HTTP 网络应用特征串的自动提取

吴昊[1,2]，程光[1,2]

(1.东南大学 计算机科学与工程学院，江苏 南京 210096；
2.网络和信息集成教育部重点实验室，江苏 南京 210096)

摘要：网络应用特征串的提取是网络流量分类中深度包检测（DPI）系统至关重要的一环。文中提出一种自动提取不同的 HTTP 网络应用（如网络视频、网络游戏、SNS 等）的特征串的方法。该方法先对 HTTP 网络应用的全报文数据进行组流分析，然后通过一个基于 hash 表的算法提取出频繁出现的字符串，最后通过建立 HTTP 应用共性特征库，对频繁串逐个匹配等方式筛选出该 HTTP 应用的个性特征字符串。

关键词：流量分类；自动提取特征串；共性与个性特征库；深度包检测
中图分类号：TP393.0　　**文献标识码：**A

Automatically extraction of HTTP application signatures

WU Hao[1,2]，CHENG Guang[1,2]

（1.College of Computer Science and Engineering, Southeast University ,Nanjing 210096, China;
2.Key Laboratory of Computer Network and Information Integration, Ministry of Education, Nanjing 210096, China）

Abstract: Extraction of network application's signatures is the key point of the Deep Packet Inspecting (DPI) system in the field of traffic classification. A method is provided to automatically extract the feature strings of different HTTP applications (network video, online games, SNS, etc). First, the full packet data of HTTP application was combined into flows. Then a hash based algorithm was taken out to extract the frequent strings. At last a common feature library was built to get individual feature strings out of frequent strings by matching each of them.

Key words: traffic classification; automatically extract signatures of applications; common and individual feature library; deep packet inspecting

HTTP 网络应用是指在应用层 HTTP 协议基础上的一系列网络应用。常见的基于 HTTP 的应用有网络视频、网络游戏、SNS 应用、网络博客、电子邮件等。网络应用的特征字符串是指在应用层载荷中固定位置出现的字节组合。特征字符串用以构建深度包检测（DPI）系统的特征库。特征字符串的获取是 DPI 系统的基础，而 DPI 系统又是获得网络分类所需的标准数据集的基础。

针对网络应用特征串的提取和 DPI 系统已经有了一些研究，但专门针对 HTTP 应用的特征串研究较少。刘兴彬等[1]利用 Apriori 算法从应用的报文载荷中提取特征串，但该理论过于复杂，算法执行效率较低。W.Li 等[2]利用其自己提取的特征串，使用深度包检测的方法对 HTTP 上的应用进行了分类，但他们的特征串并未公开。而且即便公开，由于时间不同、地域不同，也会导致很多网络应用的不同。比如，同属 SNS，在国外是 facebook 和 twitter，在国内是人人网和微博。同属视频应用，在国外是 YouTube，在国内是优酷、土豆等。所以，有必要开发一个方法，来对网络应用的特征串进行实时实地的提取。

本文先分析 HTTP 网络应用特征串的特点，然后针对其特点设计相应的方法来进行自动提取，最后通过实验来实现。

收稿日期:2011-09-06;**修订日期**:2011-09-28
基金项目：国家 973 项目（No.2009CB320505），国家自然基金项目（No.60973123）
通讯联系人：程光（1973-），男，安徽黄山人，东南大学教授，博士，博士生导师;E-mail: gcheng@njnet.edu.cn

1 HTTP 网络应用特征串的特点

不同的协议或不同的应用所对应的特征字符串一般不同，甚至随着时间的推移，同一个协议或应用的不同版本所对应的特征字符串都有可能不同。因此，获取网络应用的特征串是一项比较有挑战的工作，能获取到全面的、完整的、最新的特征串绝非易事。

在数据流层面，网络应用的特征串一般有如下的特点：

①不同的特征串长度可能不一样；

②同一个特征串可能在每条流中都出现，也有可能不是在每条流中都出现；

③同一个特征串有可能在每条流中出现一次，也有可能在同一个流的不同位置出现多次。具体来说有以下三种情况：

①同一个特征串有可能在每条流中只出现一次，位置较固定；

②同一个特征串在每条流中都出现 m 次，在每条流中的 m 个位置较固定；

③同一个特征串在不同的流中出现的次数不一样，出现的位置不固定。

以上是网络应用特征串的一般特点。而对于 HTTP 网络应用，除了拥有上述特点外，还拥有着自己的一些特点。这些特点分为两个方面：

①共性特征。因为 HTTP 网络应用首先是基于 HTTP 协议的，所以 HTTP 协议中的一些报文格式，这些应用都必须遵守。于是，相应的，一些固定出现在 HTTP 协议的请求报文和应答报文中的特征串（如“GET”、“HTTP”、“Host”等）就属于这些 HTTP 网络应用的共性特征。

②个性特征。除去共性特征，剩下的在网络应用流中固定位置频繁出现的字符串便是个性特征。此个性特征可以区分不同的 HTTP 网络应用。比如，不同的 URL、HOST 主机，不同的 Content-Type 等。

2 HTTP 应用的特征串提取方法

为了提取 HTTP 网络应用的特征串，先通过频繁串提取模块提取出某特定 HTTP 应用固定位置出现的频繁串，然后通过筛选模块筛选出属于该 HTTP 应用的个性特征。方法流程见图 1。

2.1 频繁串提取模块算法

本模块的功能是寻找同一种网络应用的不同流之间的共有的频繁出现的串，也就是频繁串。算法的输入是某种特定的网络应用的多条拥有全报文数据的流，算法的输出是在这些流中频繁出现的字符。算法的核心是把每条流都截取成为一系列定长字符串的集合，然后对这些定长的字符串进行 hash 运算映射到 hash 表中，如果某个 hash 表项被多次映射到，就说明映射之前的定长字符串有可能是一个频繁串。同时，本算法还针对上文中提到的网络应用特征串的特点进行如下的处理与优化：

针对特点①，本算法包含两个部分：一个是算法的基本功能模块，就是找到这些流中共有的固定长度字符串；另一个是拓展功能模块，在已经找到的频繁串两边的相邻字符串进行匹配，以期找到这个位置的最大长度的频繁串。算法的两部分相结合，就可以找到不同长度的频繁串。

针对特点②，在算法中会设定阈值，只有出现次数超过这个阈值的字符串才被认定是频繁串。

针对特点③，算法的拓展功能模块会在进行相邻字符串匹配之前，根据基本功能模块执行后的 hash 表项节点信息，判断映射到该节点的字符串属于哪种情况，然后会对不同的情况使用不同的算法处理。

算法的两个部分——基本功能模块和拓展功能模块，是顺序执行的关系，先执行第一部分，再执行第二部分。两个部分中，都需要原始的报文载荷信息。要是将流文件全部放入缓存，则需要较大的开销。因此，为了节省内存，用时间换空间，本算法采取两次遍历流文件的方法，每次只缓存一个流的报文，来为算法的两个部分提供报文载荷。

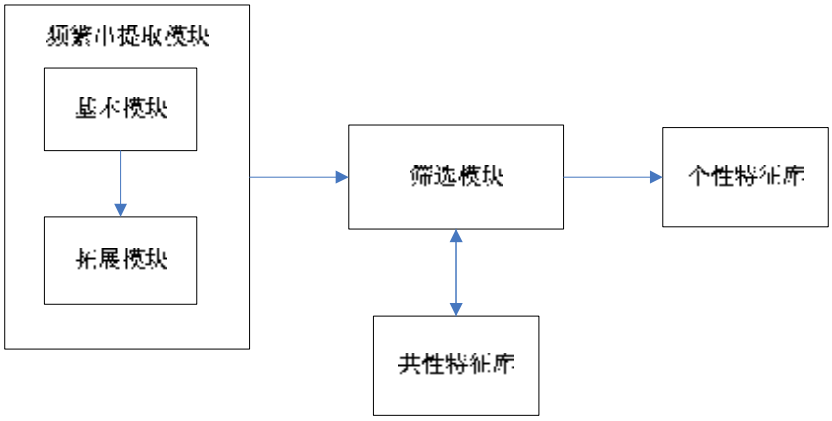

图 1　特征串提取方法流程
Fig.1 Process of signature extraction

2.2 个性特征筛选模块

经过频繁串提取模块后，会得到大量的频繁串。本模块就是对这些频繁串进行筛选。先建立一个 HTTP 共性特征库，然后对每一个得到的频繁串，放到库中进行查询，如果查到，则该频繁串不是特征串；否则，就保留并输出至个性特征库。

3　实验结果与分析

3.1 实验结果

源数据为 CERNET 江苏省网到国家主干链路上采集的 2008 年 11 月某天下午半个小时的全报文数据。从中筛选出某些主流视频网站的视频流数据，组流，作为下面特征提取算法的输入。本算法使用 C++语言编写，hash 表采用 C++ technical report 1 中的 unordered_map，是下一代的 C++标准库，用 g++编译时需加上-std=c++0x 选项。 实验环境：Linux，2.6.28-19-generic #66-Ubuntu，内存 2G，CPU 4 核 1.9GHz。

对于每个流，我们提取最开始的 1000 个载荷字节存入缓存。阈值设为 1，即映射次数大于一次的字符串就认为其是频繁字符串（实验结果说明这个阈值虽小，但是合理）。基本算法模块中定长字符串，定长取 4。拓展模块中左右固定长度相邻字符串长度取 10，即本算法可以匹配找到长度为 24 以下的频繁串。

对于共性特征库，参考 HTTP 协议标准[3]，对于 HTTP 的两类报文（GET 报文和 RESPONSE 报文）中固定出现的字段（如 GET 报文中的 GET、HEAD、POST、PUT 等十余种常用方法、RESPONSE 报文中成功请求必然会出现的“200 OK”等）加入共性特征库，并用正则表达式表示如下：

```
HTTP/(0\.9|1\.0|1\.1) [1-5][0-9][0-9] [\x09-\x0d -~]*
(connection:|content-type:|content-length:|date:)|post
[\x09-\x0d -~]* HTTP/[01]\.[019]。
```

试验中，经过频繁串提取模块后，我们获得了大约 700 余个频繁串。经过筛选模块后，我们获得了大约 50 个特征串。将部分特征串以正则表达式的形式列举如下：

```
^GET /youku/.*\.flv HTTP/1.1$
^Content-Type: video/x-flv$
^GET /.*\.f4v HTTP/1.1$
^GET /.*\.hlv HTTP/1.1$
^Content-Type: video/mpeg$
^Content-Type: video/x-msvideo$
^Content-Type: video/x-sgi-movie$
```

^HTTP://static.youku.com/.*/v/swf/player.swf$

……

同时，我们也对中间结果——筛选前的频繁串的长度进行了统计，用以评估我们的系统。统计结果如图 2，length 值即为频繁串的长度。x 轴为不同的 length 值，y 轴是具有该 length 值的频繁串的数目。

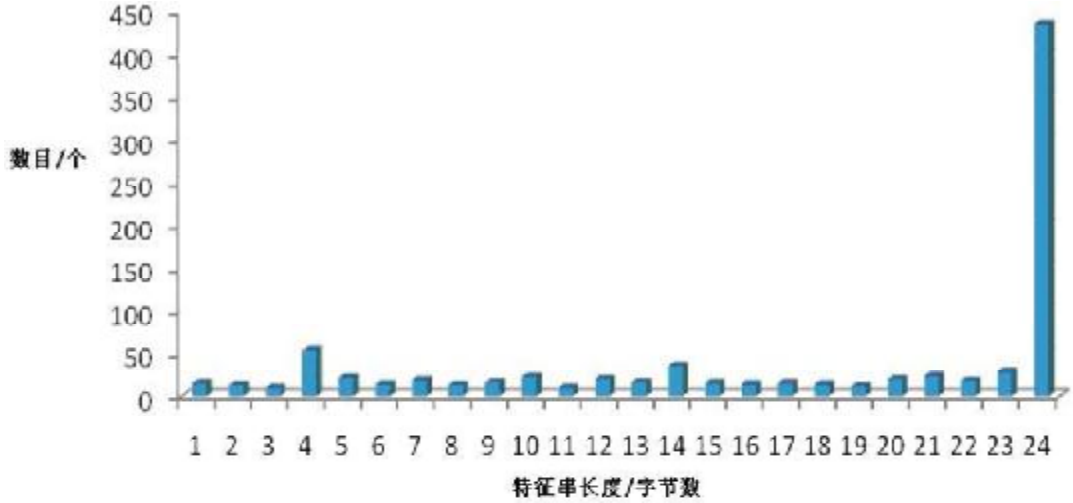

图 2 具有特定 length 值的频繁串数

Fig.2 Number of frequent strings that contain certain number of bytes

3.2 实验结果分析

对于 HTTP 网络视频的特征串，我们发现有很多特征是以下情况：请求报文中包含特定的主机域名；特定的视频后缀名；应答报文中特定的内容类型。其实，对于 HOST 字段（主机域名）和 Referer（本页面之前的链接页面）字段，也是有可以挖掘的地方。不过这两个字段的内容随时间变化大，只适合作为某几天甚至某几个小时的临时特征串,所以没有在结果中列举。如果对特征串的实时性要求较强，而对其后效性或者其后续影响要求较低，那么，这两个字段的特征串也可以使用，而且会更有针对性。

通过对图 2 的观察可以发现，除了 7.9%的频繁串具有长度 4、5 以外，大部分（45.7%）的频繁串长度都是 24，而 24 是我们之前定的能取到的最大的特征串的长度。这说明大部分的频繁串长度较大，很有可能 24 都不够。这个现象就是因为 HTTP 协议格式是固定的，会有大段的长的共性特征串的出现。

这个中间结果的统计也证明了筛选模块的重要性。因为如果没有筛选模块，这些大段的“特征串”都会保留，但是却对识别某个特定的 HTTP 网络应用毫无意义。经过筛选模块后，很多长度为 24 的频繁串就会被去除，剩下的频繁串长度分布就会比较均匀。图 2 中在 4 和 14 的长度位置有两个小高峰，说明有较多的字符串长度是 4 和 14。这也印证了我们一开始在算法基本模块中将定长字符串的长度设为 4 的正确性。

4 结论

本算法通过频繁串提取与个性特征筛选两个模块，可以有效地自动提取出某类特定 HTTP 网络应用的特征字符串。但本算法基本功能模块的字符串与拓展功能的相邻字符串都是选取的定长，这样，在实验一开始的时候，只能凭借经验来确定长度的取值。但所确定的长度有可能过小，也有可能过大，这有待于进一步的工作来解决定长参数自动配置的问题。

参考文献：

[1] 刘兴彬,杨建华,谢高岗.基于 Apriori 算法的流量识别特征自动提取方法[J].通信学报,2008(12)：51-59.

[2] LI W,MOORE A,CANINI M. Classifying HTTP traffic in the new age. ACM SIGCOMM 2008,Poster,August 2008.

[3] RFC 2616,Hypertext Transfer Protocol HTTP/1.1[S].New York: IETF, Dec 2006