于 2013 年已经将网页指纹识别视为某种潜在威胁[41],然而至今未推送任何主动防御措施.可见,设计更具实践性的网页指纹识别,比挫败某种主动防御更紧迫.

2) 针对 Tor 协议的特性设计网页指纹识别方法.网页指纹相关研究的热点虽然已面向 Tor 加密的网页访问,但研究者往往是通过提升分类器的性能及鲁棒性的方法来攻克 Tor 对流量分析技术的干扰.这类方法一般具有通用性,可以同时应对多种加密和防御手段.但对于使用最广泛的匿名访问网络 Tor,研究者应该针对 Tor 协议本身的特性(例如未施加干扰的数据包间时间差、路由选择策略等)来开发更高效率的网页指纹识别方法,预期能达到更好的效果[50].

3) 使用数据增强改进网页指纹识别.数据增强方法被广泛应用于图像处理领域,通过对图像进行遮挡、旋转、融合和尺度变换来扩展数据集,模糊分类器的决策边界.这些经过简单变换的样本与真实样本很接近,有利于提升分类模型的表现.网页指纹识别的研究,也可以通过数据增强的方法增加训练数据,提升模型的准确率.这种改进方案,不但降低了网页指纹识别对于大量真实数据的依赖,也大大减少了搜集数据训练分类器的准备时间.

4) 研究针对网页指纹识别的防御.随着网页指纹识别方法的发展,涌现了对应的防御措施[56-65].最基本的是通过填充垃圾包和主动延迟某些数据包(或组合),对网页访问流量进行保护,提高网页指纹识别的假阳率或降低其准确率.但这 2 种措施都会影响网页访问体验,过量添加垃圾包会对用户带宽带来额外负担,过长的数据包延迟会加大用户访问的时延开销.由于网页指纹识别的现有防御措施开销都比较大,难以实际部署到 Tor 等匿名网络中.平衡防御效果与开销成为一大难题,而在图像领域兴起的对抗网络技术为此打开了新思路,已有研究使用对抗样本解决网页指纹识别的威胁并且取得了不错的效果[63,65].防御者通过制造对抗样本产生足够针对网页指纹识别算法的网络干扰,同时保证这些干扰不会影响网络的正常通信.

5)增强网页指纹识别的网络监管能力.匿名网络能够提供用户匿名访问互联网的能力,有效地保护用户隐私.但若被不法分子滥用来进行非法活动,则增加了监管部门对网络犯罪的审查难度.网页指纹识别的研究能够被应用于合法的网络监管之中,防范不法分子对匿名网络的滥用.研究网页指纹识别方法的最终目的并不是破坏隐私保护体系,而是

通过寻找可能的网页指纹识别方法来测试现有隐私保护体系的性能和潜在威胁,并且在法律准许的范围内进行有效的网络监管.通过不断对网页指纹识别方法的推进与研究,才能提出更好的隐私保护方案,增强对合法用户的隐私保护能力,增强对非法用户的网络监管能力.

参 考 文 献

[1] Rescorla E. RFC 8446 The Transport Layer Security (TLS) Protocol Version 1.3 [S/OL]. 2018 (2020-03-07) [2020-04-10]. https://tools.ietf.org/html/rfc8446

[2] Rescorla E. RFC 2818 HTTP Over TLS [S/OL]. 2000 (2013-03-02) [2020-04-10]. https://tools. ietf. org/html/rfc2818

[3] Benjamin D. RFC 8740 Using TLS 1.3 with HTTP/2 [S/OL]. 2020 (2020-02-23) [2020-04-10]. https://tools.ietf.org/html/rfc8740

[4] Gu Xiaodan, Yang Ming, Luo Junzhou, et al. Website fingerprinting attack based on hyperlink relations [J]. Chinese Journal of Computers, 2015, 38(4): 833-845 (in Chinese)
(顾晓丹, 杨明, 罗军舟, 等. 针对 SSH 匿名流量的网站指纹攻击方法[J]. 计算机学报, 2015, 38(4): 833-845)

[5] Hu Zi, Zhu Liang, Heidemann J, et al. RFC 7858 Specification for DNS over Transport Layer Security (TLS) [S/OL]. 2016 (2018-12-20) [2020-04-10]. https://tools.ietf.org/html/rfc7858

[6] Dickinson S, Gillmor D, Reddy T, et al. RFC 8310 Usage Profiles for DNS over TLS and DNS over DTLS [S/OL]. 2018 (2018-03-21) [2020-04-10]. https://tools.ietf.org/html/rfc8310

[7] Hoffman P, McManus P. RFC 8484 DNS Queries over HTTPS (DOH) [S/OL]. 2018 (2019-01-15) [2020-04-10]. https://tools.ietf.org/html/rfc8484

[8] Rescorla E, Sullivan N. TLS Encrypted Client Hello [S/OL]. 2020 (2020-06-01) [2020-07-05]. https://tools.ietf.org/html/draft-ietf-tls-esni-07

[9] Dingledine R, Mathewson N, Syverson P. Tor: The second-generation onion router [C] //Proc of the 13th USENIX Security Symp. Berkley, CA: USENIX Association, 2004: 303-320

[10] Luo Junzhou, Yang Ming, Ling Zhen, et al. Anonymous communication and darknet: A survey [J]. Journal of Computer Research and Development, 2019, 56(1): 103-130 (in Chinese)
(罗军舟, 杨明, 凌振, 等. 匿名通信与暗网研究综述[J]. 计算机研究与发展, 2019, 56(1): 103-130)

[11] Grün K G, Wendolsky R, Daniel D, et al. Project AN. ON [EB/OL]. 2005 [2020-04-15]. https://anon. inf. tu-dresden. de/

[12] Clarke I, Miller S G, Hong T W, et al. Protecting free expression online with Freenet [J]. IEEE Internet Computing, 2002, 6(1): 40-49

[13] The I2P Development Team. The invisible Internet project [EB/OL]. (2018-09-01) [2020-04-15]. https://geti2p.net/zh/

[14] Krever T. International criminal law: An ideology critique [J]. Leiden Journal of International Law, 2013, 26(3): 701-723

[15] Minarik T, Osula A M. Tor does not stink: Use and abuse of the Tor anonymity network from the perspective of law [J]. Computer Law & Security Report, 2016, 32(1): 111-127

[16] He Gaofeng, Yang Ming, Luo Junzhou, et al. Online identification of Tor anonymous communication traffic [J]. Journal of Software, 2013, 24(3): 540-556 (in Chinese)
(何高峰, 杨明, 罗军舟, 等. Tor 匿名通信流量在线识别方法[J].软件学报,2013,24(3): 540-556)

[17] Bissias G D, Liberatore M, Jensen D, et al. Privacy vulnerabilities in encrypted HTTP streams [C] //Proc of Int Workshop on Privacy Enhancing Technologies. Berlin: Springer, 2005: 1-11

[18] Liberatore M, Levine B N. Inferring the source of encrypted HTTP connections [C] //Proc of the 13th ACM Conf on Computer and Communications Security. New York: ACM, 2006: 255-263

[19] Hintz A. Fingerprinting websites using traffic analysis [C] //Proc of Int Workshop on Privacy Enhancing Technologies. Berlin: Springer, 2002: 171-178

[20] Zhang Lei, Cui Yong, Liu Jing, et al. Application of machine learning in cyberspace security research [J]. Chinese Journal of Computers, 2018, 41(9): 1943-1975 (in Chinese)
(张蕾, 崔勇, 刘静, 等. 机器学习在网络空间安全研究中的应用[J]. 计算机学报, 2018, 41(9): 1943-1975)

[21] Herrmann D, Wendolsky R, Federrath H. Website fingerprinting: Attacking popular privacy enhancing technologies with the multinomial naïve-Bayes classifier [C] //Proc of ACM Workshop on Cloud Computing Security. New York: ACM, 2009: 31-42

[22] Cai Xiang, Zhang Xincheng, Joshi B, et al. Touching from a distance: Website fingerprinting attacks and defenses [C] //Proc of the 2012 ACM Conf on Computer and Communications Security. New York: ACM, 2012: 605-616

[23] Hayes J, Danezis G. k-fingerprinting: A robust scalable website fingerprinting technique [C] //Proc of the 25th USENIX Security Symp. Berkley, CA: USENIX Association, 2016: 1187-1203

[24] Sun Qixiang, Simon D R, Wang Yimin, et al. Statistical identification of encrypted Web browsing traffic [C] //Proc of IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2002: 19-30

[25] Cheng H, Avnur R. Traffic analysis of SSL encrypted Web browsing [EB/OL]. 1998 [2020-04-16]. http://citeseerx.ist.psu.edu/viewdoc/summary? doi=10.1.1.3.1201

[26] Wang Tao, Cai Xiang, Nithyanand R, et al. Effective attacks and provable defenses for website fingerprinting [C]//Proc of the 23rd USENIX Security Symp. Berkley, CA: USENIX Association, 2014: 143-157

[27] Dingledine R, Murdoch S J. Performance improvements on Tor or, why Tor is slow and what we're going to do about it [EB/OL]. (2009-03-11) [2020-04-16]. http://www.torproject.org/press/presskit/2009-03-11-performance.pdf

[28] Panchenko A, Lanze F, Pennekamp J, et al. Website fingerprinting at Internet scale [C] //Proc of the 23rd Network and Distributed System Security Symp. Rosten: The Internet Society, 2016: 1-15

[29] Wang Tao, Goldberg I. Improved website fingerprinting on tor [C] //Proc of the 12th ACM Workshop on Privacy in the Electronic Society. New York: ACM, 2013: 201-212

[30] Juarez M, Afroz S, Acar G, et al. A critical evaluation of website fingerprinting attacks [C] //Proc of ACM Special Interest Group on Security, Audit and Control Conf on Computer and Communications Security. New York: ACM, 2014: 263-274

[31] Smith F D, Campos F H, Jeffay K, et al. What TCP/IP protocol headers can tell us about the Web [C] //Proc of ACM Special Interest Group (SIG) for the Computer Systems Performance Evaluation Community Int Conf on Measurement and Modeling of Computer Systems. New York: ACM, 2001: 245-256

[32] Mozilla Labs. Test Pilot: Tab open/close study: Results [EB/OL]. (2013-03-17) [2020-05-10]. https://testpilot.mozillalabs.com/testcases/tab-Open-close/results.html

[33] Christian W, Hauswirth M. Dobbs: Towards a comprehensive dataset to study the browsing behavior of online users [C] //Proc of IEEE/WIC/ACM Int Joint Conf on Web Intelligence and Intelligent Agent Technologies. Piscataway, NJ: IEEE, 2013: 51-56

[34] Rimmer V, Preuveneers D, Juarez M, et al. Automated website fingerprinting through deep learning [EB/OL]. (2017-12-05) [2020-05-10]. https://arxiv. org/pdf/1708.06376

[35] Wang Tao, Goldberg I. On realistically attacking Tor with website fingerprinting [J]. Proceeding on Privacy Enhancing Technologies, 2016, 2016 (4): 21-36

[36] Cui Weiqi, Chen Tao, Fields C, et al. Revisiting assumptions for website fingerprinting attacks [C] //Proc of ACM Asia Conf on Computer and Communications Security. New York: ACM, 2019: 328-339

[37] Xu Yixiao, Wang Tao, Li Qi, et al. A multi-tab website fingerprinting attack [C] //Proc of the 34th Annual Computer Security Applications Conf. New York: ACM, 2018: 327-341

[38] Gu Xiaodan, Yang Ming, Luo Junzhou. A novel website fingerprinting attack against multi-tab browsing behavior [C] //Proc of the 19th IEEE Int Conf on Computer Supported Cooperative Work in Design (CSCWD). Piscataway, NJ: IEEE, 2015: 234-239

[39] Cao Zigang, Xiong Gang, Zhao Yong, et al. A survey on encrypted traffic classification [C] //Proc of Int Conf on Applications and Techniques in Information Security. Berlin: Springer, 2014: 73-81

[40] Dyer K P, Coull S E, Ristenpart T, et al. Peek-a-boo, I still see you: Why efficient traffic analysis countermeasures fail [C] //Proc of IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2012: 332-346

[41] Perry M. A critique of website traffic fingerprinting attacks [EB/OL]. (2013-11-07) [2020-05-10]. https://blog.torproject.org/critique-website-traffic-fingerprinting-attacks

[42] Panchenko A, Niessen L, Zinnen A, et al. Website fingerprinting in onion routing based anonymization networks [C] //Proc of the 10th Annual ACM Workshop on Privacy in the Electronic Society. New York: ACM, 2011: 103-114

[43] Bhat S, Lu D, Kwon A, et al. Var-CNN: A data-efficient website fingerprinting attack based on deep learning [J]. Proceeding on Privacy Enhancing Technologies, 2019, 2019 (4): 292-310

[44] Sirinam P, Imani M, Juarez M, et al. Deep fingerprinting: Undermining website fingerprinting defenses with deep learning [C] //Proc of ACM Special Interest Group on Security, Audit and Control Conf on Computer and Communications Security. New York: ACM, 2018: 1928-1943

[45] Sirinam P, Mathews N, Rahman M S, et al. Triplet fingerprinting: More practical and portable website fingerprinting with N-shot learning [C] //Proc of the 2019 ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2019: 1131-1148

[46] Wang Tao. Optimizing precision for open-world website fingerprinting [EB/OL]. (2018-02-15) [2020-05-10]. https://arxiv.org/pdf/1802.05409

[47] Van Rijsbergen C J. Information Retrieval [M]. 2nd ed. London: Butterworths, 1979

[48] Navarro G. A guided tour to approximate string matching [J]. ACM Computing Surveys, 2001, 33(1): 31-88

[49] Abe K, Goto S. Fingerprinting attack on Tor anonymity using deep learning [J]. Proceeding of the Asia-Pacific Advanced Network, 2016, 42: 15-20

[50] Rahman M S, Sirinam P, Matthew M, et al. Tik-Tok: The utility of packet timing on website fingerprinting attacks [EB/OL]. (2019-11-01) [2020-05-10]. https://arxiv.org/pdf/1902.06421.pdf

[51] Ma Chencheng, Du Xuehui, Cao Lifeng, et al. Burst-Analysis website fingerprinting attack based on deep neural network [J]. Journal of Computer Research and Development, 2020, 57(4): 746-766 (in Chinese)
(马陈城, 杜学绘, 曹利峰, 等. 基于深度神经网络 burst 特征分析的网站指纹攻击方法[J]. 计算机研究与发展, 2020, 57(4): 746-766)

[52] Li Shuai, GuoHuajun, Hopper N. Measuring information leakage in website fingerprinting attacks and defenses [C] //Proc of ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2018: 1977-1992

[53] Liu Xuying, Wu Jianxin, Zhou Zhihua. Exploratory undersampling for class-imbalance learning [J]. IEEE Transactions on Systems, Man, and Cybernetics, 2009, 39 (2): 539-550

[54] Chen Tianqi, Guestrin C. XGBoost: A scalable tree boosting system [C] //Proc of the 22nd ACM Special Interest Group on Knowledge Discovery in Data Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2016: 785-794

[55] Zhuo Zhongliu, Zhang Yang, Zhang Zhili, et al. Website fingerprinting attack on anonymity networks based on profile hidden Markov model [J]. IEEE Transactions on Information Forensics and Security, 2017, 13(5): 1081-1095

[56] Gong Jiajun, Wang Tao. Zero-delay lightweight defenses against website fingerprinting [C] //Proc of the 29th USENIX Security Symp. Berkley, CA: USENIX Association, 2020: 717-734

[57] Juarez M, Imani M, Perry M, et al. Toward an efficient website fingerprinting defense [C] //Prof of European Symp on Research in Computer Security. Berlin: Springer, 2016: 27-46

[58] Cai Xiang, Nithyanand R, Wang Tao, et al. A systematic approach to developing and evaluating website fingerprinting defenses [C]//Proc of ACM Special Interest Group on Security, Audit and Control Conf on Computer and Communications Security. New York: ACM, 2014: 227-238

[59] Cai Xiang, Nithyanand R, Johnson R. Cs-buflo: A congestion sensitive website fingerprinting defense [C] //Proc of the 13th Workshop on Privacy in the Electronic Society. New York: ACM, 2014: 121-130

[60] Lu D, Bhat S, Kwon A, et al.Dynaflow: An efficient website fingerprinting defense based on dynamically-adjusting flows [C] //Proc of the 2018 Workshop on Privacy in the Electronic Society. New York: ACM, 2018: 109-113

[61] Wright C V, Coull S E, Monrose F. Traffic morphing: An efficient defense against statistical traffic analysis [C/OL] // Proc of NDSS. San Diego, CA: NDSS, 2009 [2020-06-16]. https://www.ndss-symposium.org/ndss2009/

[62] Nithyanand R, Cai Xiang, Johnson R. Glove: A bespoke website fingerprinting defense [C] //Proc of the 13th Workshop on Privacy in the Electronic Society. New York: ACM, 2014: 131-134

[63] Wang Tao, Cai Xiang, Nithyanand R, et al. Effective attacks and provable defenses for website fingerprinting [C] //Proc of the 23rd USENIX Security Symp. Berkley, CA: USENIX Association, 2014: 143-157

[64] Imani M, Rahman M S, Wright M. Adversarial traces for website fingerprinting defense [C] //Proc of ACM Special Interest Group on Security, Audit and Control Conf. New York: ACM, 2018: 2225-2227

[65] Imani M, Rahman M S, Mathews N, et al. Mockingbird: Defending againstdeep-learning-based website fingerprinting attacks with adversarial traces [EB/OL]. (2019-02-21) [2020-06-15]. https://arxiv.org/pdf/1902.06626

Sun Xueliang, born in 1996. Received his BSc degree in information security from Harbin Engineering University in 2019. Currently MSc candidate at Xiamen University. His main research interests include network security and machine learning.

孙学良,1996 年生.2019 年获得哈尔滨工程大学信息安全学士学位,现厦门大学硕士研究生.主要研究方向为网络安全和机器学习.

Huang Anxin, born in 1997. Received his BSc degree in computer science from Anhui Institute of Information Technology in 2018 and his MSc degree from Xiamen University in 2021. Currently works at Beijing Qihoo Technology Co., Ltd. His main research interests include network security and computer network.

黄安欣,1997 年生.2018 年获得安徽信息工程学院计算机科学学士学位,2021 年获得厦门大学信息学院计算机技术硕士学位,现任职于北京奇虎科技有限公司.主要研究方向为网络安全和计算机网络.

Luo Xiapu, born in 1977. Received his PhD degree from The Hong Kong Polytechnic University and then spent two years at the Georgia Institute of Technology as a postdoctoral research fellow. Currently associate professor at Department of Computing, The Hong Kong Polytechnic University. His current research interests include network and system security, blockchain and smart contract, mobile and IoT security.

罗夏朴,1977 年生.在香港理工大学获得博士学位,并在佐治亚理工学院担任 2 年博士后研究员.现香港理工大学计算机系副教授.主要研究方向为网络和系统安全、区块链和智能合约、移动和物联网安全.

Xie Yi, born in 1979. Received her BSc and MSc degrees from Xi'an Jiaotong University, as well as her PhD degree from The Hong Kong Polytechnic University. Currently associate professor at Computer Science Department of Xiamen University, China. Member of CCF and ACM. Her current research interests include network security, wireless communication, network protocol analysis, and modeling & simulation. (csyxie@xmu.edu.cn)

谢 怡,1979 年生.在西安交通大学获得本科和硕士学位,香港理工大学获得博士学位.现厦门大学计算机科学与技术系副教授,CCF 和 ACM 会员.主要研究方向为网络安全、无线通信、网络协议分析、建模和仿真.