

网络数据包的协议分析算法设计与实现

王锦超, 李 飞, 沈明玉

(合肥工业大学 计算机与信息学院, 安徽 合肥 230009)

摘 要:为有效地监听网络状况和数据传输, 截获网络传输的数据包, 分析网络性能, 排除网络故障, 文中在以太网的基础上, 通过对网络数据包的协议分析, 设计并实现了一个网络数据包的协议分析算法(PLA算法)。PLA算法可以有效地对网络中传输的数据包进行协议分析, 解决了如何判别在网络上传输的数据包是什么类型的数据包, 每一个数据包都用到了哪些协议。通过 PLA 算法对数据包的分析, 可以使得流量统计和流量收费更加精确。

关键词:协议分析; 网络管理; PLA 算法; 网络安全

中图分类号: TP393.07

文献标识码: A

文章编号: 1005-3751(2006)04-0030-03

Design and Implementation of Protocols Analysis Algorithm for Network Datagram Packets

WANG Jin-chao, LI Fei, SHEN Ming-yu

(Computer & Information School of Hefei University of Technology, Hefei 230009, China)

Abstract: In order to monitor network status and data transfer, intercept and capture datagram packets on the network, analyze network capability, exclude network malfunction, system intrusion and detection, this paper designed and implemented the protocol analyze (PLA) algorithm based on the Ethernet. PLA can effectively process protocol analysis for datagram packets. It's a solution to distinguish the types of the datagram packets and which protocols the datagram packets use. Through PLA, flux analysis can be even more exact.

Key words: protocols analysis; network management; PLA algorithm; network security

0 引言

随着网络的普及, 其规模越来越大, 应用的范围也越来越广。运行在网络中的设备和软件都遵守着一定的规则, 这些规则被称为协议^[1]。网络上传输的数据要按照一定的协议格式进行打包封装成数据包后才能在网络中正确地传输。由于网络中存在多种协议, 故需要分析网络中传输的数据包使用何种协议, 即对数据包进行协议分析。

在网络管理和网络安全方面, 协议分析是非常重要的手段之一。利用协议分析技术, 对网络数据包按协议分类统计, 可以提高流量统计的精确度, 使得网络按流量计费更合理, 进一步分析网络的性能及其状况、排除网络故障等^[2]。网络管理和网络安全中, 常常需要对数据包进行协议过滤和内容过滤, 同样离不开对数据包进行协议分析。

文中设计了一个网络数据包的协议分析算法(以下简称 PLA 算法), 将网络上捕获的数据包按照协议类型由低到高层分析, 并可读出数据包的内容, 以便对数据包进行相应的处理。

1 协议分析基础

对数据包进行协议分析, 先要捕获数据包。在以太网中, 要捕获所有流经网卡的数据包, 先将网卡设置为混杂模式^[3], 这样就能监听所有流经网卡的数据包。由于只对数据包进行协议分析, 所以捕获数据包的工作采用其它工具来实现。为简单起见, 在算法的程序实现中采用了 Windows 2000 DDK 中的 Packet 驱动来捕获数据包。另外, 网络接口卡的驱动程序会负责计算校验和, 并取走帧中的同步码与校验和字段, 因此接收的数据包仅是帧头和载荷部分^[4]。

对数据包进行协议分析前, 必须了解以太网的帧的层次结构, 了解各种协议间的层次关系和协议封装格式, 以及协议封装格式中每个字段的含义。由于篇幅的关系, 这里只给出以太帧的层次结构(见图 1)和常见的以太帧格式^[5,6](见图 2)。详细的协议格式、帧格式及各字段含义可以查阅其它参考资料^[1,7]。

2 PLA 算法及实现

协议分析的算法 PLA 算法的步骤如下^[8]:

(1) 启动一个写缓冲区线程, 将捕获到的数据包存储到一个足够大的缓冲区 CaptureBuffer 中。若缓冲区 CaptureBuffer 已满, 则丢弃该包, 等待接收下一个数据包; 若

收稿日期: 2005-07-20

作者简介: 王锦超(1982-), 男, 安徽肥东人, 硕士研究生, 研究方向为网络管理及信息安全技术、主动网络管理技术和安全技术; 沈明玉, 博士, 副教授, 主要研究领域为计算机网络技术、主动网络和信息安全技术等。

缓冲区 CaptureBuffer 未 满,则 写 入 CaptureBuffer 尾 部,尾 指 针 后 移 一 位。

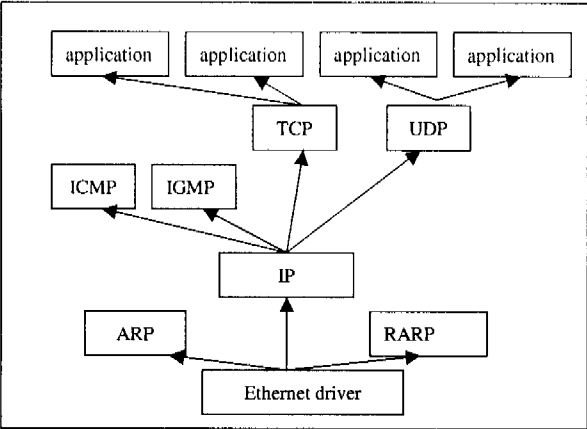


图 1 以太网层次结构

6	6	2	1	1	1-2	38-1492	4
目的地址	源地址	数据长度	DSAP	SSAP	Control	数据	检查和

(a). IEEE802.2 帧

6	6	2	1	1	1	3	2	38-1492	4
目的地址	源地址	数据长度	DSAP (AA)	SSAP (AA)	Control (03)	OUI	ID	数据	检查和

(b). IEEE802.2 SNAP 帧

6	6	2	0-1500	0-46	4
目的地址	源地址	类型	数据	填充字段	检查和

(c). 以太网

图 2 以太网格式图

(2)启动一个读分析缓冲区线程,若缓冲区 Capture-Buffer 为 空,则 等 待;若 不 为 空,则 从 缓 冲 区 CaptureBuffer 头 部 读 取 一 个 数 据 包,存 入 变 量 CurrentDataFrame 中,并 将 该 数 据 包 从 该 缓 冲 区 中 删 除,头 指 针 后 移 一 位。

(3)将取得的数据包 CurrentDataFrame 的前 12 字节 (0~11 字节)取出,其中前 6 字节表示数据包发送的目的 MAC 地址,存入变量 TempDestinationMAC 中;后 6 字节表示数据包发送的源 MAC 地址,存入变量 Temp-SourceMAC 中。

(4)将 CurrentDataFrame 数据包的 12~13 字节取出,存入变量 TypeOrDataLength 变量中。比较 TypeOrDataLength 的值,若 0x05DC < TypeOrDataLength < 0x0600,则 转 (11);若 TypeOrDataLength ≥ 0x0600,则 转 (9);若 TypeOrDataLength ≤ 0x05DC,转 (5)。

(5)检查 CurrentDataFrame 数据包的 LLC(逻辑链路控制)字段,从第 14 字节开始,每字段所占字节数如图 3 所示,此字段格式如下:

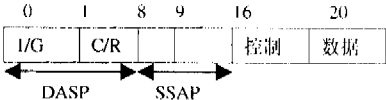


图 3 局域网 LLC 帧格式

比较 DSAP(Destination Service Access Point,目标服务访问点)和 SSAP (Source Service Access Point,源服务访问点)两字段,若 DSAP ≠ SSAP,则 转 (12);若 DSAP = SSAP = 0xF0,则 转 (13);若 DSAP = SSAP = 0xFF,则 转 (14);否则 转 (6)。

(6)检查 LLC 的控制字段,若 LLC 控制字段不等于 0x03,则 转 (15);否则 转 (7)。

(7)检查 LLC 的 DSAP 和 SSAP 字段,若 DSAP = SSAP ≠ 0xAA,转 (16);否则 转 (8)。

(8)检查紧跟着控制字段的后面的 3 个字节,此 3 字节即为 OUI 字段。若 OUI 字段的值等于 0x00-00-F8,则 转 (17);若等于 0x08-00-07,则 转 (18);若等于 0x00-00-00,则 转 (9);否则 转 (19)。

(9)检查接下来的两字节,即类型字段,按表 1 进行协议分析,若类型字段为 0x0800;则 转 (10),否则按表 1 所得的协议类型进行协议格式分析,转 (20)。

(10)对后面的字段用 IP 协议格式进行分析^[9],检查数据包中 IP 协议格式中的“协议”字段,按表 2 进行协议分析。转 (20)。

(11)此数据包为错误的数据报,清空 CurrentDataFrame(即丢弃该数据包),做相应的记录并转 (20)。

(12)下层协议类型为未知类型,做相应记录,转 (20)。

(13)下层协议为 NETBIOS;按 NET-BIOS 协议的封装格式对数据包进行分析,转 (20)。

表 1 协议-值对照表

值	协议
0x0800	IP
0x0806	IP-ARP
0x8035	IP-RARP
0x8137	IPX
0x809B	Apple Talk
0x80F3	Apple Talk-ARP
0x6001,0x6002	DEC MOP
...	...

表 2 协议号-协议对照表

1	ICMP	11	NVP-II	21	PRM
2	IGMP	12	PUP	22	XNS-IDP
3	GGP	13	ARGUS	23	TRUNK-1
4	IP	14	EMCON	24	TRUNK-2
5	ST	15	XNET
6	TCP	16	CHACS	95	MICP
7	UCL	17	UDP	96	SCC-SP
8	EGP	18	MUX	97	ETHERIP
9	IGP	19	DCN-MEAS	98	ENCAP
10	BBN-RCC-MON	20	HMP	100	GMTP

(14)下层协议为 IPX(802.3RAW);按 IPX(802.3RAW)协议的封装格式对数据包进行分析,转 (20)。

(15) 下层协议类型未知;做相应记录并转(20)。

(16) 根据 DSAP 和 SSAP 的值,按表 3 进行协议分析。转(20)。

表 3 常见 DSAP-SSAP 值

DSAP & SSAP	protocol	DSAP & SSAP	protocol	DSAP & SSAP	protocol
00	NULL	42	BPDU	E0	IPX
02	LLC	4E	RS-511	EC	CLNP
03	LLC	5E		F0	NETRUEI
04	SNA	7E	ISO208	F4	LM
05	SNA	80	XNS	F5	LM
06	IP	86	NESTAR	F8	
08	SNA	8E	IEC955	FA	
0C	SNA	AA	VSNAP	FC	RPL
0E	IEC955	BC	VIP	FE	
20	CLNP	DC		FF	LLC
34	CLNP	D4			

(17) 下层协议为 802.1H Bridge Tunneling,按 802.1H Bridge Tunneling 协议的封装格式对数据包进行分析,转(20)。

(18) 检查下两字节是否为 0x809B,若是,则为 Apple Talk;否则为未知类型。转(20)。

(19) 未知协议类型,做相应记录,转(20)。

(20) CurrentDataFrame 数据包协议分析结束,若 CurrentDataFrame 非空,将 CurrentDataFrame 数据包和分析结果存入已分析的数据包缓冲区,以便做其它处理;从读分析线程中读取 CaptureBuffer 缓冲区,若为空,则等待;否则从 CaptureBuffer 尾部读取一个数据包到 CurrentDataFrame 中,并从缓冲区中删除该数据包,尾指针后移一位。转(3)。

3 总结

通过实验表明,写线程的工作量比较小,即使网络负荷较重的情况下,也不会发生来不及接收数据包而产生丢包现象。而读分析线程的工作量要远远比写线程的工作量大。

对于网络流量较小的状况下,开启一个写线程和一个读分析线程,基本上不会出现接收缓冲区满而产生丢包现象。当网络流量逐渐增大时,一段时间后,就会出现读分析线程来不及读取分析造成缓冲区溢出,进而出现丢包现象。为了解决这个问题,通过开启多个读分析线程读取缓冲区 CaptureBuffer,同步进行协议分析。由于网络流量的不确定性,为了使开启的读线程数更加合理性,采用动态增减读分析线程数来合理分配线程资源。

这样,可以根据缓冲区的利用情况来决定读分析线程数,合理分配线程资源,既不会发生丢包现象,也不会造成

资源浪费。

图 4 是在算法实验过程中截取的数据包协议分析示意图。

```

- IEEE 802.3 Frame, 从 0x00-04-61-52-06-49 到 0xFF-FF-FF-FF-FF-FF, IPX
+ 目标MAC地址:0xFF-FF-FF-FF-FF-FF
+ 源MAC地址:0x00-04-61-52-06-49
- IEEE802.3 LLC, DSAP = 0xE0, SSAP = 0xE0, Control = 0x03
  DSAP:0xE0(224)
  SSAP:0xE0(224)
  Control:0x03(3)
- IEEE 802.3, IPX
  帧类型: Unnumbered Frame
  command: UI
- IPX 帧结构
  IPX: Checksum=0xFFFF
  IPX: IDP Length=0x0060
  IPX: Transport control=0x00
  IPX: Packet type=IPX
+ IPX: Destination Address Summary=0x00000000FFFFFFFFFFFF0452
+ IPX: Source Address Summary=0x0000000000004615206494000
IPX, 0x5C(92) Bytes
  
```

图 4 HTTP 协议分析示例

本算法已经在操作系统为 Windows 2000 下用 Visual C++ 6.0 开发工具实现^[10]。在本算法的实现中,只分析数据包中用到哪些协议,并按照相应的协议格式读出数据包中个字段的内容,没有做其它相关处理,比如做碎片分析、按协议进行流量统计、协议过滤等,不过这些都可以在本算法的基础上进行扩展来实现。

参考文献:

- [1] Stevens W R. TCP/IP 详解(卷 1:协议)[M]. 北京:机械工业出版社,2002.
- [2] 周明天,汪文勇. TCP/IP 网络原理与技术[M]. 北京:清华大学出版社,1993.
- [3] 谭思亮. 监听与隐藏:网络侦听解密与数据保护技术[M]. 北京:人民邮电出版社,2002.
- [4] 高传善,钱松荣,毛迪林. 数据通信与计算机网络[M]. 北京:高等教育出版社,2001.
- [5] 索红光,石乐义,梁玉环. TCP/IP 协议分析器的设计开发[J]. 计算机工程与应用,1999(11):80-83.
- [6] 李德鹏,史清华. 使用协议分析预警、排除 NOVELL 网络故障[J]. 计算机应用研究,1999(3):52-54.
- [7] 谢朝曦,熊其邦. 以太协议分析仪的实现[J]. 福建电脑,1994(2):18-24.
- [8] McCanne S, Jacobson V. The BSD Packet Filter: A New Architecture for User-level Packet Capture[M]. CA: [s. n], 1992.
- [9] 谢 鲲,张大方. 共享网段网络协议分析系统设计与实现[J]. 计算机工程与科学,2002,24(2):25-28.
- [10] 马恒太,蒋建春,刘克龙,等. 一个基于 Unix 平台下的分布式网络监听器系统[J]. 计算机研究与发展,2000(3):268-274.

(上接第 29 页)

部运动[J]. 计算机应用,2004,12(3):338-342.

[4] 于金辉,李一兵. 计算机动画原理与制作技术[M]. 北京:清

华大学出版社,1995.

[5] 刘 真. 实用计算机图形与动画技术[M]. 北京:电子工业出版社,1998.