

一种高效的 TCP 会话数据流重组算法及应用

赵启升, 李存华

(淮海工学院 计算机工程学院, 江苏 连云港 222005)

摘 要: 在网络行为管理系统中, 当数据包以 TCP 流方式在网络上传输时, 传统的基于数据包的模式匹配算法已无法满足需要. 对此提出了一种高效的 TCP 会话数据流重组算法, 详细论述了多连接管理、重复报文处理以及无序报文处理等重组过程中出现问题的解决方法, 并利用此算法实现了 FTP 协议数据包的截获与分析. 通过与 Libnids 库相比较, 试验结果表明, 该算法更具效率和稳定性.

关键词: 网络行为管理; TCP 协议; 报文重组; FTP

中图分类号: TP31

文献标识码: A

文章编号: 1000-7180(2010)07-0129-04

An Efficient Algorithm and Application for Restructuring TCP Session Data Streams

ZHAO Qi-sheng, LI Cun-hua

(School of Computer Engineering, Huaihai Institute of Technology, Lianyungang 222005, China)

Abstract: The traditional algorithms for packets-based pattern matching are unable to meet the needs of the network behavior management system if data packets flow through the network in TCP streams. After presenting an efficient algorithm for restructuring TCP session streams, this paper discusses detailedly solution to the problems appeared in multiple connection management, repeatable message processing, out-of order message processing and etc. Finally, we implement the interception and analysis of data packets complying to the FTP protocol by means of the algorithm. Comparing with the Libnids library, the experimental results demonstrates that our algorithm is more efficient and stable.

Key words: network behavior management; TCP protocol; segments reorganization; FTP

1 引言

随着互联网的普及, 各式各样的网络犯罪层出不穷^[1]. 对于企业而言, 管理人员总是希望有一种有效的手段监控企业员工的网络行为, 对各种非法的网络行为及时报警, 防止商业、技术机密的外泄. 或者在机密外泄后能迅速定位责任人, 从而把损失降到最低. 然而, 一旦数据经过加密或压缩, 并以 TCP 数据流的形式在网络上传送, 便给信息的提取带来极大地难度. 在这种情况下, 传统的基于数据包的模式匹配算法便无法满足现实的应用. 对此, 本文详细论述了一种 TCP 数据流的高效重组算法, 通过

对 TCP 会话进行完整的跟踪和还原, 为各种应用层协议分析系统的开发打下了基础.

2 概述

TCP 协议 (Transmission Control Protocol) 是 TCP/IP 协议族中一个重要的组成部分, 成百上千的应用层协议使用了 TCP, 例如 HTTP、FTP 等. 因此, TCP 数据流的重组是高层协议分析系统设计和实现的基础. TCP 协议是面向连接的可靠传输协议, 而 TCP 之下的 IP 协议却是面向报文的不可靠协议, 这就带来一个问题: IP 不能保证 TCP 报文段可靠地、顺序地交付到对方主机. 为了解决这一问

题,以及处理通信双方非平衡的发送与接收速率问题,TCP 采取了一系列措施,例如滑动窗口机制,为字节流编号的机制,快速重传算法等.这保证了数据的可靠交付,却也为接收方对数据流的重组带来一定的难度.

3 基于哈希表和局部性原理的快速定位方法

在以太网环境中,任意时刻可能同时存在成百上千的 TCP 连接.对于一个新到达的报文,必须尽可能快地定位并交付给对应的链接.由于 TCP 链接数量过多,若采用线性遍历的方式会在定位 TCP 链接时浪费大量的时间,无法满足高流量的网络环境的应用.因此,我们使用哈希表对 TCP 链接节点进行管理.对于任意一个 TCP 链接,都存在一个唯一的四元组:〈源 IP 地址,目的 IP 地址,源端口,目的端口〉,以其作为参数构造哈希函数.对于哈希冲突,我们把索引号相同的 TCP 链接节点用链表组织起来.哈希表长度的取值应结合网络的实际负载,以及系统的可用内存资源,选取最为恰当的经验值.哈希表存储结构如图 1 所示^[2].

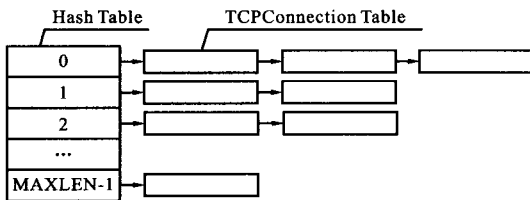


图 1 用哈希表管理多连接

在实际应用中,网络数据包所表现出的局部性可以优化上述哈希表结构.同属于一个 TCP 链接的数据包一般会在较短的时间间隔内陆续到达,据此,接收方每收到某个 TCP 链接的数据包,便把该 TCP 链接节点移值冲突链表的首部.这样,在系统运行一段时间以后,那些活动较为频繁的 TCP 链接节点将集中在冲突链表中靠前的位置,因而极大地缩短了链接的定位时间.使用局部性原理优化哈希表带来的另一个好处是:当发现系统中出现大量僵死链接节点时(僵死链接节点是指那些因 fin 报文或 rst 报文没有捕获到而无法得到释放的链接节点),能很快地从冲突链表的尾部找到这些僵死节点进行释放,保证了系统资源的有效利用.

4 单一连接的报文段重组算法

假设一个新到达的报文段经过以上算法处理,准确定位到某一个特定的 TCP 连接^[3],那么接下来

需要解决下面的问题,即对于某个特定的链接,对于以顺序或者乱序到达的报文段,如何以最为合理的方式重组,并恰当处理可能出现的报文段覆盖.

4.1 顺序到达的报文段的处理

有研究显示,只有 0.1%~2.0% 的报文段需要经过重组才能交付.对于按序到达的报文段,系统不应对其进行缓存,而是尽可能早地提交,以提高系统的整体性能.考虑这样一种情况:系统没有接收到序号为 m 的报文段,而接收到了序号为 n ($n > m$) 的报文段,系统在 n 之前产生了一个大小长度为 $n - m$ 的空洞,并把序号为 n 的报文段进行缓存.在后续到达的报文中,系统接收到序号为 m ,长度为 $n - m$ 的报文段,并将其提交,此时,系统必须检查缓存中的报文段,并将序号为 n 的报文一并提交.图 2 中的①和②分表示缓存链表为空的情况以及缓存链表中存在可以合并提交的报文段的情况.

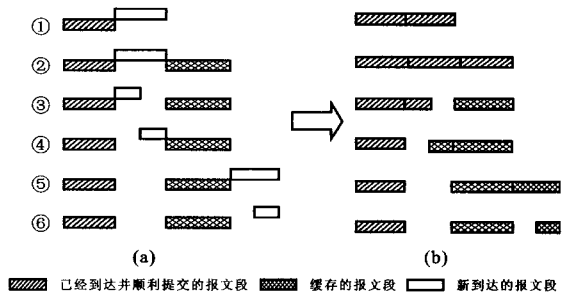


图 2 对新到达的报文段进行处理

4.2 非顺序到达的报文段的处理

TCP 协议规定^[4],接收方一旦接收到一个序号大于本机最近发出的确认号的报文段,需要对其进行缓存,并再一次向对方主机发出前一个报文段的请求.在本文开发的系统中,也提供相同的机制,以确保系统与被监听主机数据的一致性,并用链表结构组织缓存中的报文段.当接收到一个新的报文段,系统可能会出现图 2(a)所示多种情形中的任意一种,这时就需要根据报文段的序号,顺序遍历报文段链表,寻找合适的位置并进行插入操作.图 2(b)显示了经系统处理后情形.在这里需要指出的是顺序遍历链表的方向,当系统中出现多个不连续的数据空洞,采用反向遍历的方法具有更高的效率.根据 TCP 协议规范,TCP 发送端并不需要等待前一个报文的确认报文到达后才进行下一个报文的发送,而是根据 TCP 接收方声明的接收窗口大小,连续发送多个报文段,这就意味这一旦接收方的缓存节点链表中出现一个空洞,下一个到达的报文段并不一定能立刻填补这个空洞,而是在多个连续的报文段到

达之后,与这个空洞相对应的报文段才会到达.从链表的尾端往前搜索,就可以保证在多数情况下都不需要遍历整个链表即可找到合适的插入位置,这无疑能提高系统的整体性能.

在高流量的网络环境下,通过端口镜像捕获数据包很可能存在丢包现象,若不对此进行处理,而是将随后到达的所有报文段进行缓存,等待那个永远不会到来的报文段,必然会导致资源的迅速耗尽.对此的解决方案如下:若接缓存中出现空洞,却在接收方数据报文段中对空洞中的数据进行确认,便可以确定这个空洞是因为丢包产生,系统向上层报告这个丢失报文的序号和长度,并将后续的缓存节点中可以提交的部分全部提交后释放;对于长时间不到的报文,为防止开辟大量内存,也应放弃等待并继续处理接下来的报文段.

4.3 出现报文段覆盖的处理

由于 TCP 数据传输的复杂性^[5],在实际的传输过程中,报文段的重复与覆盖现象是无法避免的.图 3 例举了一个新的报文段到达时系统可能出现的情况.在各种 TCP 协议栈的实现中,对于覆盖的报文段的处理是不同的.以 Linux 为例,新到达的报文段总是会覆盖之前到达的报文段,而对于 solaris 来说,情况刚好相反.在我们的系统中,所有顺序到达的报文段将会被立即提交,对于图 3 所示的情况,我们别无选择地丢弃这个报文段,因为此报文段携带的所有数据都已经顺利提交并解析.对于其他可能出现的状况,我们用统一的办法进行处理:新的报文段中所有已经提交或缓存的部分直接丢弃,只留下与空洞对应的部分数据.

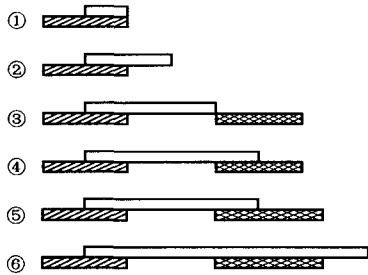


图 3 出现报文段覆盖

上述过程可以用以下伪码描述:

```
1. S = ReceiveNewSegment();
2. (Data, Info, Sender, Receiver) = ParseSegment(S);
3. if (Info.ack > Sender. BuffList. hole[x]. sequence)
4.   ReportSegmentLost ( Sender. hole [ x ]. sequence,
Sender. hole [ x ]. length);
5. if(Sender. BuffList. ReadyNodeNumber != 0)
```

```
6. Forward(Sender. BuffList. ReadyNode);
7. FreeBuffNode(Sender. BuffList. ReadyNode);
8. else
9.   V = DiscardOverlappingData (Data, Receiver);
10.  if (V is not empty)
11.    R = MakeNewBuffNodes(V);
12.    InsertIntoBuffList(R, Receiver);
13.  if(Receiver. BuffList. ReadyNodeNumber != 0)
14.    Forward(Receiver. BuffList. ReadyNode);
15.    FreeBuffNode(Receiver. BuffList. ReadyNode);
16.  else
17.    DropSegment(S).
```

5 实现与结果分析

在 Linux 系统中实现 该算法,采用 C 语言为编程工具,底层采用 Libpcap-mmap 数据报捕获模块进行数据包的捕获,IP 数据包重组部分使用了 Libnids 的代码.用本文所述算法与 Libnids API 开发了两套测试程序,在相同的网络环境下,分别对基于 TCP 的 FTP 通信协议进行截取分析.

一个 FTP 会话通常使用两个 TCP 连接^[6-7],21 端口用于传送控制信息,20 端口用于传送数据文件.通过对 21 端口进行监听,截取用户名、密码、文件名等信息,并对 20 端口传送的数据文件进行完整地保存.分析时选取国外某 FTP 站点作为数据源(选取距离较远的国外 FTP 站点是为了尽可能多地产生无序报文段),FTP 下载速度约为 360kB/s,使用两套测试程序分别截取并还原大小不同的数据文件,比较结果如图 4 所示.

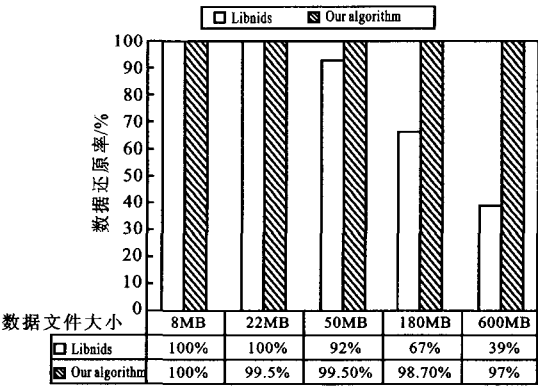


图 4 本文所述算法与 Libnids 数据还原能力的比较

由图 4 可见,在不出现丢包的情况下,本算法与 Libnids 均可保证 100% 的数据还原率,一旦出现丢包现象,Libnids 便无法还原丢包之后的所有报文,根据丢包的时间早晚不同,Libnids 便呈现不同的还原率;而在本算法中,根据 4.2 节所述,因为对丢包

以及报文长时间不到的情况进行了特殊处理,保证了 97% 以上的重组率。

6 结束语

本算法的低开销、高灵活度,为繁杂的应用层协议分析系统节省了宝贵的系统资源。但是,由于 Libpcap 在高速网络下的数据包捕获能力有限,导致 TCP 报文段的大量丢失,一定程度上影响了算法的效率与数据的还原率。因此,下一步将研究更为高效的抓包机制,提高系统的整体性能。目前,本算法实际运用于自主开发的网络行为管理系统中,运行稳定。

参考文献:

- [1] 韩东东,杨立国,赖杰贤. 校园网环境下的电子邮件监控与审计系统[J]. 中国教育信息化,2008(5):53-55.
- [2] 姜恩勇,岳猛. 一种可扩展的 webmail 监视系统的设计

与实现[J]. 计算机安全,2009(4):18-20.

- [3] 邓子宽,范明钰,王光卫,等. Snort 入侵检测系统中 TCP 流重组的研究[J]. 信息安全与通信保密,2007(2):65-67.
- [4] 熊兵,陈晓苏,陈宁. A real-time TCP stream reassembly mechanism in high-speed network[J]. 西南交通大学学报:英文版,2009,17(3):185-191.
- [5] Ethan Blanton, Mark Allman. On making TCP more robust to packet reordering[J]. ACM,2002,32(1):20-30.
- [6] Richard Stevens W. TCP/IP 详解 卷 1:协议 [M]. 北京:机械工业出版社,2000:174-269.
- [7] Charles M Kozierok. TCP/IP 指南,卷 1:底层核心协议 [M]. 北京:人民邮电出版社,2008:511-587.

作者简介:

赵启升 男,(1970-),高级工程师.研究方向为计算机网络.

李存华 男,(1963-),博士.研究方向为数据挖掘及图形图像处理.

(上接第 128 页)

模评价数据时性能堪忧.针对此情况,本文提出一种基于评价区间余弦相似性的协同过滤推荐算法.首先将用户评价的取值范围划分为 10 个定性评价区间,依据用户评价落入各区间的比率建模项目质量和用户评价取向,从而降低稀疏性和大规模数据对推荐系统造成的负面影响.实验结果表明,基于用户评价区间的协同过滤推荐算法可明显提高推荐系统的推荐质量。

参考文献:

- [1] 朱岩,林泽楠. 电子商务中的个性化推荐方法评述[J]. 中国软科学,2009(2):183-192.
- [2] 张海燕,丁峰,姜丽红. 基于模糊聚类的协同过滤推荐方法[J]. 计算机仿真,2005,22(8):144-147.
- [3] Sarwar B, Karypis G, Konstan J, et al. Item-Based collaborative filtering recommendation algorithms[C]// Proceedings of the 10th International World Wide Web Conference. Horgliorg, 2001: 285-295.
- [4] 邓爱林,朱扬勇,施伯乐. 基于项目评分预测的协同过

滤推荐算法[J]. 软件学报,2003,14(9):1621-1628.

- [5] Arwar B, Karypis G, Konstan J, et al. Analysis of recommendation algorithms for E-commerce[C]// Processing of 2nd ACM Conference on Electronic Commerce. New York, 2000:158-167.
- [6] Kim BM, Li Q, Park C S, et al. A new approach for combining content-based and collaborative filters[J]. Journal of Intelligent Information System, 2006,27(1):79-91.
- [7] Chedrawy Z, Abidi SSR. An adaptive personalized recommendation strategy featuring context sensitive content adaptation [C]//Proceedings of Adaptive Hypermedia and Adaptive Web-based Systems. Germany, 2006, 4018: 61-70.
- [8] 张锋,常会友. 使用 BP 神经网络缓解协同过滤推荐算法的稀疏性问题[J]. 计算机研究与发展,2006,43(4):667-672.

作者简介:

项武 男,(1968-),硕士,高级工程师.研究方向为自动控制、人工智能。