文章编号: 2095-6134(2016)04-0562-08

基于余弦测度下 K-means 的网络空间终端设备识别*

曹来成[1], 赵建军[1,2†], 崔 翔[2], 李 可[2,3]

(1 兰州理工大学计算机与通信学院,兰州 730050; 2 中国科学院信息工程研究所,北京 100093;
3 北京邮电大学计算机学院,北京 100876)

(2016 年 1 月 7 日收稿; 2016 年 3 月 17 日收修改稿)

Cao L C, Zhao J J, Cui X, et al. Cyberspace device identification based on K-means with cosine distance measure [J]. Journal of University of Chinese Academy of Sciences, 2016, 33(4): 562-569.

摘 要 针对传统 Web 指纹识别方法中识别对象局限于主流 Web 服务器软件的问题,提出一种基于余弦测度下 K-means 的网络空间终端设备识别模型. 首先,设计识别模型和确定验证方法. 其次,选取返回的 HTTP 数据包头部字段和状态码作为终端设备特征,对特征进行提取和向量化后转化为 32 维特征向量. 再次,选取余弦距离函数作为 K-means 聚类算法中的相似性度量函数. 最后,根据识别模型设计实验算法流程,对网络空间中的无标记样本和标记样本进行识别实验. 实验结果表明,该模型能够识别无线路由器、网络摄像头和智能交换机等终端设备,并具有较高的识别准确率和较低的识别遗漏率.

关键词 网络空间; 终端设备; K-means; 余弦测度; 指纹识别

中图分类号: TP393 **文献标志码:** A **doi:** 10.7523/j.issn.2095-6134.2016.04.019

Cyberspace device identification based on K-means with cosine distance measure

CAO Laicheng[1], ZHAO Jianjun[1,2], CUI Xiang[2], LI Ke[2,3]

(1 *School of Computer and Communication, Lanzhou University of Technology, Lanzhou* 730050, *China*;
2 *Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093, *China*;
3 *School of Computer Science, Beijing University of Posts and Telecommunications, Beijing* 100876, *China*)

Abstract Since the traditional web fingerprinting methods are limited to identification of mainstream web server softwares, a kind of cyberspace device identification model based on K-means with cosine distance measure is proposed. Firstly, identification model is designed and verification method is determined. Secondly, the header fields and the status code of HTTP response are selected as characteristics of terminal device and then the characteristics are transformed into 32-dimensional feature vector by feature extraction and vectorization. Thirdly, cosine distance function is selected as similarity measuring function in K-means. Finally, experiment algorithm process is designed according to the identification model and the experiments for unlabeled samples and labeled

* 国家自然科学基金(61562059, 61461027)资助

† 通信作者, E-mail: zhaojianjun@hotmail.com

samples are carried out. The results show that the identification model works for many kinds of terminal devices, including wireless router, web camera, and intelligent switch, and has high accuracy rate and low omission rate.

Key words　cyberspace; terminal device; K-means; cosine measure; fingerprinting

随着大数据、IoT(Internet of Things) 技术的发展，越来越多的终端设备接入网络空间. 数字视频摄像机、VoIP 网络电话、网络打印机、智能交换机等新型终端设备和传统的服务器、路由器共同构成新的网络环境. 当前网络空间中的终端设备具有规模庞大、类型复杂的特点. 据统计，除普通网站和主机外，接入网络空间的终端设备数量已超过 500 万，大类超过 20 种[1]. 复杂的网络环境带来了更多的安全隐患，新型的黑客攻击可能针对一个无线路由器以重置路由信息、获得管理员权限、窃取用户隐私流量等[2-4]，甚至攻击大型工业控制系统扰乱工业生产[5-7].

终端设备存在的安全漏洞时刻威胁着用户的隐私安全和财产安全，对终端设备进行漏洞检测与风险评估是保障网络空间安全的基本途径之一. 特定的安全漏洞往往威胁某一特定类型或版本的网络设备. 因此，进行安全评估和安全预警就要求能够对终端设备的类型及其型号进行准确的识别. 以往的识别技术仅针对 Apache、IIS 等传统的 Web 服务器软件，并不能完全适用于新的网络环境. 本文在此方向上探索和研究新的终端设备识别方法，并为网络空间终端设备安全评估和安全预警提供思路.

1　研究现状与相关工作

近年来，网络指纹识别技术已成为网络安全领域的研究热点，国内外学者提出了诸多网络指纹识别的理论和方法.

文献 [8] 提出通过服务标识(Banner) 来识别 Web 服务器软件的方法. 由于部分终端设备的 HTTP 返回包中并不含有 Banner 信息，因此该方法在识别终端设备时具有一定局限性. 文献 [9] 提出一种不依赖 Banner 信息的识别方法，即通过返回的某些原因短语差异和超长 URL 处理方式差异来识别. 但此种方法可能会增加终端设备的处理负担，造成拒绝服务，或被防火墙等设备判定为攻击行为，引发报警. 文献 [10] 提出一种类似识别 TCP/IP 栈指纹[11] 的 Web 指纹识别方法. 该方法通过构造畸形的 HTTP 请求，并根据不同 Web 服务器软件处理方式的差异进行识别. 但此种差异的数量是有限的，不同类型和型号终端设备的数量远大于此种差异的数量，通过这种方法来识别将会出现重复，导致误判. 文献 [12] 向 Web 服务器分别发送 15 种畸形的 HTTP 请求，并以返回的状态码作为输入，构造朴素贝叶斯分类器进行分类，实现对主流 Web 服务器软件的识别，但未能实现对无线路由器、IP 摄像头、智能交换机等其他终端设备的识别.

在此前的研究中，识别对象仅为传统的 Web 服务器软件. 而如今的网络空间中终端设备类型复杂、数量繁多，传统的识别方法难以保证对新型网络空间终端设备的识别准确率. 因此，本文提出一种基于 K-means 聚类算法的网络空间终端设备识别模型. 该模型通过选取适当的 HTTP 头字段是否存在作为特征，以余弦距离测度作为相似性度量测度，在大量无标记样本中加入标记样本，通过无监督机器学习的方法对终端设备进行识别，解决了传统 Web 指纹识别方法中识别对象局限于主流 Web 服务器的缺陷，实现了对无线路由器、IP 摄像机、智能交换机等新型终端设备的识别.

2　K-means 聚类模型

K-means 聚类算法是一种无监督学习方法，通过将研究对象样本集按相似性准则划分为若干簇，最终达到簇内相似、簇间相异的聚类效果[13]. 在聚类过程中，首先人工确定聚类数 K，并在样本集中随机选取 K 个样本作为初始聚类中心; 对样本集中的每个样本，计算其与所有聚类中心的相似度，并将其划分给最相似的簇; 然后重新计算各簇的簇内平均值作为新的聚类中心. 整个过程重复进行，直到聚类准则函数收敛.

算法步骤如下:

1) 设给定样本集 $X = \{x_1, x_2, \cdots, x_n\}$，其中 x_i 为第 i 个样本的 d 维特征向量; 给定聚类数 K，并在样本集 X 中随机选取 K 个初始聚类中心，记为

$c_1, c_2, \cdots, c_K$;

2) 对给定样本集的 n 个样本，根据相似性度量函数计算它们与各聚类中心的相似性程度，并按相似性程度划分为 K 个簇 $C_1, C_2, \cdots, C_K$;

3) 分别计算各簇的簇内平均值，作为新的聚类中心;

4) 计算聚类准则函数

$$J = \sum_{j=1}^{K} \sum_{x_i \in C_j} d(x_i, c_j), \quad (1)$$

式中，c_j 表示簇 C_j 的聚类中心，$d(x_i, c_j)$ 表示相似性度量函数;

5) 若聚类准则函数收敛，则终止算法; 否则重复执行步骤 2) 至步骤 4)，直至聚类准则函数收敛;

6) 聚类准则函数收敛时，J 值最小，聚类效果最优.

3 网络空间终端设备识别模型

3.1 识别模型

设计识别模型如图 1 所示，该模型主要包括 4 大功能模块.

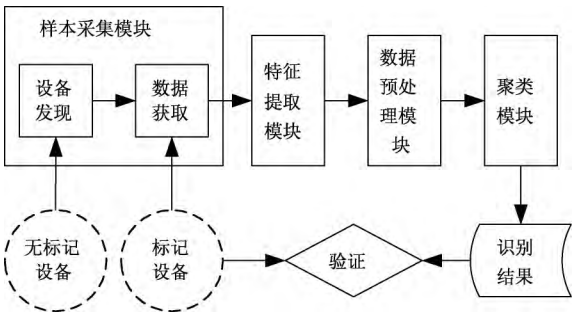

图 1 网络空间终端设备识别模型

Fig. 1 Cyberspace device identification model

样本采集模块: 负责在整个 IPv4 地址空间中进行端口扫描，获取开放 Web 服务的无标记设备的 IP 地址，加入已知类型的标记设备的 IP 地址后形成设备 IP 地址集; 向 IP 地址集中所有 IP 地址发送 HTTP-GET 请求以获取完整的 HTTP 返回包头部作为原始样本.

获取到的原始样本的信息可能为如下形式:

Code: 200
Date: Wed Oct 14 18:24:11 2015
Server: DVRDVS-Webs
Last-modified: Fri Dec 21 08:25:34 2012
Content-length: 1577
Content-Type: text/html

或

Code: 401
Server: Router Webserver
Connection: close
WWW-Authentication: Basic realm = "TP-LINK Wireless N Router WR841N"
Content-Type: text/html

特征提取模块: 提取 HTTP 返回包头部中能够反映终端设备特性的信息作为样本特征，去除冗余信息，以便降低计算复杂度，提高识别效率.

经提取后的样本信息可能为如下形式:

Code: 200
Date
Server
Last-modified
Content-length: 1577
Content-Type

或

Code: 401
Server
Connection
WWW-Authentication
Content-Type

数据预处理模块: 对提取后的样本特征进行预处理，将文本类型的特征数据转化为数值型，对特征数值进行向量化，使其能够适用于聚类算法.

经数据处理后的特征信息可能为如下形式:

0 1 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1577

或

0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 10 0

聚类模块: 以样本的多维特征向量作为输入，使用 K-means 聚类算法对数据进行划分，输出为聚类后的若干簇.

验证: 首先，根据标记设备划分情况对聚类结果分配标签; 其次，验证是否所有相同类型的标记样本都被划分到同一簇、同一个簇中与标签类型相同的终端设备占总数的百分比.

给出标签分配规则如下:

定义 3.1 标签分配规则 对于设备类型 Y，若所有 Y 型标记样本均被划分至同一簇，则定义该簇的类型为 Y; 否则，定义具有 Y 型标记样本数量最多的簇的类型为 Y.

3.2　特征选取

接收到 HTTP-GET 请求后，不同的终端设备返回的响应包间具有一定的差异，差异体现在返回包中头部组成不同、返回状态码不同、返回包整体长度不同、返回包内容不同；而同一类型的终端设备返回的 HTTP 包具有一定的相似性. 因此，根据 HTTP 返回包中的这些差异特征能够对终端设备的类型进行识别. 选择较好的特征集以及采用较好的特征处理方法能够使识别模型获得更好的识别性能.

本文在整个 IPv4 地址空间中随机选取20 000 个开放 Web 服务的 IP 地址进行抽样调查. 分别发送 HTTP-GET 请求，统计 HTTP 返回包中的头字段总数，选取出现频率最高的 30 个头字段（不含“content-length”字段）作为特征 1 至特征 30；选取 HTTP 返回状态码及“content-length”字段值作为特征 31 和特征 32，如表 1 所示. 其中 x_i 表示第 i 个样本的特征向量，x_{ij} 表示第 i 个样本的第 j 个特征.

表 1　选取的样本特征

Table 1　Selected sample features

x_i	字段	x_i	字段
x_{i1}	ext	x_{i17}	pragma
x_{i2}	content-type	x_{i18}	www-authenticate
x_{i3}	connection	x_{i19}	location
x_{i4}	date	x_{i20}	x-frame-options
x_{i5}	server	x_{i21}	x-cache
x_{i6}	expires	x_{i22}	content-language
x_{i7}	cache-control	x_{i23}	via
x_{i8}	last-modified	x_{i24}	x-ua-compatible
x_{i9}	set-cookie	x_{i25}	p3p
x_{i10}	accept-ranges	x_{i26}	content-location
x_{i11}	mime-version	x_{i27}	x-aspnet-version
x_{i12}	etag	x_{i28}	link
x_{i13}	x-powered-by	x_{i29}	x-pingback
x_{i14}	transfer-encoding	x_{i30}	cf-ray
x_{i15}	age	x_{i31}	状态码
x_{i16}	vary	x_{i32}	content-length

对原始特征进行预处理和向量化后才能作为聚类算法的输入，本文采用的处理方式如下：

1) 若 HTTP 返回包头部中存在特征 1 至特征 30 中的头字段，则相应位置的数值为 1；若不存在，数值为 0；

2) 返回的 HTTP 状态码根据状态码集 S 中的索引，特征 31 数值标记为 1 ~ 36 中某值.

状态码集 S 为

$$S = \{200,202,203,204,205,301,302,307,400,401,402,403,404,405,406,407,408,410,412,416,451,456,461,479,500,501,502,503,504,508,510,520,534,535,550,596\}$$

3) 若 HTTP 返回包头部中存在头字段“content-length”，则特征 32 数值为“content-length”字段具体数值，若不存在，数值为 0.

特征预处理和向量化伪代码如下：

一个原始 HTTP 返回包 *header* 经过数据预处理模块 *feature* 为样本特征集，如表 1 所示；S 为状态码集；输出 *featureVector* 为 32 维特征向量.

```
    //feature 1 - 30
1   for i from 1 to 30
2       if feature[i] in header then
3           featureVector[i] ←1
4       else
5           featureVector[i] ←0
6       end if
7   end for
    //feature 31
8   for x from 1 to 36
9       if code(header) = S[x] then
        // code() return the HTTP status code of the header
10          featureVector[31] ←x
11      end if
12  end for
    //feature 32
13  if content-length in header then
14      featureVector[32] ←content-length(header)
        //content-length() return the value of content-length field
15  else
16      featureVector[32] ←0
17  end if
18  return featureVector
```

最终，将原始样本 *HTTP* 返回包文本特征转换为 *32* 维特征向量，为聚类模型提供输入.

3.3　相似性度量函数

在 K-means 聚类过程中，2 个样本间的相似性计算非常重要，相似性度量函数的优劣决定最终的聚类效果. 在向量空间模型下，可以借助向量之间的某种距离表示样本间的相似度. 目前，研究者已提出多种方法来评价同一个特征空间中 2 个特征向量间的距离，然而并非所有的距离测度在各种情况下都适用. 文献 [14] 指出，余弦距离测度和谷本距离测度相比欧氏距离测度等更适合作为文本文档的相似性测度. 本文选取的特征来源于 HTTP 返回包头部信息，经文本文档特征提取

和向量化得到. 因此,本文选取余弦距离函数作为相似性度量函数.

特征向量 x_i 与聚类中心 c_j 的余弦相似度计算如下:

$$\cos\theta = \frac{x_i c'_j}{\sqrt{(x_i x'_i)(c_j c'_j)}}, \quad (2)$$

根据式(2)得出相似性度量函数

$$d(x_i, c_k) = 1 - \frac{x_i c'_j}{\sqrt{(x_i x'_i)(c_j c'_j)}}, \quad (3)$$

由式(3)可知,特征向量 x_i 与聚类中心 c_j 的余弦相似度越高,相似性度量函数 $d(x_i, c_j)$ 的值越小.

根据式(3)得出聚类准则函数

$$J = \sum_{j=1}^{K} \sum_{x_i \in C_j} d(x_i, c_j) = \sum_{j=1}^{K} \sum_{x_i \in C_j} \left(1 - \frac{x_i c'_j}{\sqrt{(x_i x'_i)(c_j c'_j)}}\right). \quad (4)$$

由式(4)可知,当聚类准则函数收敛时,函数值 J 达到最小,即所有簇的簇内元素与簇中心的余弦相似度均达到最高,聚类效果最优.

下面对余弦距离测度和欧氏距离测度下 K-means 聚类效果进行讨论.

设有 3 种类型的终端设备 A,B,C 各 4 个,2 个 A 设备由于语言支持不同,第 32 个特征出现了 10 字节的偏差,B 设备没有 content-length 字段. 12 个设备的前 31 维特征向量均为:

1 0 1 1 1 0 1 1 0 0 0 0 1 0 0 1 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 1

在第 32 个特征不同的 4 种情况下,对 12 个终端设备在不同距离测度下进行 30 次聚类实验,正确聚类次数结果如表 2 所示.

表 2 不同距离测度下聚类结果

Table 2 Clustering results for different distance measures

第 32 个特征	正确聚类次数	
	欧氏距离	余弦距离
A: 90/100; B: 0; C: 10	19	30
A: 990/1 000; B: 0; C: 10	22	30
A: 9 990/10 000; B: 0; C: 10	23	30
A: 99 990/100 000; B: 0; C: 10	23	30

由表 2 可知,无论在各设备的第 32 个特征差别极大或极小的情况下,2 种 A 设备相差 10 字节,差距较小;而 B 设备和 C 设备相差 10,实际上是一个字段的有无,差距较大,此时采用余弦距离测度进行聚类的结果正确率要明显高于采用欧氏距离. 因此,对于本文选取的特征向量,余弦距离函数更适合作为 K-means 聚类中的相似性度量函数,证明本文选取余弦测度的合理性和先进性.

3.4 算法描述

根据网络空间终端设备识别模型,给出实验相应算法:

1) 扫描 IPv4 空间中开放 Web 服务的 IP 地址,加入已知设备类型的 IP 地址,形成 IP 地址集;

2) 获取原始样本集 $D = \{d_1, d_2, \cdots, d_n\}$,包括标记样本和无标记样本,其中 d_i 为第 i 个原始样本的 HTTP 头部;

3) 对原始样本 d_i 进行特征提取、预处理和向量化;

4) 得到样本集 $X = \{x_1, x_2, \cdots, x_n\}$,其中 x_i 为第 i 个样本的 32 维特征向量;给定聚类数 K,并在样本集 X 中随机选取 K 个初始聚类中心,分别为 $c_1, c_2, \cdots, c_k$;

5) 对给定样本集的 n 个样本,按照它们与各聚类中心的余弦相似性划分为 K 个簇 $C_1, C_2, \cdots, C_K$;

6) 分别计算各簇内平均值,作为新的聚类中心;

7) 根据式(4)计算聚类准则函数 J;

8) 若聚类准则函数 J 收敛,则终止算法;否则重复执行步骤 4) 至步骤 6),直至聚类准则函数 J 收敛;

9) 得到最终聚类结果簇 $C_1, C_2, \cdots, C_K$,按标记样本划分情况分配标签;

10) 根据标签对簇内元素进行识别准确度验证.

4 实验及结果分析

4.1 实验准备

本文在整个 IPv4 地址空间中随机选取50 000 个开放 80 端口的 IP 地址作为无标记样本,随机选取避免了终端设备局部相似的现象,获取的研究对象样本更加合理;另选取 50 个 IP 地址作为标记样本,其中共有 5 种已知类型的终端设备,每种 10 个.

对 50 050 个 IP 地址发送 HTTP-GET 请求,获取 HTTP 返回包头部,对头部数据进行特征提取和向量化,将样本原始数据转换为 32 维特征向量. 将 50 000 个无标记样本和 50 个标记样本重组

为样本集 1、样本集 2、样本集 3,分别含有 30 050、40 050、50 050 个样本.

4.2　实验结果

按照本文给出的算法流程,对样本集 1、样本集 2、样本集 3 分别取 $K = 3\,000$、$K = 4\,000$、$K = 5\,000$ 进行识别实验.

根据定义 3.1 中标签分配规则,为聚类结果分配标签. 对 5 个有标签簇进行统计后得到标记样本识别数和簇样本数结果如表 3 所示.

表 3　有标签簇样本数

Table 3　Sample number of labeled clusters

设备类型 Y	标记样本识别数/簇样本数		
	样本集 1	样本集 2	样本集 3
TP-Link 无线路由器	10/69	10/87	10/103
Hikvision 摄像机	10/91	10/111	10/113
MikroTik 路由器	10/28	10/36	10/41
D-Link VoIP 网关	10/26	10/28	10/28
NetGear 智能交换机	10/20	10/28	10/32

样本识别准确率计算方法如下:

定义 4.1　样本识别准确率　设聚类结果中某簇按照标签分配规则被标记为 Y 型,则 Y 型设备的样本识别准确率 $\mathrm{Ra}(Y)$ 为

$$\mathrm{Ra}(Y) = \frac{Y\text{型簇中}Y\text{型样本数}}{Y\text{型簇中样本总数}}. \tag{5}$$

对聚类结果进行人工验证后得到识别准确率如表 4 所示.

表 4　有标签簇识别准确率

Table 4　Identification accuracy for labeled clusters

设备类型 Y	识别准确率 $\mathrm{Ra}(Y)$ /%		
	样本集 1	样本集 2	样本集 3
TP-Link 无线路由器	69.57	63.22	64.08
Hikvision 摄像机	100.00	99.10	100.00
MikroTik 路由器	100.00	100.00	100.00
D-Link VoIP 网关	100.00	100.00	100.00
NetGear 智能交换机	85.00	89.29	90.63

样本识别遗漏率计算方法如下:

定义 4.2　标记样本识别遗漏率　设聚类结果中某簇按照标签分配规则被标记为 Y 型,且 Y 型标记样本不全在 Y 型簇内,则 Y 型设备的标记样本识别遗漏率 $\mathrm{Ro}(Y)$ 为

$$\mathrm{Ro}(Y) = \frac{\text{非}Y\text{型簇中}Y\text{型标记样本总数}}{Y\text{型标记样本总数}}. \tag{6}$$

由于同一种类型的终端设备的标记样本和无标记样本并无差别,标记样本可视为所有同类样本的抽样,因此标记样本识别遗漏率能在一定程度上反映总体识别遗漏情况.

由表 3 可知,3 次实验中,同种类型的标记样本始终被划分至同一簇,因此各种设备的标记样本识别遗漏率 $\mathrm{Ro}(Y)$ 均为 0.

对聚类结果中的无标签簇进行分析后发现,多个无标签簇中含有大量同种类型的终端设备,对于无标签簇,需人工后验进行标签分配,分配规则如下:

定义 4.3　后验标签分配规则　对于设备类型 Y,若某无标签簇中 Y 型设备所占比例最高,则定义该簇的类型为 Y.

经人工后验统计筛选并分配标签后得到样本数结果如表 5 所示.

表 5　后验标签簇样本数

Table 5　Sample number of posterior labeled clusters

设备类型 Y	簇内样本数		
	样本集 1	样本集 2	样本集 3
SonicWALL 防火墙	8	11	16
Synology DSM 网络存储	12	21	27
GeoVision 摄像机	16	23	25
华为 HG532E 无线路由器	81	81	81
AirOS 无线操作系统	290	324	366
HP 网络打印机	46	50	52
Cisoc IOS	160	243	292
ZyXEL 家庭网关	56	58	60
HiSilicon 摄像机	77	105	122

根据式(5)计算识别准确率如表 6 所示.

表 6　后验标签簇识别准确率

Table 6　Identification accuracy for posterior labeled clusters

设备类型 Y	识别准确率 $\mathrm{Ra}(Y)$ /%		
	样本集 1	样本集 2	样本集 3
SonicWALL 防火墙	100.00	100.00	100.00
Synology DSM 网络存储	100.00	100.00	100.00
GeoVision 摄像机	87.50	86.96	84.00
华为 HG532E 无线路由器	100.00	100.00	100.00
AirOS 无线操作系统	100.00	100.00	100.00
HP 网络打印机	71.74	70.00	69.23
Cisoc IOS	99.38	99.59	98.97
ZyXEL 家庭网关	100.00	100.00	100.00
HiSilicon 摄像机	100.00	100.00	100.00

4.3　实验分析

根据以上实验结果中表 3 可知,对于一种类型的终端设备,随着样本数量的增加,识别出的设

备总数呈递增趋势，并且该类型的所有标记样本始终被成功划分至同一类. 因此证明本文提出的识别模型能够成功识别出多种类型的终端设备，并且具有较低的识别遗漏率.

由表 4 可知，除 TP-Link 无线路由器外，该模型对其余几种终端设备均能达到较高的识别准确率. 对被标记为 TP-Link 无线路由器的簇进行分析后发现，该簇内包含其他类型的终端设备，列举 2 种被误划分至同一簇的终端设备如下:

DIGI PortServer TS4 MEI 串口通信服务器:

Code: 401
WWW-Authenticate: Basic realm = "PortServer TS 4 MEI"
Content-Type: text/html
Transfer-Encoding: chunked
Server: Allegro-Software-RomPager/3. 12
Connection: close

APC UPS 网络管理卡:

Code: 401
WWW-Authentication: Basic-realm = " APC Management Card"
Content-Type: text/html
Transfer-Encoding: chunked
Server: Allegro-Software-RomPager/2. 10
Connection: close

特征向量分别为:

0 1 1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 10 0
0 1 1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 10 0

标记样本 TP-Link 无线路由器的特征向量为:

0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 10 0

可见，该簇内各设备的 HTTP 返回包头部字段个数较少，且不同样本间的特征差异极小，结果导致聚类器误判，致使识别出的设备总数过多，识别准确率下降.

由表 5 和表 6 可知，后验标签簇中的样本数随着样本总数的增加呈递增趋势，且均具有较高的识别准确率. 可见，该模型在遇到未知标签设备时仍然能够具有较好的识别率，证明了该方法的普适性.

5 讨论

本文提出的模型，在识别网络空间终端设备时具有一定的先进性. 与文献 [12] 相比，在探测稳健性方面，本文使用的探测流量均为正常的 HTTP 请求，而非构造的畸形请求. 畸形的请求对识别对象具有一定的危害性，容易造成拒绝服务，并且畸形的请求在探测防火墙等设备时会被判定为攻击行为，造成探测误差; 在识别对象方面，文献 [12] 实现了对主流 Web 服务器软件的识别，而本文针对的识别对象理论上能够覆盖大部分类型和版本的终端设备，包括无线路由器、IP 摄像头、VoIP 网关等; 在指纹储备方面，普通模型需要维护一个较大的指纹库，且对于未知指纹，识别能力较差，本文模型采用无监督学习方法，可事先加入标记样本，同时支持对聚类结果进行后验，即能够在聚类结束后再给簇添加相应的标签，具有较高的普适性. 本文模型与其他模型相比具有的先进性如表 7 所示.

表 7 本文模型先进性

Table 7 Advantages of the proposed identification model

	其他模型	本文模型
流量	畸形请求	正常请求
对象	Web 服务器软件	网络空间终端设备
指纹	指纹库	标记样本标签、后验标签

当然，该模型还存在一些不足和待优化之处，比如目前所针对的识别对象均为开放 HTTP 服务的终端设备，而对于只能通过 Telnet、SSH 等其他方式进行访问的终端设备，还需要进一步研究适用的网络指纹; 对于最优聚类数 K 值的确定，还需要进一步研究网络空间终端设备总类型数的估算方法; 对于个别类型的终端设备，该模型的识别误判率较高，还需进行大量研究以继续优化样本特征向量，这也是本文接下来的研究重点.

6 结束语

网络空间中终端设备的数量一直呈不断增长的趋势，因此，终端设备识别技术将会是一种非常实用而且必需的技术. 该技术对于全面统计终端设备的分布情况以及预警设备潜在威胁具有很强的实用意义. 本文提出的基于余弦测度下 K-means 的网络空间终端设备识别模型，充分考虑了终端设备类型复杂、数量繁多的特点，通过从终

端设备返回包的 HTTP 数据包头字段和状态码中提取特征信息，并使用以余弦距离函数作为相似性度量函数的 K-means 聚类算法，成功实现了对网络空间终端设备的识别，为网络空间终端设备安全评估和安全预警提供思路和理论支持.

参考文献

[1] ZoomEye. 网络设备统计分析[EB/OL]. (2015-12-31)[2015-12-31]. https://www.zoomeye.org/statistic/device.

[2] Gallagher S. Backdoor in wireless DSL routers lets attacker reset router, get admin[EB/OL]. (2014-01-03)[2015-12-31]. http://arstechnica.com/security/2014/01/backdoor-in-wireless-dsl-routers-lets-attacker-reset-router-get-admin/.

[3] Chirgwin R. Hacker backdoors Linksys, Netgear, Cisco and otheh routers[EB/OL]. (2014-01-06)[2015-12-31]. http://www.theregister.co.uk/2014/01/06/hacker_backdoors_linksys_netgear_cisco_and_other_routers/.

[4] 国家互联网应急中心. 关于多款 D-LINK 路由器产品存在后门漏洞的情况通报[EB/OL]. (2013-10-25)[2015-12-31]. http://www.cert.org.cn/publish/main/9/2013/20131025152943288740930/20131025152943288740930_.html.

[5] Singh D, Sinha R, Songara P, et al. Vulnerabilities and attacks targeting social networks and industrial control systems[J]. Eprint Arxiv, 2014, 4(1): 133-142.

[6] 彭勇，江常青，谢丰，等. 工业控制系统信息安全研究进展[J]. 清华大学学报: 自然科学版，2012，52(10): 1 396-1 408.

[7] 卢慧康. 工业控制系统脆弱性测试与风险评估研究[D]. 上海: 华东理工大学，2014.

[8] Shah S. An introduction to HTTP fingerprinting[EB/OL]. (2004-05-19)[2015-12-31]. http://net-square.com/httprint_paper.html.

[9] Lee D, Rowe J, Ko C, et al. Detecting and defending against Web-server fingerprinting[C]//CSAC 2002: 2002 Computer Security Applications Conference. United States: IEEE Computer Society, 2002: 321-330.

[10] 杨可新，鞠九滨. 利用 Web 指纹进行服务映射[J]. 计算机工程与应用，2004，40(4): 7-9.

[11] Fyodor. Remote OS detection via TCP/IP stack fingerprinting[J]. Phrack Magazine, 1998, 17(3): 1-10.

[12] 吴少华，孙丹，胡勇. 基于贝叶斯理论的 Web 服务器识别[J]. 计算机工程，2015，41(7): 190-193,198.

[13] 刘三民，孙知信，刘余霞. 基于 K 均值集成和 SVM 的 P2P 流量识别研究[J]. 计算机科学，2012，39(4): 46-48,74.

[14] 陈磊磊. 不同距离测度的 K-Means 文本聚类研究[J]. 软件，2015，36(1): 56-61.