



(12) 发明专利申请

(10) 申请公布号 CN 103401877 A

(43) 申请公布日 2013. 11. 20

(21) 申请号 201310346769. 4

(22) 申请日 2013. 08. 09

(71) 申请人 上海斐讯数据通信技术有限公司
地址 201616 上海市松江区广富林路 4855 弄 20 号、90 号

(72) 发明人 付保林

(74) 专利代理机构 上海光华专利事务所 31219
代理人 余明伟

(51) Int. Cl.
H04L 29/06 (2006. 01)

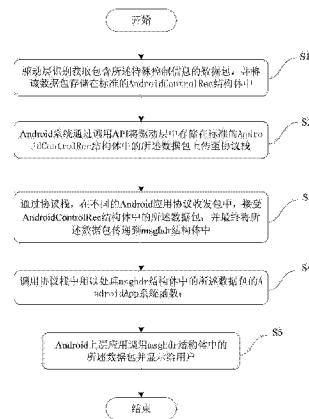
权利要求书2页 说明书8页 附图3页

(54) 发明名称

获取驱动层数据包控制信息的方法及系统

(57) 摘要

本发明提供一种获取驱动层数据包控制信息的方法及系统,所述方法是把包含特殊控制信息的数据包带入 AndroidControlRec 这个结构体中,因为整个协议栈都可以处理 AndroidControlRec 结构体中所带的数据包信息,而在上层传输过程中,可以使用 MsgHdr 这个结构体,不同协议收包都有确认使用该 MsgHdr 结构体存储一些自带的控制信息,最后通系统调用函数把 MsgHdr 中的数据包传输到上层应用。本发明在达到分析 Android 设备的上网性能时也可以像分析电脑的各项性能指标一样简单明了,数据实时有效,而且能彻底解放驱动对于数据包的内容检查判断。



1. 一种获取驱动层数据包控制信息的方法,应用于 Android 系统中,Android 系统接受和发送数据包中源 MAC 地址和协议号之间增加有固定的一段特殊控制信息,其特征在于,所述方法包括:

步骤 S1,驱动层识别获取包含所述特殊控制信息的数据包,并将该数据包存储在标准的 AndroidControlRec 结构体中;

步骤 S2,Android 系统通过调用 API 将驱动层中存储在标准的 AndroidControlRec 结构体中的所述数据包上传至协议栈;

步骤 S3,通过协议栈,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 Msghdr 结构体中;

步骤 S4,调用协议栈中用以处理 Msghdr 结构体中的所述数据包的 AndroidApp 系统函数;

步骤 S5,Android 上层应用调用 Msghdr 结构体中的所述数据包并显示给用户。

2. 根据权利要求 1 所述的获取驱动层数据包控制信息的方法,其特征在于,在步骤 S1 中,所述特殊控制信息为 10 个字节。

3. 根据权利要求 1 或 2 所述的获取驱动层数据包控制信息的方法,其特征在于,在步骤 S1 中,所述特殊控制信息包含识别标签,队列号,物理端口号,Vlan 标签,Vlan 号,RAT 标签,RAT 信息。

4. 根据权利要求 3 所述的获取驱动层数据包控制信息的方法,其特征在于,在步骤 S1 中,若 Android 系统采用 GPRS 通信,所述特殊控制信息还包含 modem 相关接入层特殊控制信息。

5. 根据权利要求 1 所述的获取驱动层数据包控制信息的方法,其特征在于,在步骤 S2 中,协议栈包括 WAP 协议,ARP 协议,IP 协议以及 TCP 协议。

6. 根据权利要求 5 所述的获取驱动层数据包控制信息的方法,其特征在于,所述数据包上传至协议栈,所述协议栈首先处理是否为 ARP 协议,若是,先对 ARP 协议做地址解析,然后依次通过 IP 协议,TCP 协议将所述数据包上传至 Android 上层应用层;若否,直接通过 IP 协议,TCP 协议将所述数据包上传至 Android 上层应用层。

7. 根据权利要求 1 所述的获取驱动层数据包控制信息的方法,其特征在于,在协议栈中至少包含驱动上调函数 DriverKernel_rx,IP 层收包处理函数 ip_recv,应用层收包处理函数 AndroidApp_recvmsg。

8. 根据权利要求 1 或 7 所述的获取驱动层数据包控制信息的方法,其特征在于,AndroidApp 系统函数支持 Android 系统现有的网络接口。

9. 一种获取驱动层数据包控制信息的系统,应用于 Android 系统中,Android 系统接受和发送数据包中源 MAC 地址和协议号之间增加有固定的一段特殊控制信息,其特征在于,所述系统包括:

数据包识别模块,用于驱动层识别获取包含所述特殊控制信息的数据包;

数据包存储模块,用于将该数据包存储在标准的 AndroidControlRec 结构体中;

数据包上传模块,通过 Android 系统调用 API 将驱动层中 AndroidControlRec 结构体中的所述数据包上传至协议栈模块;

数据包传递模块,通过协议栈模块,在不同的 Android 应用协议收发包中,接受

AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 Msghdr 结构体中;

AndroidApp 系统函数调用模块,用于调用协议栈中用以处理 Msghdr 结构体中的所述数据包;

上层应用获取模块,用于调用 Msghdr 结构体中的所述数据包并显示给用户。

10. 根据权利要求 9 所述的获取驱动层数据包控制信息的系统,其特征在于,所述协议栈模块包括 WAP 协议模块,ARP 协议模块,IP 协议模块以及 TCP 协议模块。

获取驱动层数据包控制信息的方法及系统

技术领域

[0001] 本发明涉及分组交换通信技术领域,特别是涉及 Android 系统的通信技术领域,具体为一种获取驱动层数据包控制信息的方法及系统。

背景技术

[0002] 随着 Android 系统应用越来越广泛,特别是基于 Android 智能手机大行其道。加上运营商对手机上网网络建设的快速发展,用户使用 Android 设备越来越多,如何分析和测试我们 Android 设备的上网性能成为一个迫切的问题。一般情况下,测试 Android 手机只能通过 Android SDK 提供的调试接口分析问题,不能直接通过应用层获取下层数据的特殊控制信息。

[0003] 而一些上层网络协议并不满足于单单获取原始的数据包内容,而是需要获取更多的数据包控制信息,比如手机连接网络中的物理端口号, VLAN ID 号,或者其他一些特殊控制信息。

[0004] 首先介绍一下现有处理方式,以 Android 智能终端为例。

[0005] 一般 Android 手机主板所带数据包信息会额外添加该 modem 芯片特有的标签信息,即所包含的该数据包来源的 MAC 芯片物理端口号,来源 VLAN Id 号, modem Id, RAT: 接入技术参数,可以是 GPRS/EDGE, WCDMA/HSDPA/LTE 等,来源的 QoS 队列号等。

[0006] 而一般所添加的位置为原始数据包的源 MAC 之后,协议号之前。格式一般为标签号加信息,长度也为固定长度。而作为上层协议栈对于原始数据包中的这些额外标签信息并不识别,所以驱动一般会对数据包做些特殊处理,即通过识别不同的协议包进行去除标签信息和保留标签信息操作。当然上层协议应用对于这些标签信息也需要做一些特殊的处理才能识别相应的控制信息。

[0007] 随着应用于 Android 手机上层协议越来越多,越来越复杂,驱动需要去识别的协议也越来越多,这势必会影响到驱动效率,而且这种处理方式本身也有悖与 Android 系统架构,驱动只需要关心底层数据包合法性和完整性即可,完全不需要关心数据包的具体内容。

[0008] 从以上解决技术方案来看,原有的 Android 系统上层网络应用使用标准的网络编程是无法获取数据包的这一部分控制信息,协议栈也并没有支持这种特殊需求,这就需要设计一种额外的机制去实现这种应用。而真正的难度在于如何兼容现有的协议栈,特别是兼容现有的 Android 标准网络编程。

发明内容

[0009] 鉴于以上所述现有技术的缺点,本发明的目的在于提供一种获取驱动层数据包控制信息的方法及系统,用于解决现有技术中 Android 上层应用无法实现获取驱动层数据包控制信息的问题。

[0010] 为实现上述目的及其他相关目的,本发明在一方面提供一种获取驱动层数据包控

制信息的方法,应用于 Android 系统中,Android 系统接受和发送数据包中源 MAC 地址和协议号之间增加有固定的一段特殊控制信息,所述方法包括:步骤 S1,驱动层识别获取包含所述特殊控制信息的数据包,并将该数据包存储在标准的 AndroidControlRec 结构体中;步骤 S2,Android 系统通过调用 API 将驱动层中存储在标准的 AndroidControlRec 结构体中的所述数据包上传至协议栈;步骤 S3,通过协议栈,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 MsgHdr 结构体中;步骤 S4,调用协议栈中用以处理 MsgHdr 结构体中的所述数据包的 AndroidApp 系统函数;步骤 S5,Android 上层应用调用 MsgHdr 结构体中的所述数据包并显示给用户。

[0011] 作为本发明的一种优选方案,在步骤 S1 中,所述特殊控制信息为 10 个字节。

[0012] 作为本发明的一种优选方案,在步骤 S1 中,所述特殊控制信息包含识别标签,队列号,物理端口号,Vlan 标签,Vlan 号,RAT 标签,RAT 信息。

[0013] 作为本发明的一种优选方案,在步骤 S1 中,若 Android 系统采用 GPRS 通信,所述特殊控制信息还包含 modem 相关接入层特殊控制信息。

[0014] 作为本发明的一种优选方案,在步骤 S2 中,协议栈包括 WAP 协议,ARP 协议,IP 协议以及 TCP 协议。

[0015] 作为本发明的一种优选方案,所述数据包上传至协议栈,所述协议栈首先处理是否为 ARP 协议,若是,先对 ARP 协议做地址解析,然后依次通过 IP 协议,TCP 协议将所述数据包上传至 Android 上层应用层;若否,直接通过 IP 协议,TCP 协议将所述数据包上传至 Android 上层应用层。

[0016] 作为本发明的一种优选方案,在协议栈中至少包含驱动上调函数 DriverKernel_rx,IP 层收包处理函数 ip_recv,应用层收包处理函数 AndroidApp_recvmsg。

[0017] 作为本发明的一种优选方案,AndroidApp 系统函数支持 Android 系统现有的网络接口。

[0018] 本发明在另外一方面提供一种获取驱动层数据包控制信息的系统,应用于 Android 系统中,Android 系统接受和发送数据包中源 MAC 地址和协议号之间增加有固定的一段特殊控制信息,所述系统包括:数据包识别模块,用于驱动层识别获取包含所述特殊控制信息的数据包;数据包存储模块,用于将该数据包存储在标准的 AndroidControlRec 结构体中;数据包上传模块,通过 Android 系统调用 API 将驱动层中 AndroidControlRec 结构体中的所述数据包上传至协议栈模块;数据包传递模块,通过协议栈模块,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 MsgHdr 结构体中;AndroidApp 系统函数调用模块,用于调用协议栈中用以处理 MsgHdr 结构体中的所述数据包;上层应用获取模块,用于调用 MsgHdr 结构体中的所述数据包并显示给用户。

[0019] 作为本发明的一种优选方案,所述协议栈模块包括 WAP 协议模块,ARP 协议模块,IP 协议模块以及 TCP 协议模块。

[0020] 如上所述,本发明的一种获取驱动层数据包控制信息的方法及系统,具有以下有益效果:

[0021] 1、本发明把包含特殊控制信息的数据包带入 AndroidControlRec 这个结构体中,因为整个协议栈都可以处理 AndroidControlRec 结构体中所带的数据包信息,而在上层传

输过程中,可以使用 MsgHdr 这个结构体,不同协议收包都有确认使用该 MsgHdr 结构体存储一些自带的控制信息,最后通系统调用函数把 MsgHdr 中的数据包传输到上层应用。所以本发明能彻底解放驱动对于这些数据包的内容检查判断,而且本发明也不会增加驱动负担。

[0022] 2、在本发明避免了对现有的 Android 架构和协议栈做大的改动,没有影响到其他正常的通信功能。

[0023] 3、在本发明达到分析 Android 设备的上网性能时也可以像分析电脑的各项性能指标一样简单明了,数据实时有效。

附图说明

[0024] 图 1 显示为本发明的一种获取驱动层数据包控制信息的流程示意图。

[0025] 图 2 显示为本发明的一种获取驱动层数据包控制信息的方法中实现传送数据包特殊控制信息的流程示意图。

[0026] 图 3 显示为本发明的一种获取驱动层数据包控制信息的方法中数据包格式及 MsgHdr 结构体示意图。

[0027] 图 4 显示为本发明的一种获取驱动层数据包控制信息的方法中 AndroidApp 系统函数(通信函数)列表。

[0028] 图 5 显示为本发明的一种获取驱动层数据包控制信息的系统的结构示意图。

具体实施方式

[0029] 以下通过特定的具体实例说明本发明的实施方式,本领域技术人员可由本说明书所揭露的内容轻易地了解本发明的其他优点与功效。本发明还可以通过另外不同的具体实施方式加以实施或应用,本说明书中的各项细节也可以基于不同观点与应用,在没有背离本发明的精神下进行各种修饰或改变。

[0030] 本发明涉及手机基于 GPRS/WIFI 等分组交换通信领域,本发明应用在 Android 系统中,实现应用层获取数据包特殊控制信息实现,像浏览器或者 Email 程序可以通过访问特殊提取函数提取控制信息。达到分析我们的 Android 设备的上网性能时,可以像分析电脑的各项性能指标一样简单明了,数据适时有效。所以我们需要设计的机制是能彻底解放驱动对于这些数据包的内容检查判断,按照特定结构存放到协议栈中,当有上层协议应用需要这些信息时,可通过特定调用去获取这些信息。所以本发明也不会增加驱动负担。而这些调用方式或者说机制就是本发明需要去设计并保护的内容。

[0031] 同时我们需要做到的是避免对现有的 Android 架构和协议栈做大的改动,从而影响到其他正常的通信功能。

[0032] 本发明的目的在于提供一种获取驱动层数据包控制信息的方法及系统,用于解决现有技术中 Android 上层应用无法实现获取驱动层数据包控制信息的问题。以下将详细阐述本发明的一种获取驱动层数据包控制信息的方法及系统的原理及实施方式,使本领域技术人员不需要创造性劳动即可理解本发明的一种获取驱动层数据包控制信息的方法及系统。

[0033] 请参阅图 1,显示为本发明的一种获取驱动层数据包控制信息的流程示意

图。如图 1 所示,本发明在一方面提供一种获取驱动层数据包控制信息的方法,

[0034] 步骤 S1,驱动层识别获取包含所述特殊控制信息的数据包,并将该数据包存储在标准的 AndroidControlRec 结构体中。

[0035] 步骤 S2,Android 系统通过调用 API 将驱动层中存储在标准的 AndroidControlRec 结构体中的所述数据包上传至协议栈。

[0036] 步骤 S3,通过协议栈,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 MsgHdr 结构体中。

[0037] 步骤 S4,调用协议栈中用以处理 MsgHdr 结构体中的所述数据包的 AndroidApp 系统函数。

[0038] 步骤 S5,Android 上层应用调用 MsgHdr 结构体中的所述数据包并显示给用户。

[0039] 在对上述各步骤进行详细说明以前,请参阅图 2。通过图 2 可以对本发明的方法有一个总体初步的认识。

[0040] 首先根据附图对本发明的技术方案有个初步的认识。

[0041] 如附图 2 中 Android 系统通过 GPRS/WLAN 网络收发网络数据,并将收发网络数据通过 GPRS/WLAN 承载业务层进入系统,系统的 MODEN/WIFE 驱动接受来自分组域或 WIFI 使用数据包,以 MODEN 为例一般会在数据包中自动插入一些特殊信息。例如附图 3 中显示的,会在 Android 系统接受和发送数据包的源 MAC 地址和协议号之间新增固定的一段特殊控制信息,一般为 10 个字节,分别为识别标签,队列号,物理端口号,Vlan 标签,Vlan 号,RAT 标签,RAT 信息。若是通过 GPRS 网络收发数据,也会标识 MODEN 相关接入层特殊控制信息。

[0042] 系统调用 API 接收来自驱动的数据包,并通过 WAP 协议模块传送数据包到 TCP/IP 协议栈,TCP/IP 协议栈首先处理是否为 ARP 协议,(这里 ARP 做地址解析,对 GPRS 上网的接入点等相关信息也做解析)若是,进入 APR 协议模块,然后从 APR 协议模块传送到 IP 协议模块,TCP/IP 协议栈首先处理是否为 ARP 协议时,若不是,则直接传输至 IP 协议模块,之后传输至 TCP 协议模块,最后上传至上层应用程序。

[0043] 如附图 4 中表示协议栈中通信函数列表,其中主要函数有 DriverKernel_rx,驱动上调函数;ip_recv,IP 层收包处理函数;AndroidApp_recvmsg,应用层收包处理函数。

[0044] 通过简单了解了整个数据包流程后,那么如何实现通过这些函数从驱动层传输特殊信息到上层应用? 本发明采用的方法是,把数据包带入 AndroidControlRec 这个结构体中,因为整个协议栈都可以处理 AndroidControlRec 中所带的数据包信息。而在上层传输过程中,可以使用 MsgHdr 这个结构体,不同协议收包都有确认使用该结构体存储一些自带的控制信息。最后本发明设计一套系统调用函数用以把 MsgHdr 中数据包信息带给上层应用,同时该系统调用也需要支持原有的网络接口。

[0045] 以下对上述各步骤进行详细说明。

[0046] 首先执行步骤 S1,驱动层识别获取包含所述特殊控制信息的数据包,并将该数据包存储在标准的 AndroidControlRec 结构体中。本发明把包含特殊控制信息的数据包带入 AndroidControlRec 这个结构体中,因为整个协议栈都可以处理 AndroidControlRec 结构体中所带的数据包信息。

[0047] AndroidControlRec 结构体为实现本发明所描述的功能添加的核心私有类,其功能是为获取的特殊资源做整合,提供程序输入输出便于实现相关功能和管理实现,并解析

相关特殊标签或信息。

[0048] 具体地,在步骤 S1 中,所述特殊控制信息一般为 10 个字节,可根据需要调节,可为 8 ~ 15 个字节。

[0049] 同时,在步骤 S1 中,所述特殊控制信息包含识别标签,队列号,物理端口号, Vlan 标签, Vlan 号, RAT 标签, RAT 信息。需要注意的是,若 Android 系统采用 GPRS 通信,所述特殊控制信息还包含 modem 相关接入层特殊控制信息,例如 modem ID 号等信息。接着执行步骤 S2。

[0050] 在步骤 S2 中,Android 系统通过调用 API 将驱动层中存储在标准的 AndroidControlRec 结构体中的所述数据包上传至协议栈。

[0051] 具体地,在步骤 S2 中,协议栈包括 WAP 协议, ARP 协议, IP 协议以及 TCP 协议。

[0052] 同时在本实施例中,所述数据包上传至协议栈,所述协议栈首先处理是否为 ARP 协议,若是先对 ARP 协议做地址解析,然后继续执行步骤 S3,然后依次通过 IP 协议, TCP 协议将所述数据包上传至 Android 上层应用层;若否,直接通过 IP 协议, TCP 协议将所述数据包上传至 Android 上层应用层。

[0053] 在步骤 S3 中,通过协议栈,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 Msghdr 结构体中。Msghdr 结构体如图 3 所示。在上层传输过程中,可以使用 Msghdr 这个结构体,不同协议收包都有确认使用该 Msghdr 结构体存储一些自带的控制信息,最后通系统调用函数把 Msghdr 中的数据包传输到上层应用。

[0054] 在步骤 S4 中,调用协议栈中用以处理 Msghdr 结构体中的所述数据包的 AndroidApp 系统函数。在协议栈中包含若干通信函数,至少包含驱动上调函数 DriverKernel_rx, IP 层收包处理函数 ip_recv,应用层收包处理函数 AndroidApp_recvmsg 等上调函数。

[0055] 具体地,包含的函数如图 4 中所示的列表,具体包括:通信函数 sock_read, sock_recvmsg, sys-read, inet_recvmsg;应用层收包处理函数 AndroidApp_recvmsg, AndoidControl_recv_data, sock_queue_rcv_skb, AndroidApp_queue_rcv_skb, AndroidApp_rcv;ip_local_deliver_finish, ip_local_deliver, IP 层收包处理函数 ip_recv, net_rc_action,驱动上调函数 DriverKernel_rx。

[0056] 下面都是 TCP/IP 协议的公知函数,均为本领域技术人员所熟知。这里只做简单描述, sock_read:套接字读取;sock_recvmsg:套接字消息结构;sys-read:系统内核调用;inet_recvmsg:因特网接收;ip_local_deliver_finish:ip 发送确认;ip_local_deliver:ip 发送;ip_recv:IP 层收包处理函数;DriverKernel_rx:驱动上调函数。

[0057] Android 应用层收包处理函数 AndroidApp_recvmsg, AndoidControl_recv_data, sock_queue_rcv_skb, AndroidApp_queue_rcv_skb, AndroidApp_rcv。

[0058] 这里 AndroidApp_rcv 用于接收获取下层特殊的控制信息, AndroidApp_queue_rcv_skb 负责将获取的控制信息排队等候处理, sock_queue_rcv_skb 负责将 AndroidApp_queue_rcv_skb 信息管理成 TCP/IP 套接字可接收的队列信息, AndoidControl_recv_data 负责解析获取的控制信息,并提供可支持 TCP/IP 原有的网络接口的 API,这样保证公知的套接字接口也可调用。AndroidApp_recvmsg 负责提供 Android 系统可以调用显示的 API,

Android 应用程序可以通过此接口生成用户可以见的 Android activity。

[0059] 此外,需要注意的是,AndroidApp 系统函数都应该支持 Android 系统现有的网络接口。

[0060] 步骤 S5,Android 上层应用调用 MsgHdr 结构体中的所述数据包并显示给用户。这样,通过本发明的方法 Android 上层应用便获取了驱动层数据包的控制信息。可见,在本发明的方法达到分析 Android 设备的上网性能时也可以像分析电脑的各项性能指标一样简单明了,数据实时有效。

[0061] 为实现上述方法,如图 5 所示,本发明在另外一方面提供一种获取驱动层数据包控制信息的系统,应用于 Android 系统中,Android 系统接受和发送数据包中源 MAC 地址和协议号之间增加有固定的一段特殊控制信息,具体地,所述系统 1 包括:数据包识别模块 11、数据包存储模块 12、数据包上传模块 13、数据包传递模块 14、AndroidApp 系统函数调用模块 15 和上层应用获取模块 16。

[0062] 所述数据包识别模块 11 用于驱动层识别获取包含所述特殊控制信息的数据包。所述特殊控制信息一般为 10 个字节,可根据需要调节,可为 8~15 个字节。所述特殊控制信息包含识别标签,队列号,物理端口号,Vlan 标签,Vlan 号,RAT 标签,RAT 信息。需要注意的是,若 Android 系统采用 GPRS 通信,所述特殊控制信息还包含 modem 相关接入层特殊控制信息,例如 modem ID 号等信息。

[0063] 所述数据包存储模块 12 用于将该数据包存储在标准的 AndroidControlRec 结构体中。本发明把包含特殊控制信息的数据包带入 AndroidControlRec 这个结构体中,因为整个协议栈都可以处理 AndroidControlRec 结构体中所带的数据包信息。

[0064] AndroidControlRec 结构体为实现本发明所描述的功能添加的核心私有类,其功能是为获取的特殊资源做整合,提供程序输入输出便于实现相关功能和管理实现,并解析相关特殊标签或信息。

[0065] 所述数据包上传模块 13 通过 Android 系统调用 API 将驱动层中 AndroidControlRec 结构体中的所述数据包上传至协议栈模块 17。具体地,在本实施例中,所述协议栈模块 17 包括 WAP 协议模块,ARP 协议模块,IP 协议模块以及 TCP 协议模块。

[0066] 同时在本实施例中,所述数据包上传至协议栈模块 17,协议栈模块 17 首先处理是否为 ARP 协议,若是通过 ARP 协议模块先对 ARP 协议做地址解析,然后再依次通过 IP 协议模块,TCP 协议模块将所述数据包上传至 Android 上层应用层;若协议栈模块 17 首先处理不是 ARP 协议,直接通过 IP 协议模块,TCP 协议模块将所述数据包上传至 Android 上层应用层。

[0067] 所述数据包传递模块 14 通过协议栈模块 17,在不同的 Android 应用协议收发包中,接受 AndroidControlRec 结构体中的所述数据包,并最终将所述数据包传递到 MsgHdr 结构体中。MsgHdr 结构体如图 3 所示。在上层传输过程中,可以使用 MsgHdr 这个结构体,不同协议收包都有确认使用该 MsgHdr 结构体存储一些自带的控制信息,最后通系统调用函数把 MsgHdr 中的数据包传输到上层应用。

[0068] 所述 AndroidApp 系统函数调用模块 15 用于调用协议栈中用以处理 MsgHdr 结构体中的所述数据包。调用协议栈中用以处理 MsgHdr 结构体中的所述数据包的 AndroidApp 系统函数。在协议栈中包含若干通信函数,至少包含驱动上调函数 DriverKernel_rx,IP 层

收包处理函数 `ip_recv`, 应用层收包处理函数 `AndroidApp_recvmsg` 等上调函数。

[0069] 具体地, 包含的函数如图 4 中所示的列表, 具体包括: 通信函数 `sock_read`, `sock_recvmsg`, `sys-read`, `inet_recvmsg`; 应用层收包处理函数 `AndroidApp_recvmsg`, `AndoidControl_recv_data`, `sock_queue_rcv_skb`, `AndroidApp_queue_rcv_skb`, `AndroidApp_rcv`; `ip_local_deliver_finish`, `ip_local_deliver`, IP 层收包处理函数 `ip_recv`, `net_rc_action`, 驱动上调函数 `DriverKernel_rx`。

[0070] 下面都是 TCP/IP 协议的公知函数, 均为本领域技术人员所熟知。这里只做简单描述, `sock_read`: 套接字读取; `sock_recvmsg`: 套接字消息结构; `sys-read`: 系统内核调用; `inet_recvmsg`: 因特网接收; `ip_local_deliver_finish`: ip 发送确认; `ip_local_deliver`: ip 发送; `ip_recv`: IP 层收包处理函数; `DriverKernel_rx`: 驱动上调函数。

[0071] Android 应用层收包处理函数 `AndroidApp_recvmsg`, `AndoidControl_recv_data`, `sock_queue_rcv_skb`, `AndroidApp_queue_rcv_skb`, `AndroidApp_rcv`。

[0072] 这里 `AndroidApp_rcv` 用于接收获取下层特殊的控制信息, `AndroidApp_queue_rcv_skb` 负责将获取的控制信息排队等候处理, `sock_queue_rcv_skb` 负责将 `AndroidApp_queue_rcv_skb` 信息管理成 TCP/IP 套接字可接收的队列信息, `AndoidControl_recv_data` 负责解析获取的控制信息, 并提供可支持 TCP/IP 原有的网络接口的 API, 这样保证公知的套接字接口也可调用。 `AndroidApp_recvmsg` 负责提供 Android 系统可以调用显示的 API, Android 应用程序可以通过此接口生成用户可以见的 Android activity。

[0073] 此外, 需要注意的是, `AndroidApp` 系统函数都应该支持 Android 系统现有的网络接口。

[0074] 所述上层应用获取模块 16 用于调用 `Msghdr` 结构体中的所述数据包并显示给用户。这样, 通过本发明的方法 Android 上层应用便获取了驱动层数据包的控制信息。可见, 在本发明的方法达到分析 Android 设备的上网性能时也可以像分析电脑的各项性能指标一样简单明了, 数据实时有效。

[0075] 综上所述, 本发明的一种获取驱动层数据包控制信息的方法及系统, 达到了以下有益效果:

[0076] 1、本发明把包含特殊控制信息的数据包带入 `AndroidControlRec` 这个结构体中, 因为整个协议栈都可以处理 `AndroidControlRec` 结构体中所带的数据包信息, 而在上层传输过程中, 可以使用 `Msghdr` 这个结构体, 不同协议收包都有确认使用该 `Msghdr` 结构体存储一些自带的控制信息, 最后通系统调用函数把 `Msghdr` 中的数据包传输到上层应用。所以本发明能彻底解放驱动对于这些数据包的内容检查判断, 而且本发明也不会增加驱动负担。

[0077] 2、在本发明避免了对现有的 Android 架构和协议栈做大的改动, 没有影响到其他正常的通信功能。

[0078] 3、在本发明达到分析 Android 设备的上网性能时也可以像分析电脑的各项性能指标一样简单明了, 数据实时有效。

[0079] 所以, 本发明有效克服了现有技术中的种种缺点而具高度产业利用价值。

[0080] 上述实施例仅例示性说明本发明的原理及其功效, 而非用于限制本发明。任何熟悉此技术的人士皆可在不违背本发明的精神及范畴下, 对上述实施例进行修饰或改变。因

此,举凡所属技术领域中具有通常知识者在未脱离本发明所揭示的精神与技术思想下所完成的一切等效修饰或改变,仍应由本发明的权利要求所涵盖。

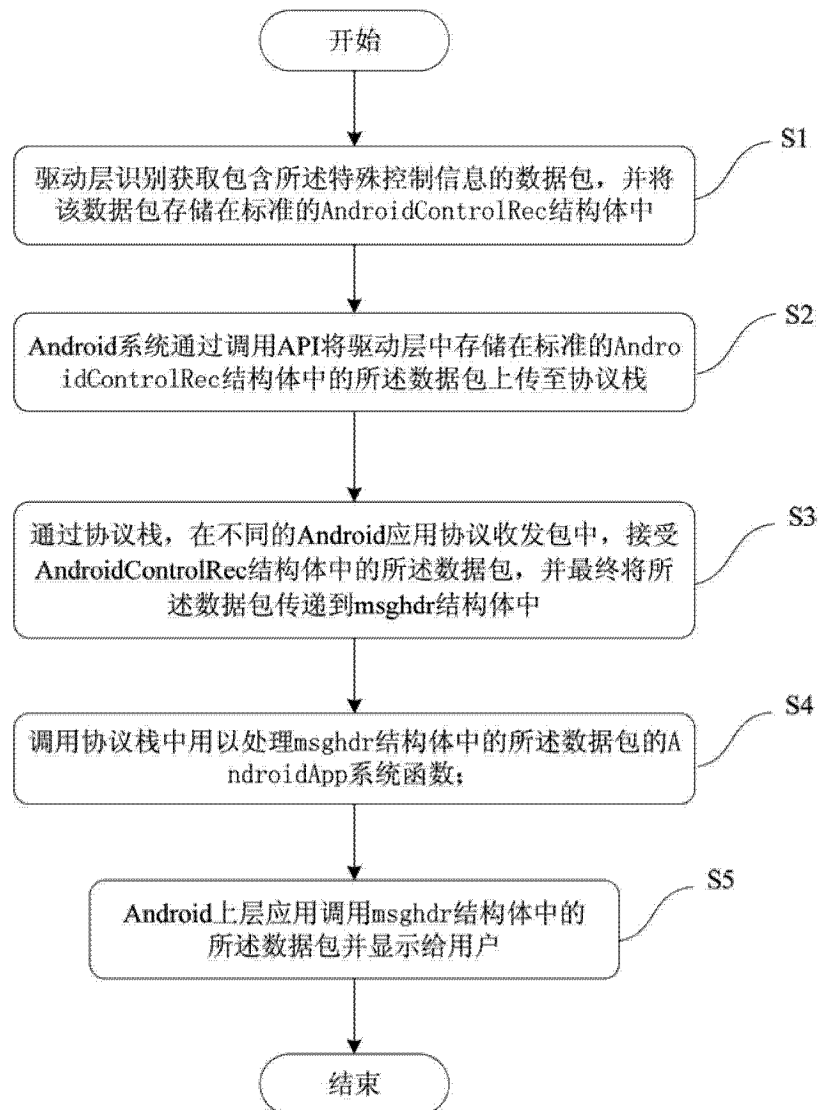


图 1

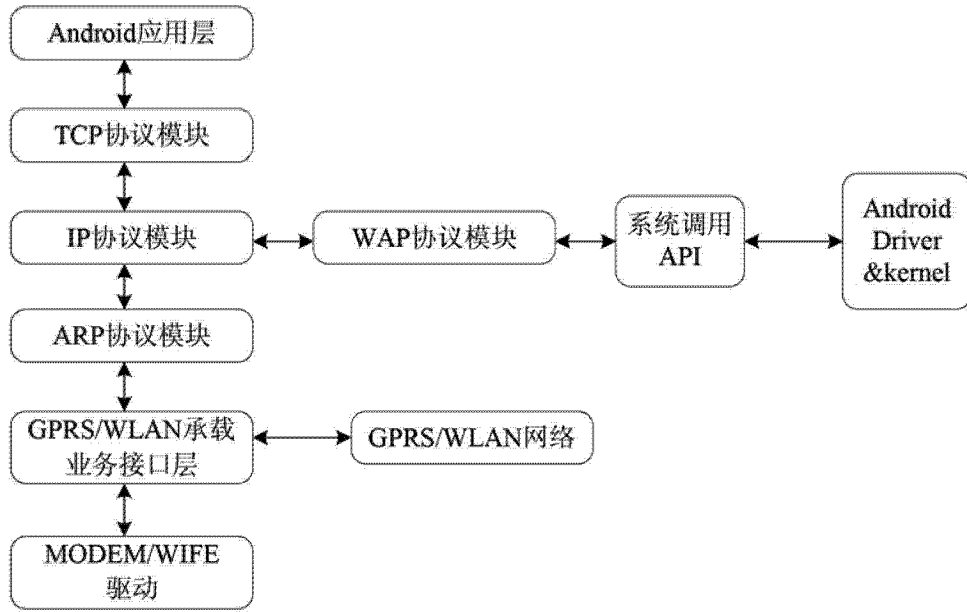


图 2

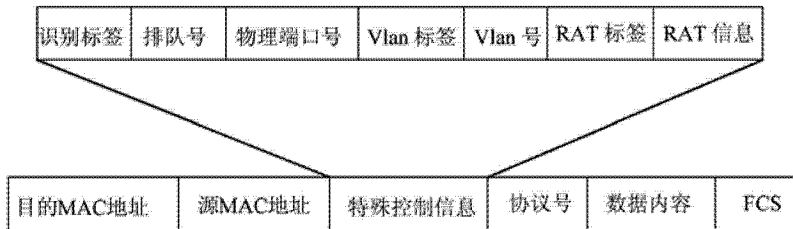


图 3

sock_read
sock_recvmsg
sys_read
inet_recvmsg
AndroidApp_recvmsg
AndoidControl_recv_data
sock_queue_rcv_skb
AndroidApp_queue_rcv_skb
AndroidApp_rcv
ip_local_deliver_finish
ip_local_deliver
ip_rcv
net_rc_action
DriverKernel_rx

图 4

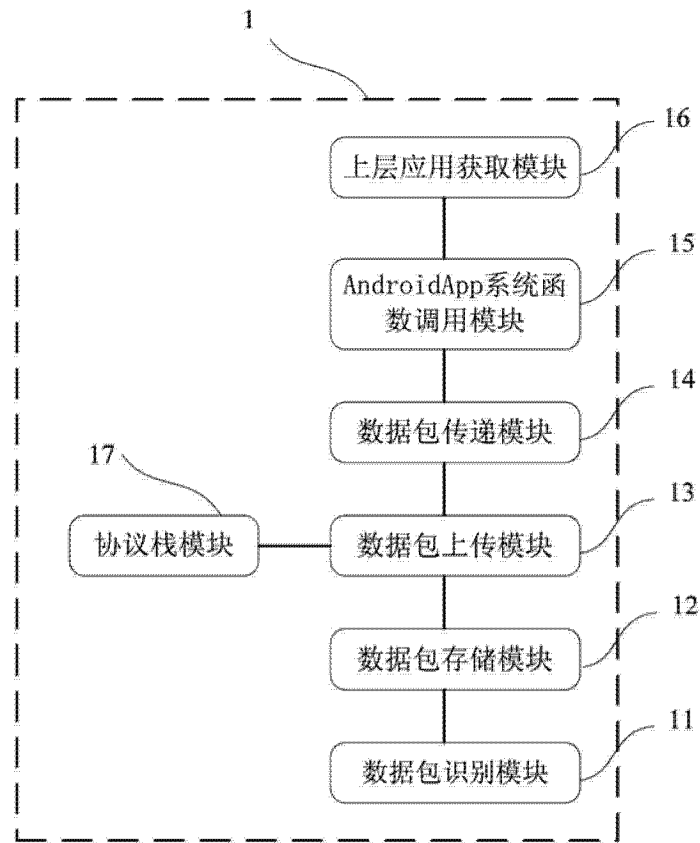


图 5