**“网络工程专业”2019年专业实践指导书**

**（网络工程17级）**

**一、总体要求**

**1. 分组名单**

|  |  |  |  |
| --- | --- | --- | --- |
| 组别 | 组长 | 成员 | 备注 |
| 第一组 | 吕燕洁 | 唐帅、王路、韦岩方、刘一卓 |  |
| 第二组 | 杜文基 | 彭树洋、陈子弘、盛晟、刘杨 |  |
| 第三组 | 陈婧缓 | 方一淼、徐兆宏、束欣然、王鹏飞 |  |
| 第四组 | 付辰曦 | 赵昕阳、陈香岚、张泽宇、周永 |  |
| 第五组 | 陈裕哲 | 甘伟将、刘煜、李成瀚、雷靓男 |  |
| 第六组 | 刘玥 | 岳雨琛、陈瑶瑶、刘硕华、张浩（网工16） |  |

组长责任：组长负责保证整个实践项目的完成，负责项目答辩（需要准备PPT）、并将整个项目拆解

**2. 任务与分工**

为以下任务分配给组员完成：

1. 开发系统—1人负责

虚拟机系统Slackware Linux；web服务器lighttpd； PHP解析进程php-fpm; 数据库mysql，可以使用Java Web；并实现无线网卡处于Monitor模式下监听2.4G WLAN13个信道的WLAN数据帧

2) 守护进程设计与实现 --2人负责

成员A 负责编写基于libpcap的守护进程捕获WLAN帧，并提取HTTP协议的有效载荷

成员B 负责设计数据库，并编写写库程序，将HTTP协议的有效载荷拆解后保存在数据库中

要求：捕获的数据包量为30000个封装有HTTP协议的WLAN数据帧

数据库设计要求符合第三范式的要求

保存的信息有：捕获的数据包顺序号、源IP地址、目的IP地址、HTTP协议的get头

User-Agent头、host头、Full request URI头

成员A与成员B的接口为.h头文件

3) WEB程序设计与实现—1人负责

设计与实现一简单的web系统显示数据

该系统的组成模块：认证、源IP地址统计（Echart图）、目的IP地址统计

每个Host头所拥有的公司（域名所属的公司）

要求：每个页面未经认证的用户不可访问（采用session实现）

认证页面要求具有校验码

4) 实现whois程序---1人负责

根据whois类，编写程序，实现每个host头对应的域名所注册的单位及联系电话，并保存于数据库中（可以使用任何脚本）

**3. 成绩**

个人成绩=系统实现（20%）+实践报告（25%）+个人代码质量（30%）+答辩（25%）

由组长负责答辩，但教授会对每个人的代码实现方法进行问询

**4. 最终完成时间**

2019年12月20日下午2:00前

**5. 资料提交**

在答辩完成后，2019年12月27日前提交下列材料

1. 个人的“网络工程中期检查实践报告”
2. 光盘

整个系统需要刻盘，并由组长交至西配楼110办公室

光盘包含4个目录：

（1）数据库目录：Mysqldump文件数据库的所有表的mysqldump原始数据，并附describe <表名>所列的所有表的表结构（word文档）

（2）守护进程目录：C语言程序的源代码，并附Makefile文件

（3）web系统目录：web系统的所有文件的源代码

（4）实践报告目录：组长及组员的实践报告电子文档

**6. 关于系统的软件著作权**

每个小组将系统完成后，可以各自申请软件著作权；

但需要明确，数据包捕获部分是由北京林业大学“网络教研室”根据libpcap函数库编写，这一部分内容不应包涵在软件著作权范围内，**若软件著作权中确需这一部分内容，可根据教师提供的教学程序重新自组织编写，忽直接引用教学程序。**

**二、项目实践内容**

**1、系统分析**

系统名称：基于libpcap的WLAN帧HTTP载荷分析系统的设计与实现

采用数据流图完成整个系统的分析，重点分析得出以下内容：

1. 系统的组成模块
2. 每个模块的工作流程
3. 每个模块数据的存贮方式

要求：1）数据流图至少画到第三层（即第0层数据流图、第一层数据流图、第二层数据流图、第三层数据流图）

2）由组长招集全组分析

注：数据流图的分析方法见附件1

**2. 系统设计**

实现以下目标：

1. 系统组成模块设计
2. 数据库设计
3. 各个模块的设计（需要详细到实现函数或类）

注：web系统如果不掌握MVC，可以使用源生代码编写

Web系统如果不掌握前后端分离技术，但至少要使用bootstrap.js

若希望使用前后端分离技术，则前端建议使用：vue.js, angular.js, react.js，其它不建议使用

**3. 系统实现**

分模块编写的实现文档，需要详细到每个函数的实现流程

**4. 系统测试**

**4.1 功能测试（所谓的黑盒测试）**

**4.2 性能测试**

测试web系统的性能，见网络服务实验指导书

**三、数据包捕获环境**

**1. 采集地点：学研大厦**

**2. 数据采集Topology**



**图1 数据采集示意**

说明：

1）无线网卡处于monitor模式下

2）虚拟机安装有libpcap函数库（slackware带有，若需要新版本可从互联网下载：<https://www.tcpdump.org/#latest-releases>），其安装方法见网络服务课程实验指导书

3）WLAN帧结构及将网络设置于Monitor模式的方法见附件2

**四、实验报告内容**

**题目：**基于libpcap的WLAN帧HTTP载荷分析系统的设计与实现

中文摘要

1. **绪言**

1.1 研究的目的意义

1.2 数据包捕获的国内外研究现状

1.2.1 国内研究现状

1.2.2 国外研究现状

**2．系统分析**

2.1 顶层数据流图

2.2 一层数据流图

2.2 二层数据流图

2.3 三层数据流图

**3. 系统设计**

* 1. 系统组成模块设计
  2. 数据库设计

3.3 各个模块的设计（需要详细到实现函数或类）

**4. 系统实现**

4.1 sniffer模块实现

4.2 ……

**4. 系统测试**

4.1 功能测试（所谓的黑盒测试）

4.2 性能测试

**5. 结论与讨论**

5.1 研究结论

5.2 讨论

**附件1**基于SNMP协议的网络监测系统的设计与实现

**1. 主要任务**

1）编写基于SNMP协议的数据采集守护进程，并将采集到的数据写入memcache与数据库中

2）编写web程序，采用Ajax技术，每隔固定的间隔将保存于memcache中的数据反映于web界面中

3）编写web程序实现保存于数据库中数据的统计

**2. 系统分析**

**2.1 第0层OFD图**



图1 顶层OFD图

**2.2第1层OFD图**



2.3 P2 设备登记模块

作用：将所有监测网络设备按照其所属区块登录IP地址、用户名、密码、enable密码、设备位置、共同体字符串、

主机名信息，

说明：1）所属区块分为：核心区块、WAN区块、服务器区块（仅考虑该区块内的网络设备）、DMZ区块（仅关注该区块内的网络设备）、交换区块（可设置为多个，关注该区块内的汇接层交换机与接入层交换机）

2) 登记的内容写入配置文件中，配置文件在/etc/monitor.cfg文件

P2 工作流程：



2.4 p3 监测数据模块

任务：1）编写asyncapp程序，从设备表中读取拓扑结构中每个设备的IP地址及公同体字符串，采用轮询方式获取数据：每个设备选择1个或2个互联端口获取该端口的ifInPackets, ifOutPackets, ifInError, ifInOctets, ifOutOctets数据，保存于memcached与mysql数据库中。

要求：(1) 轮询间隔为60秒

(2) 程序采用单实例守护进程模式编写

(3) 需求轮询的端口写入/etc/monitor.cfg配置文件中

(4) 可以采用net-snmp函数库中的程序snmpwalk遍历设备的对象，从而获得对象名称或OID

2）监测数据的显示。将保存于memcached中的数据，以列表的方式，采用ajax技术显示于web界面中

要求：(1)列表的CSS要清晰，可读性强

(2)ajax的刷新间隔期为65秒

2.5 P4 数据统计模块

作用：按学期统计：统计各个交换区块入/出流量、入/出数据包量；统计WAN区块出口入/出流量，入/出数据包量；

DMZ区块入/出流量、入/出数据包量；对内服务区块入/出流量、入/出数据包量

说明：1）要求具有打印功能

2）每个交换区块一页，B5幅面，整个页面分为上、下两部分

上半部：统计表

下半部：echart图

统计表样式：交换区块A统计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 月份 | 入流量 | 出流量 | 入数据包量 | 出数据包量 |
| 2或8 |  |  |  |  |
| 3或9 |  |  |  |  |
| 4或10 |  |  |  |  |
| 5或11 |  |  |  |  |
| 6或12 |  |  |  |  |
| 7或1 |  |  |  |  |

附件2 WLAN帧结构及wlan0无线网卡Monitor设置方法---802.11帧

一、变更wlan0的工作模式

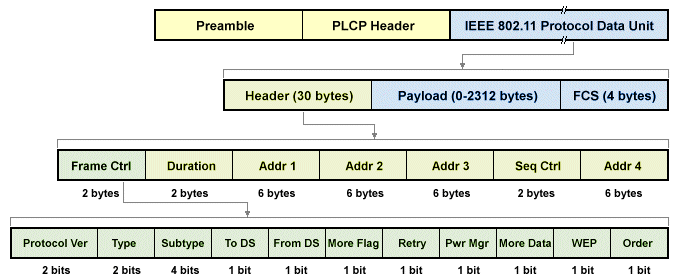
#ifconfig wlan0 down //成功关闭

#iwconfig wlan0 mode monitor //设置monitor模式

#ifconfig wlan0 up

二、IEEE 802.11 MAC帧格式





注：addr4(可选)：仅在分布式WLAN中出现

（一）、Frame Control

所有帧的开头均是长度为2个字节的Frame Control



|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 长度 | 默认值 | 作用 |
| Protocol | 2 bit | 0 | 协议编号 |
| Type与Sub type | 6 bit |  | 用来指定所使用的帧类型，   1. Type=00, subtype值与名称   0000连接请求 0001连接应答 0010重新连接请求 0011重新连接应答  0100Probe request 0101Probe response 1000beacon导引信号  1001数据代传指标通知信号 1010解除连接  1011身份验证 1100解除认证   1. Type=01,subtype的值与名称   1010省电模式-轮询 1011RTS请求发送 1100CTS允许发送 1101ACK应答  1110CF-End免竞急期间结束 1111CF-End+CF-Ack免竞争期间结束，免竞争期间回应   1. Type=10, subtype的值与名称   0000Data数据 0001DATA+CF-Ack 0010Data\_CF-poll 0100Null data无数据  0101CF-Ack未发送数据 0110 CF-Poll未发送数据 0111Data+CF-Ack+CF-poll  1000QoS Data(由802.11e任务小组提议，但尚未标准化)，这些帧均以1开头  1001Qos data+CF-Ack 1010 Qos data+CF-poll 1011 Qos data+CF-Ack+CF-POLL  1100 QOS Null未发送数据 1101QOS CF-ACK未发送数据  1110 Qos CF-POLL未发送数据 1111 QOS CF-ACK+CF-Poll |
| TO DS | 1 |  | 用来指标帧的目的地是否为传输系统，它为1且From DS也为1时，addr4有效 |
| From DS | 1 |  | 用来指标帧的目的地是否为传输系统，它为1且To DS也为1时，addr4有效 |
| Protected Frame | 1 |  | 如果帧受到链路层安全协议的保护，则此帧为1，此bit也称为wep bit |
| Order bit | 1 |  | 严格依序传送，则此位被设定为1 |

(二)、Duration/ID位

有三种可能的形式



1. NAV 表示目前所进行的传输预计使用介质多少微秒
2. 免竞争期所传送的帧

Contention-free period，简称CFP，它让没有收到beacon（信标）帧的任何工作站，得以公告免竞争期间

1. PS-Poll帧

从休眠状态醒来的工作站必须送出一个Ps-poll帧，以便从基站取得之前暂存的任何帧

（三）、Address位

802.11帧最多可以包含4个地址位



Address1(RA:BSSID) 代表接收端(48bit) (所有包都包含它)

Address2(SA/TA)代表传送端(48bit)(除ACK和CTS包外的其它包都包含它)

Address3(DA) 被接收端用来过滤地址(48bit)，只用于管理包和数据包

**Address4(optional) 无线分布系统模式下，From DS和TO DS都被置位时使用，否则不存在**

(四)、Sequence Control顺序控制

长度为16位，用来重组帧以及丢弃重复帧



（五）、帧主体（Frame Body）



负责在工作站间传送上层数据。

在IP网络中，Path MTU Discovery将可避免大于1500byte的帧传递

（六）、帧校验和（FCS）

与以太网一致，是CRC

二、802.11对上层协议的封装

和所有其它的802链路层一样，802.11可以传输各种不同的网络层协议。

和以太网不同的是，802.11是以802.2的逻辑链路控制协议封装来携带上层协议，



802.11里的IP封装

例：Logical-Link Control

DSAP: SNAP(0xaa)

SSAP: SNAP(0xaa)

Control field: U, func=UI(0x03)

Organization Code: Encapsulated Ethernet(0x000000)

Type: IPv4(0x0800)

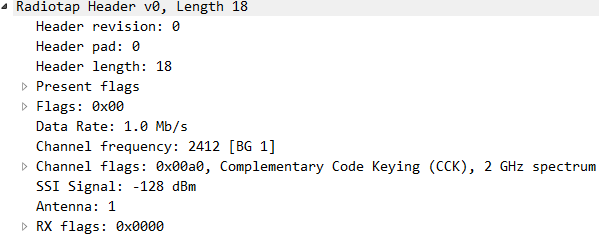
(一)、Monitor模式802.11帧封装

在Monitor模式下，802.11帧封装的顺序是：radiotap:wlan\_radio:wlan:11c:ip:tcp:http

在Monitor模式时，Linux内核在MAC帧前面提供额外的信息，称为Radiotap Header，其中包含的信息有MAC时间戳，BSSID信息强度、频道频率等 。

例：完整的802.11帧封装

1. **Radiotap Header----共计18字节长（radiotap附加信息）**





C定义如下：

//radiotap头部

//radiotap <http://www.radiotap.org/>

The radiotap capture format starts with a radiotap header:

struct ieee80211\_radiotap\_header {

u\_int8\_t it\_version; /\* set to 0 \*/

u\_int8\_t it\_pad;

u\_int16\_t it\_len; /\* entire length \*/

u\_int32\_t it\_present; /\* fields present \*/

} \_\_attribute\_\_((\_\_packed\_\_));

it\_version：表示版本号，当前为0。

it\_pad：没有使用，仅仅是为了结构体对齐。

it\_len：表示长度，包括了radiotap头部和数据两部分，如果不需要了解radiotap，则可以直接跳到

ieee802.11头部。

it\_present：表示radiotap数据的位掩码。radiotap的数据紧跟其头部。当其中的位掩码为true时，表示

有对应的数据，可以认为每一比特表示一种类型。比如bit5为1表示有通道数据，则可以获

取到信号强度，反之就是没有对应的数据。因此，radiotap的长度其实是不固定的。bit31

为1表示还有多个it\_present。

1. **Radiotap的附加信息**



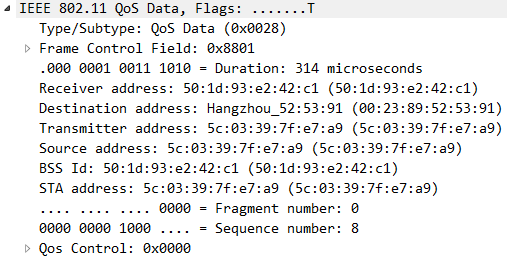
目前官方认可的附加信息字段参见如下：

* MPDU status
* Antenna noise //Antenna天线
* Antenna signal
* Antenna
* Channel
* FHSS
* Flags
* Lock quality
* MCS
* RX flags
* Radiotap Namespace
* Rate
* TSFT
* TX attenuation
* VHT
* Vendor Namespace
* dB TX attenuation
* dB antenna noise
* dB antenna signal
* dBm TX power
* timestamp

目前还在讨论中的字段类型定义如下：

* HE-MU-other-user
* HE-MU
* HE
* RSSI
* RTS retries
* TX flags
* XChannel
* data retries
* extended flags
* hardware queue

1. **IEEE802.11头-----30字节长**



// 802.11帧头

struct wlan\_frame {

uint16\_t fc;

uint16\_t duration;

uint8\_t addr1[6];

uint8\_t addr2[6];

uint8\_t addr3[6];

uint16\_t seq;

union {

uint16\_t qos;

uint8\_t addr4[6];

struct {

uint16\_t qos;

uint32\_t ht;

} \_\_attribute\_\_ ((packed)) ht;

struct {

uint8\_t addr4[6];

uint16\_t qos;

uint32\_t ht;

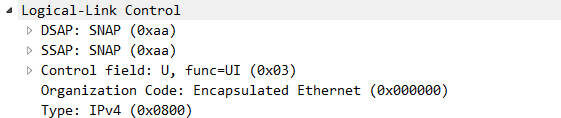
} \_\_attribute\_\_ ((packed)) addr4\_qos\_ht;

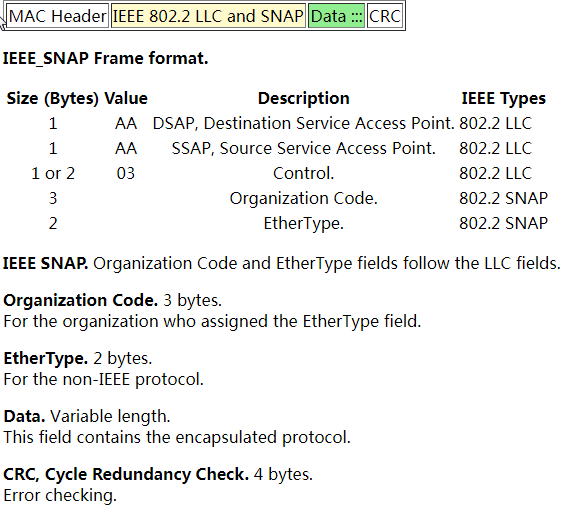
} u;

} \_\_attribute\_\_ ((packed));

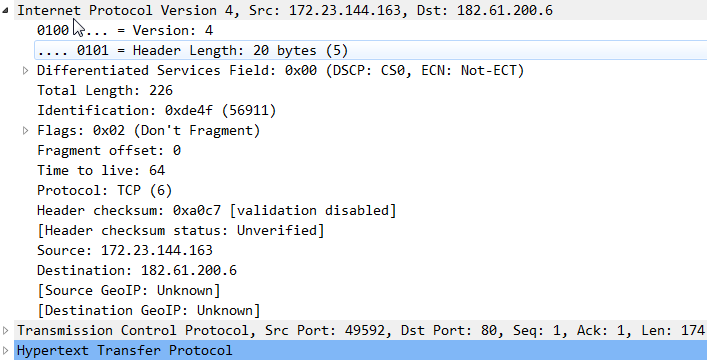
addr1为接收方(Receiver)MAC地址，addr2为发送者(Transmitter)MAC地址，addr3为BSSID。不同类型的帧，Receiver和Transmitter不同，对于probe request类型帧来说，Transmitter地址即为所需要的MAC号——因为probe都是广播，目标地址都是ff。

1. **LLC –---8字节长**





1. **载荷**



三、复杂性

当在Monitor mode下捕获IEEE802.11帧时，许多无线网卡驱动程序用上层提供“custom headers detailing a number of pieces of information about the captured packet, such as signal strength and noise values”。但服从于2个最常用的标准：the prism header and the AVS header

1. The Prism monitor

The Prism monitor mode header最初作为linux-wlan-ng项目（http://www.linux-wlan.com）的一部分，被用来开发Prism II 802.11b card for linux的驱动程序，现在被大多数网卡驱动程序及libpcap所支持

**Libpcap link type: DLT\_PRISM\_HEADER**

**The prism monitor mode header格式**

1. struct prism\_value

(

Uint32 did; //4字节

Uint16 status; //2字节

Uint16 len; //2字节

Uint32 data; //4字节

}; //共12字节

2) struct prism\_header

{

Uint32 msgcode; 4字节

Uint32 msglen; 4字节

U8char devname[16]; 16字节

Struct prism\_value hosttime; 12字节

Struct prism\_value mactime; 12字节

Struct prism\_value channel; 12字节

Struct prism\_value rssi; 12字节

Struct prism\_value sq; 12字节

Struct prism\_value signal; 12字节

Struct prism\_value noise; 12字节

Struct prism\_value rate; 12字节

Struct prism\_value istx; 12字节

Struct prism\_value frmlen; 12字节

}; 共计144字节

1. AVS

AVS捕获头是新被设计用来代替Prim monitor mode header. 它提供了额外的信息，AVS捕获头可以捕获关于802.11a和802.11g packet-capture sources.

Libpcap link type: DLT\_IEEE802\_11\_RADIO\_AVS

AVS capture header是64字节长

struct AVS\_header

{

uint32 version;

uint32 length;

uint64 mactime;

uint64 hosttime;

uint32 phytype;

uint32 channel;

uint32 datarate;

uint32 antenna;

uint32 priority;

uint32 ssi\_type;

int32 ssi\_signal;

int32 ssi\_noise;

uint32 preamble;

uint32 encoding;

};

新的可以进入monitor mode无线网卡驱动程序要么支持Prism monitor mode header, 要么支持AVS capture header或者均支持。