



(12) 发明专利

(10) 授权公告号 CN 101572670 B

(45) 授权公告日 2011.08.10

(21) 申请号 200910137700.4

(22) 申请日 2009.05.07

(73) 专利权人 成都市华为赛门铁克科技有限公司

地址 611731 四川省成都市高新区西部园区清水河片区

(72) 发明人 熊鹰

(74) 专利代理机构 深圳市深佳知识产权代理事务所(普通合伙) 44285

代理人 彭愿洁 李文红

(51) Int. Cl.

H04L 12/56(2006.01)

H04L 29/06(2006.01)

G06F 17/30(2006.01)

(56) 对比文件

CN 101286936 A, 2008.10.15, 全文.

US 2007/0168377 A1, 2007.07.19, 全文.

CN 1937574 A, 2007.03.28, 全文.

JP 特开 2004-56340 A, 2004.02.19, 全文.

CN 101247337 A, 2008.08.20, 全文.

审查员 谢正程

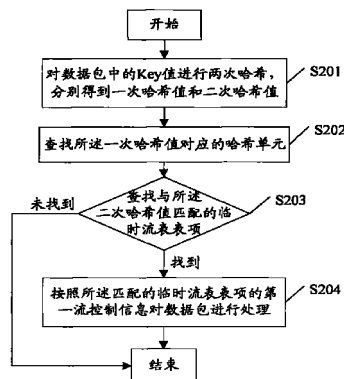
权利要求书 2 页 说明书 9 页 附图 5 页

(54) 发明名称

一种基于流表的数据包处理方法、装置和网络系统

(57) 摘要

本发明实施例公开了一种基于流表的数据包处理方法,所述流表包括哈希单元和临时流表,所述方法包括:对数据包中的键值进行两次哈希,分别得到一次哈希值和二次哈希值;查找与所述一次哈希值对应的哈希单元;在与所述哈希单元对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,并按照匹配的临时流表表项中的第一流控制信息对数据包进行处理。本发明技术方案由于在临时流表中不保存 Key 值,临时流表规格较小,且在匹配时只需要将二次哈希值进行匹配,而不需要与流表表项中的 Key 值进行一一匹配,因此可以节约存储空间并提高 CPU 性能。



1. 一种基于流表的数据包处理方法,其特征在于,所述流表包括哈希单元和临时流表,所述方法包括:

对数据包中的键值进行两次哈希,分别得到一次哈希值和二次哈希值,所述两次哈希采用不同的哈希算法,或者从所述数据包的键值中选取不同的内容进行哈希,或者对两次哈希值设定不同的哈希值长度;

查找与所述一次哈希值对应的哈希单元;

在与所述哈希单元对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,并按照匹配的临时流表表项中的第一流控制信息对数据包进行处理。

2. 如权利要求1所述的基于流表的数据包处理方法,其特征在于,所述流表中还包括正式流表,所述方法还包括:

当没有查找到与所述二次哈希值匹配的临时流表表项时,如果存在与所述哈希单元对应的正式流表,则根据与所述数据包中的键值完全匹配的正式流表表项中的第二流控制信息对所述数据包进行处理;否则,按照系统默认设置或策略对所述数据包进行处理,并创建与所述哈希单元对应的临时流表表项。

3. 如权利要求1所述的基于流表的数据包处理方法,其特征在于,还包括:

修改所述匹配的临时流表表项中的统计值;

当所述统计值超过预设阈值时,将所述临时流表表项修改为正式流表表项,并删除所述临时流表表项。

4. 如权利要求2所述的基于流表的数据包处理方法,其特征在于,所述创建与所述哈希单元对应的临时流表表项包括:

在所述临时流表中空闲的临时流表表项中建立新的临时流表表项;或

通过对统计值最少的临时流表表项或无数据包时间最长的临时流表表项进行覆盖,建立新的临时流表表项。

5. 如权利要求2所述的基于流表的数据包处理方法,其特征在于,所述第一流控制信息包括:对数据包进行转发或丢弃;所述第二流控制信息包括:对数据包进行转发、流量复制、带宽限制或丢弃。

6. 一种数据包处理装置,其特征在于,所述装置中预置有流表,所述流表包括哈希单元和临时流表,所述装置包括:

哈希处理单元,用于对数据包中的键值进行两次哈希,得到一次哈希值和二次哈希值,所述两次哈希采用不同的哈希算法,或者从所述数据包的键值中选取不同的内容进行哈希,或者对两次哈希值设定不同的哈希值长度;

第一查找单元,用于查找与所述一次哈希值对应的哈希单元;

第二查找单元,用于在所述哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项;

第一处理单元,用于根据与所述二次哈希值匹配的临时流表表项中的第一流控制信息对数据包进行处理。

7. 如权利要求6所述的数据包处理装置,其特征在于,所述流表还包括正式流表,所述装置还包括:

第一判断单元,用于当所述第二查找单元没有查找到与所述二次哈希值匹配的临时流

表表项时,判断是否存在与所述哈希单元对应的正式流表;

第二处理单元,用于当存在与所述哈希单元对应的正式流表时,根据与所述数据包中的键值完全匹配的正式流表表项中的第二流控制信息对所述数据包进行处理;当不存在与所述哈希单元对应的正式流表时,按照系统默认配置或策略对数据包进行处理。

8. 如权利要求 7 所述的数据包处理装置,其特征在于,还包括:

第一流表维护单元,用于当所述第二查找单元查找到与所述二次哈希值匹配的临时流表表项时,修改所述临时流表表项中的统计值;当所述统计值超过预设阈值时,将所述匹配的临时流表表项修改为正式流表表项,并删除所述临时流表表项;

第二流表维护单元,用于在不存在与所述哈希单元对应的正式流表时,创建与所述哈希单元对应的临时流表表项。

9. 一种数据处理系统,其特征在于,包括数据包接收装置和权利要求 6 至 8 任一项所述的数据包处理装置,所述数据包接收装置用于接收数据包,所述数据包处理装置用于对所述数据包接收装置所接收到的数据包进行控制处理。

一种基于流表的数据包处理方法、装置和网络系统

技术领域

[0001] 本发明涉及通信技术领域,具体涉及一种基于流表的数据包处理方法、装置和网络系统。

背景技术

[0002] 基于流表的网际协议 (IP, International Protocol) 数据包转发是目前许多业务相关的网络设备如防火墙、虚拟专用网络 (VPN, Virtual Private Network) 设备采用的技术。基于流表的快速转发要求为每个目的地建立一个转发表项,与传统的 IP 转发路由器或者交换机不同,流表的用途是记录基于传输控制协议 / 数据用户报协议 (TCP/UDP, Transmission Control Protocol/User Datagram Protocol) 或应用层的一些状态,以服务于对应的业务。

[0003] 流表也称会话表 (Session Table),是 IP 数据包转发的核心数据表,流表的建立和维护都较为复杂,存在流表表项数量规格问题。表项过多,可能浪费内存,增加成本;表项过少,则有时候可能会不够使用而影响转发。通常的做法是从数量上折中,满足大部分场景需要,特殊场景下流表表项不够时,采取默认的数据包处理方式。

[0004] 现有技术中常见的流表包括哈希单元和正式流表,正式流表的表项内容随着业务的不同而有所不同,通常包括:用于精确比较的键 (Key) 值 (Key 值通常为数据包的五元组信息 (源 IP 地址,目的 IP 地址,协议类型,源端口号,目的端口号))、用于老化的时间戳、数据流的统计信息等状态信息。其中,一个数据流中的数据包具有相同的 Key 值,具有不同 Key 值的数据包属于不同的数据流。

[0005] 现有技术中通常采用如下方法建立流表并对数据包进行处理:对数据包的 Key 值进行哈希 (也译作散列, hash) 后,找到对应的哈希单元,并将所述数据包中的 Key 值与正式流表中各个表项的 Key 值进行匹配,如果与某个表项的 Key 值匹配,则修改所述正式流表中对应表项的流状态信息,并按照所述对应表项中的流控制信息对数据包进行处理,例如可以根据具体的流控制信息对数据包进行转发、流量复制、带宽限制,或者将所述数据包丢弃。当正式流表中没有与该数据包的 Key 值匹配的表项时,则在该正式流表中新建流表表项,并将该数据包的 Key 值以及该数据包所在数据流的状态信息和控制信息存储于该正式流表表项中。创建正式流表表项时,可以采用链表或二叉树等方式处理哈希值相同而 Key 值不同的各个表项之间的关系。

[0006] 在对现有技术的研究和实践过程中,本发明的发明人发现,采用现有技术处理数据包时,由于流表表项内容包括 Key 值、流状态信息以及流控制信息,数据流表规格较大,在对数据包进行处理过程中,需要消耗较多的 CPU 资源对数据流表进行维护,然而,在当前互联网上,包数很少的短连接非常多,尤其是涉及部分 P2P 应用以及 DoS 攻击等,此时,若采用现有技术中的流表对这些类型的数据包进行处理会占用大量的 CPU 资源,降低资源利用效率。

发明内容

[0007] 本发明实施例的目的在于提供一种基于流表的数据包处理方法、装置和网络系统,能够提高资源利用效率。

[0008] 根据本发明实施例的一个方面,提供了一种基于流表的数据包处理方法,所述流表包括哈希单元和临时流表,所述方法包括:

[0009] 对数据包中的键值进行两次哈希,分别得到一次哈希值和二次哈希值,所述两次哈希采用不同的哈希算法,或者从所述数据包的键值中选取不同的内容进行哈希,或者对两次哈希值设定不同的哈希值长度;

[0010] 查找与所述一次哈希值对应的哈希单元;

[0011] 在与所述哈希单元对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,并按照匹配的临时流表表项中的第一流控制信息对数据包进行处理。

[0012] 根据本发明实施例的另一方面,提供了一种数据包处理装置,所述装置中预置有流表,所述流表包括哈希单元和临时流表,所述装置包括:

[0013] 哈希处理单元,用于对数据包中的键值进行两次哈希,得到一次哈希值和二次哈希值,所述两次哈希采用不同的哈希算法,或者从所述数据包的键值中选取不同的内容进行哈希,或者对两次哈希值设定不同的哈希值长度;

[0014] 第一查找单元,用于查找与所述一次哈希值对应的哈希单元;

[0015] 第二查找单元,用于在所述哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项;

[0016] 第一处理单元,用于根据与所述二次哈希值匹配的临时流表表项中的第一流控制信息对数据包进行处理。

[0017] 根据本发明实施例的再一方面,提供一种数据处理系统,包括数据包接收装置和数据包处理装置,所述数据包接收装置用于接收数据包;所述数据包处理装置用于对数据包接收装置所接收到的数据包进行控制处理,所述数据包处理装置中预置有流表,所述流表包括哈希单元和临时流表,所述数据包处理装置包括:

[0018] 哈希处理单元,用于对数据包中的键值进行两次哈希,得到一次哈希值和二次哈希值;

[0019] 第一查找单元,用于查找与所述一次哈希值对应的哈希单元;

[0020] 第二查找单元,用于在所述哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项;

[0021] 第一处理单元,用于根据与所述二次哈希值匹配的临时流表表项中的第一流控制信息对数据包进行处理。

[0022] 本发明实施例所采用的流表中包含哈希单元和临时流表,其中:临时流表的表项内容包含二次哈希值信息和第一流控制信息,处理数据包时,对数据包中的 Key 值进行两次哈希,并根据一次哈希值查找到对应的哈希单元,在所述找到的哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,并按照与所述二次哈希值匹配的临时流表表项中的第一流控制信息对数据包进行处理。由于临时流表中不保存 Key 值,临时流表规格较小,因此可以节约存储空间,且在匹配时只需要将二次哈希值进行匹配,而不需要与流表表项中的 Key 值进行一一匹配,因此可以提高 CPU 性能,综上所述,上述方案可以

提高 CPU 以及内存等资源的利用效率。本发明实施例尤其适用于当前互联网上,数据包数很少的短连接非常多的情况,例如部分 P2P 应用以及扫描和 DoS 攻击流量的情况。

附图说明

[0023] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0024] 图 1 是现有技术中流表结构示意图;

[0025] 图 2 是本发明实施例一基于流表的数据包转发方法流程图;

[0026] 图 3 是本发明实施例中所示的一种流表结构示意图;

[0027] 图 4 是本发明实施例二基于流表的数据包处理方法流程图;

[0028] 图 5 是本发明实施例三基于流表的数据包处理方法流程图;

[0029] 图 6 是本发明实施例四基于流表的数据包处理方法流程图;

[0030] 图 7 是本发明实施例五数据包处理装置结构示意图;

[0031] 图 8 是本发明实施例六数据包处理装置结构示意图;

[0032] 图 9 是本发明实施例七数据包处理装置结构示意图;

[0033] 图 10 是本发明实施例八数据处理系统结构示意图。

具体实施方式

[0034] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0035] 在实现本发明实施例的过程中,发明人发现,当前互联网上,数据包数很少的短连接非常多,如果为每个数据流都建立完整的正式流表表项(表项内容包括 Key 值、流状态信息及流控制信息),则会占用过多的内存,并且使 CPU 负荷过重。例如,完整的正式流表中每个表项都需要保存 Key 值,对于 IP 数据包来说,包括 IP 五元组(源 IP 地址,目的 IP 地址,IP 协议号,源端口号,目的端口号),由于保存了更多关于数据流的信息,因此需要消耗更多的内存资源,并且相应地,也需要更多的 CPU 资源进行匹配和处理。而对于如部分 P2P 应用以及扫描和拒绝服务(DoS, Denial of Service)攻击流量,数据包可能是无效的,这些无效的数据包使得 CPU 利用效率降低,而如果不加区分的对新建数据流不建流表或者不加区分地对已建流表进行老化,都可能损害正常业务,且无法避免无效数据流对内存及 CPU 资源的消耗。

[0036] 参照图 1,为现有技术中流表结构示意图,包括哈希单元 11 和正式流表 12,正式流表的表项内容通常包括:用于精确比较的 Key 值(如 IP 五元组(源 IP 地址,目的 IP 地址,IP 协议号,源端口号,目的端口号)),用于老化的时间戳,数据流的统计信息等流状态信息(随业务不同而不同,例如,作网络地址转换(NAT, Network Address Translation)时,需要保存 NAT 后的 IP 地址信息)以及流控制信息。流控制信息通常包括:对数据包进行转发、

流量复制、带宽限制,或者将所述数据包丢弃。当接收到的数据包与所述正式流表中某个表项的 Key 值匹配时,修改所述匹配表项的流状态信息,如用于老化的时间戳、数据流的统计信息等,并按照所述流控制信息中的具体内容对数据包进行处理。

[0037] 以下首先介绍本发明实施例数据包处理过程中所采用的流表,该流表包括:哈希单元和临时流表,临时流表中的表项内容包括:二次哈希值信息和第一流控制信息。第一流控制信息可以只是对数据包进行丢弃或转发,以进一步提高 CPU 资源的利用效率,减少 CPU 资源消耗。以下通过具体实施例介绍采用该流表的数据包处理方法:

[0038] 图 2 为本发明实施例一基于流表的数据包处理方法流程图,其中流表包括:哈希单元和临时流表,临时流表中的表项内容包括:二次哈希值信息和第一流控制信息。该方法包括:

[0039] S201、对数据包中的 Key 值进行两次哈希,分别得到一次哈希值和二次哈希值;

[0040] S202、查找与所述一次哈希值对应的哈希单元;

[0041] S203、在找到的哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,如果找到与所述二次哈希值匹配的临时流表表项,则执行 S204;否则,结束流程;

[0042] S204、按照所述匹配的临时流表表项的第一流控制信息对数据包进行处理。

[0043] 在具体实施中,第一流控制信息可以只是对数据包进行转发或者丢弃,而不包含流量复制、带宽限制等复杂的流控制操作,以进一步减少 CPU 消耗,提高 CPU 资源的利用效率。

[0044] 从本实施例可以看出,由于临时流表中不保存 Key 值,临时流表规格较正式流表小很多,因此可以节约存储空间,且在匹配时只需要将二次哈希值进行匹配,而不需要与流表表项中的 Key 值进行一一匹配,因此可以提高 CPU 性能。综上可知,上述数据包处理方法可以提高 CPU 以及内存等资源的利用效率。本发明实施例尤其适用于当前互联网上,数据包数很少的短连接非常多的情况,例如部分 P2P 应用以及扫描和 DoS 攻击流量的情况。

[0045] 本发明发明人可以在上述流表基础上进一步扩展,使其兼容现有正式流表,参照图 3,为本发明实施例中一种流表结构示意图,包括哈希单元 11、正式流表 12 和临时流表 31,其中临时流表 31 的表项内容包含二次哈希值 311 和统计值 312 和第一流控制信息 313。正式流表 12 则包括 Key 值、流控制信息和第二流控制信息(图 3 中未示出)。

[0046] 在具体实施中,如图 3 所示,一个哈希单元下有一个临时流表,临时流表可以通过指针指向正式流表。每个临时流表中可以有至少一个表项,如图 3 中所示的表项 1、表项 2..... 表项 M,每个临时流表表项内容包括:二次哈希值 311 和统计值 312 和第一流控制信息 313。统计值 312 可以是数据包数统计值,也可以是字节数统计值。

[0047] 通常,为了避免不同的数据流产生混淆(即不同的数据流经过一次哈希后得到相同哈希值),提高数据包处理的准确性,减少误判率,本发明实施例采用对数据包中的 Key 值进行两次哈希的方法。并且两次哈希可以采用不同的哈希算法,也可以从所述数据包的 Key 值中选取不同的内容进行哈希(例如,第一次哈希时对 Key 值所有内容进行哈希,第二次哈希时只对源 IP 地址和目的 IP 地址进行哈希),还可以对两次哈希值设定不同的哈希值长度,例如:一次哈希值设为 30bit、二次哈希值设为 20 比特。同时,为进一步减小 CPU 负荷,第二次哈希可以做得比第一次哈希更简化。由于算法不同且可以选择数据包中 Key 值

的不同内容进行哈希,哈希值长度也可以灵活设置,因此数据包通过两次哈希后得到相同的哈希值的概率很小,提高了数据包处理的准确性。

[0048] 除了统计值,临时流表中还可以保留一些其他可选的流状态信息。例如,记录时间戳等。其中,时间戳可以是第一次收到数据包报文的时间,也可以是最后一次收到数据包报文的时间,或者二者兼有。

[0049] 可以理解的是,为了进一步减少 CPU 负荷,临时流表中的表项数量可以根据用户的需要灵活确定。例如,假设使用传统的正式流表时,正式流表中不同 Key 值的数据包哈希到同一哈希单元下的概率为 30%,则临时流表中可以只放 1-2 个表项。

[0050] 此外,如图 3 所示,在每个哈希单元下还可以记录该哈希单元下的流表信息以及指向流表的指针。其中,在流表信息中可以保存与该哈希单元对应的正式流表表项数目与临时流表表项数目,此外,在各个哈希单元下还可以预留一些区域以供后续扩展使用,如图 3 中“预留”区域所示。

[0051] 图 4 是本发明实施例二基于流表的数据包处理方法流程图,该处理方法可基于图 3 所示流表对数据包进行处理并对图 3 所示流表进行维护,所述流表包括:哈希单元、临时流表和正式流表,其中:所述临时流表表项内容包括:二次哈希值信息、统计值和流控制信息;正式流表表项内容包括 Key 值、数据包所在的流的状态信息和流控制信息等。所述统计值可以是数据包数统计值,也可以数据包的字节数统计值等信息,为了清楚的进行说明,本发明以数据包数统计值为例进行描述。所述流控制信息具体可以是对数据包设置的一些操作,如丢弃,转发,流量复制,以及对数据流进行带宽限制等。如图 4 所示,该方法包括:

[0052] S401、对数据包中的 Key 值进行两次哈希,分别得到一次哈希值和二次哈希值;

[0053] 例如接收到的数据包中的 Key 值为 IP 五元组,通过将所述 Key 值进行两次哈希,分别得到一次哈希值和二次哈希值。两次哈希通常可以采用不同的哈希算法。

[0054] S402、查找与所述一次哈希值对应的哈希单元;

[0055] S403、在找到的哈希单元所对应的所述临时流表中查找与所述二次哈希值匹配的临时流表表项,如果找到与所述二次哈希值匹配的临时流表表项,则执行 S404;如果未找到与所述二次哈希值匹配的临时流表表项,则结束流程;

[0056] S404、按照匹配的临时流表表项中的第一流控制信息对数据包进行处理并修改所述匹配的临时流表表项中的数据包数统计值;

[0057] 具体的,如果二次哈希值与临时流表中某表项中的二次哈希值匹配,可以将所述匹配的临时流表表项中的数据包数统计值加 1。

[0058] 可以理解的是,当所述统计值为数据包的字节数统计值时,则按照匹配的临时流表表项中的第一流控制信息对数据包进行处理并增加统计到的流量中的字节数。

[0059] 可以理解的是,所述临时流表中还可以包含其他一些自定义的流状态信息,如记录时间戳等,当包括这些自定义的流状态信息时,则需要同时进行相应修改。

[0060] 在具体实施中,流控制信息可以只包括对数据包进行转发或丢弃的指示信息,则可以根据所述流控制信息对数据包进行相应处理。由于不需要对数据包进行流量复制、带宽限制等复杂的处理,因此可以减少 CPU 消耗,提高 CPU 的利用效率。

[0061] S405、判断所述数据包数统计值是否超过预设阈值,如果是,则执行 S406;否则结束流程;

[0062] 具体的,可以设置临时流表表项转为正式流表表项的数据包包数阈值,例如 3 个数据包,一旦一个数据流的数据包数统计值达到该阈值,则说明该数据流为长连接的数据流,则可以将所述临时流表表项转为正式流表表项,以通过正式流表表项对该数据流的后续数据包进行处理。

[0063] S406、将所述临时流表表项修改为正式流表表项,并删除所述临时流表表项。

[0064] 所述正式流表表项的建立方法很多,在此不作专门描述。

[0065] 可以理解的是,在具体实施中,也可以在对临时流表表项中的数据包统计值进行修改后,再确定具体对数据包进行如何处理。例如,如果所匹配的流表表项中的数据包数统计值未超过预设阈值,则按照匹配的临时流表表项中的第一流控制信息对数据包进行处理,通常只对数据包进行转发或丢弃;而如果所匹配的流表表项中的数据包数统计值超过预设阈值时,也可以将所述匹配的临时流表表项转为正式流表表项后,再按照转换后的正式流表表项中的第二流控制信息对数据包进行处理,通常对数据包进行转发、流量复制、带宽限制或丢弃。

[0066] 可以理解的是,所述统计值还可以是数据包的字节数统计值,且也可以设置数据包的字节数阈值,当所述数据包的字节数超过字节数阈值时将临时流表转为正式流表。

[0067] 本实施例中,由于临时流表中只是包含二次哈希值信息、统计值和简单的流控制信息,不保存 Key 值,因此可以节约存储空间,由于对数据包进行处理时,不需要将数据包中的 Key 值与正式流表中的 Key 值进行匹配,因此可以减少 CPU 资源消耗,因此不会出现现有技术中由于资源不足,在对流表加速老化时,无法区分具体的业务类型而对业务造成影响的情况。而通过设置数据包数统计值阈值,并在数据包数统计值超过所述阈值时,将临时流表表项转为正式流表表项,可以为包数很多的稳定连接提供较好的服务质量。综上,本实施例中数据包处理所采用的流表,其中的临时流表可以有效应对当前互联网中出现的包数很少的短连接非常多的情况,包括部分 P2P 应用以及扫描和 DoS 攻击流量。而正式流表则可以为包数很多的稳定连接提供有效的服务,保证业务服务质量。总之,上述数据包处理方法可以在兼顾数据包转发质量的情况下降低 CPU 以及内存等资源的利用率,提高资源利用效率。因此本发明实施例可以有效应对当前互联网中大量包数很少的连接,尤其是部分 P2P 应用以及扫描和 DoS 攻击流量,避免对这些连接采用正式流表进行处理而消耗大量的 CPU 及内存资源。

[0068] 在具体应用中,可以对上述方案作进一步优化,以下通过几个具体实施例进行说明:

[0069] 参照图 5,本发明实施例三基于流表的数据包处理方法流程图,与前述两个实施例的不同之处在于,在没有匹配的临时流表表项时,如果存在正式流表表项,将所述数据包中的 Key 值与正式流表表项中的 Key 值进行匹配,该方法包括:

[0070] S501、对数据包进行两次哈希,分别得到一次哈希值和二次哈希值;

[0071] S502、查找所述一次哈希值对应的哈希单元;

[0072] S503、在找到的哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项,如果找到与所述二次哈希值匹配的临时流表表项,则执行 S504;如果未找到与所述二次哈希值匹配的临时流表表项,则执行 S507;

[0073] S504、按照匹配的临时流表表项中的第一流控制信息对数据包进行处理并修改所

述匹配的临时流表表项中的数据包数统计值；

[0074] 第一流控制信息通常具有指示对数据包进行转发或者直接删除的操作。

[0075] 临时流表表项中具体包含的表项内容可以参照实施例一和实施例二，此处不再赘述。

[0076] S505、判断所述数据包数统计值是否超过预设阈值，如果否，则结束流程；如果是，则执行 S506；

[0077] 具体的，可以设置临时流表转为正式流表的数据包数统计值阈值，例如 3 个数据包，一旦一个数据流的数据包计数达到该阈值，则执行 S506，即将所述临时流表表项转为正式流表表项。

[0078] S506、将所述临时流表修改为正式流表表项，并删除所述临时流表表项；

[0079] S507、查找是否有正式流表表项，如果是，则执行 S508；否则，执行 S509；

[0080] S508、按照与所述数据包中的 Key 值匹配的正式流表表项中的第二流控制信息对数据包进行处理，并修改所述匹配的正式流表表项中的流状态信息；

[0081] 正式流表表项内容通常包括 Key 值、流状态信息以及流控制信息等。其中流控制信息包含一些对数据包或数据包所在流设置的操作，例如对接收的数据包进行转发或者丢弃，也可以在转发前进行流量复制，或者对某个数据包所在流进行带宽限制等。

[0082] S509、按照系统默认配置或策略对数据包进行处理，同时新建与该哈希单元对应的临时流表表项。

[0083] 具体建立与所述哈希单元对应的临时流表表项的过程可以如图 6 所示，参照图 6，在具体应用中，如果所述数据包在正式流表和临时流表中都没有匹配的表项时，则可在前述实施例基础上还包括按照系统默认设置或策略对所述数据包进行处理，并创建与所述哈希单元对应的临时流表表项。具体创建临时流表表项的方法可以包括：

[0084] S601、判断是否有空闲的临时流表表项，如果有，则执行 S602；如果没有，则执行 S603；

[0085] S602、建立新的临时流表表项，设置数据包数统计值为 1，按照所述新建临时流表表项中的流控制信息对数据包进行处理；

[0086] S603、选择数据包数最少的临时流表表项或无数据包时间最长的临时流表表项进行覆盖，建立新的临时流表表项，设置数据包数统计值为 1，并按照所述新建临时流表表项中的流控制信息对数据包进行处理。

[0087] 可以理解的是，如果实际统计值为字节数，则设置字节数统计值为所述数据包中的字节数。在这种情况下，如果不存在临时流表表项，则选择字节数最少的临时流表表项或者无数据包（字节数为零）时间最长的临时流表表项进行覆盖，建立新的临时流表表项。

[0088] 从本发明实施例可以看出，在没有匹配的临时流表表项时，如果存在正式流表表项，则按照与所述数据包中的 Key 值匹配的正式流表表项对数据包进行处理，可以兼容基于现有流表的数据包处理方法，利于推广应用。当既不存在与该数据包匹配的临时流表表项，也不存在与该数据包匹配的正式流表表项时，按照系统默认配置或策略对数据包进行处理，同时，新建与该哈希单元对应的临时流表表项，以对该数据流的后续数据包进行处理。

[0089] 以上对本发明实施例中流表的数据包转发方法进行了详细说明，以下对所述方法

所涉及的装置及系统进行对应描述：

[0090] 图 7 是本发明实施例五所示的数据包处理装置结构示意图,该装置中预置有流表,所述流表包括哈希单元和临时流表,所述临时流表中包含二次哈希值信息和第一流控制信息,所述装置包括：

[0091] 哈希处理单元 71,用于对数据包中的 Key 值进行两次哈希,得到一次哈希值和二次哈希值；

[0092] 第一查找单元 72,用于查找所述一次哈希值对应的哈希单元；

[0093] 第二查找单元 73,用于在第一查找单元 72 中找到的哈希单元所对应的临时流表中查找与所述二次哈希值匹配的临时流表表项；

[0094] 第一处理单元 74,用于根据与所述二次哈希值匹配的临时流表表项中的第一流控制信息对数据包进行处理。

[0095] 从本实施例可以看出,由于临时流表中不保存 Key 值,临时流表规格较正式流表小很多,因此可以节约存储空间,且在匹配时只需要将二次哈希值进行匹配,而不需要与流表表项中的 Key 值进行一一匹配,因此可以提高 CPU 性能。综上可知,该装置对 CPU 以及内存等资源的具有较高的利用效率。该装置尤其适用于当前互联网上,数据包数很少的短连接非常多的情况,例如部分 P2P 应用以及扫描和 DoS 攻击流量的情况。

[0096] 图 8 是本发明实施例六数据包处理装置结构示意图,在装置在实施例五基础上进行扩展,与实施例五中所述装置的不同之处在于,流表还包括正式流表,所述正式流表的表项内容包括 Key 值信息和第二流控制信息,所述装置在实施例五基础上还可包括：

[0097] 第一判断单元 81,用于在所述第二查找单元 73 未找到与所述二次哈希值匹配的临时流表表项时,判断是否存在与所述哈希单元对应的正式流表；

[0098] 第二处理单元 82,用于当存在与所述哈希单元对应的正式流表时,根据与所述数据包中的 Key 值完全匹配的正式流表表项中的第二流控制信息对所述数据包进行处理；当不存在与所述哈希单元对应的正式流表时,按照系统默认配置或策略对数据包进行处理。

[0099] 可以看出,本装置不但可以提高 CPU 以及内存资源的利用效率,还可与现有技术中预置有正式流表的装置兼容,利于推广应用。

[0100] 图 9 是本发明实施例七数据包处理装置结构示意图,在实施例六所示装置基础上扩展得到,与实施例六所示装置的不同之处在于,临时流表的表项内容还包括统计值,该装置还可包括：

[0101] 第一流表维护单元 91,用于当第二查找单元 73 查找到与所述二次哈希值匹配的临时流表表项时,修改所述匹配的临时流表表项的统计值；当所述统计值超过预设阈值时,将所述匹配的临时流表表项修改为正式流表表项,并删除所述临时流表表项。

[0102] 第二流表维护单元 92,用于在不存在与所述哈希单元对应的正式流表时,创建与所述哈希单元对应的临时流表表项。

[0103] 可以理解的是,在具体实施中,第一流表维护单元 91 和第二流表维护单元 92 可以集成在一起,作为流表维护单元对流表进行维护。

[0104] 可以看出,本装置通过对流表进行维护,使其完全兼容于应用现有正式流表的装置,利于推广应用。且由于临时流表中只是包含二次哈希值信息、统计值和简单的流控制信息,不保存 Key 值,因此可以节约存储空间；由于对数据包进行处理时,不需要将数据包中

的 Key 值与正式流表中的 Key 值进行匹配,因此可以减少 CPU 资源消耗,因此不会出现现有技术中由于资源不足在对流表加速老化时,无法区分具体的业务类型而对业务造成影响的情况。而通过设置数据包数统计值阈值,并在数据包数统计值超过所述阈值时,将临时流表表项转为正式流表表项,可以为包数很多的稳定连接提供较好的服务质量。本装置尤其可以有效应对当前互联网中大量包数很少的连接,尤其是部分 P2P 应用以及扫描和 DoS 攻击流量,避免对这些连接采用正式流表进行处理而消耗大量的 CPU 及内存资源。

[0105] 本发明实施例中还可提供一种数据处理系统,该数据处理系统包括数据包接收装置和上述实施例中所说的数据包处理装置,所述数据包接收装置用于接收数据包,所述数据包处理装置用于对数据接收装置所接收到的数据包进行控制处理,这里不再一一举例说明。

[0106] 上述实施例所介绍的数据处理系统包括但不限于路由器、交换机、防火墙等。

[0107] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程,是可以通过计算机程序来指令相关的硬件来完成,所述的程序可存储于计算机可读取存储介质中,该程序在执行时,可包括如上述各方法的实施例的流程。其中,所述的存储介质可为磁碟、光盘、只读存储记忆体 (Read-Only Memory, ROM) 或随机存储记忆体 (Random Access Memory, RAM) 等。

[0108] 以上对本发明实施例所提供的一种基于流表的数据包转发方法和设备进行了详细介绍,本文中应用了具体个例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上可以加以改变,而所有这些改变或替换都应属于本发明所附的权利要求的保护范围。综上所述,本说明书内容不应理解为对本发明的限制。

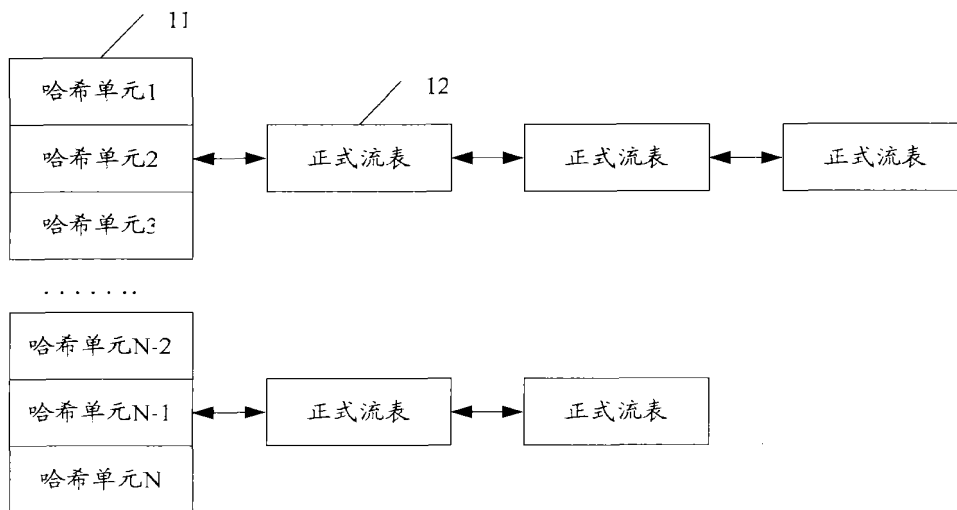


图 1

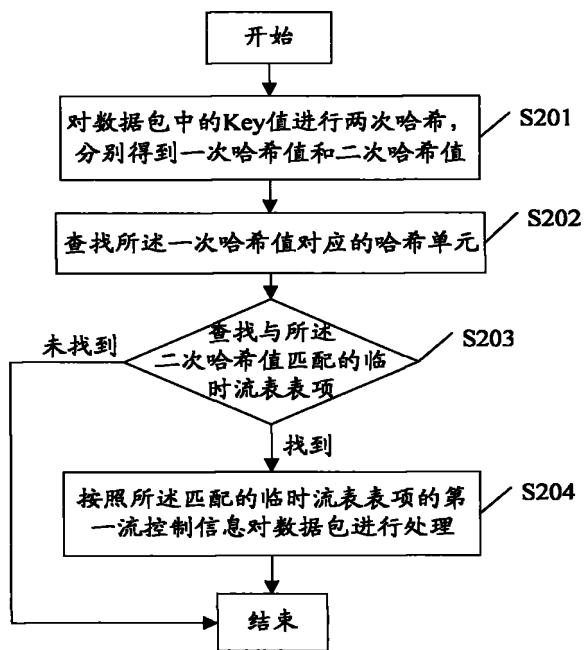


图 2

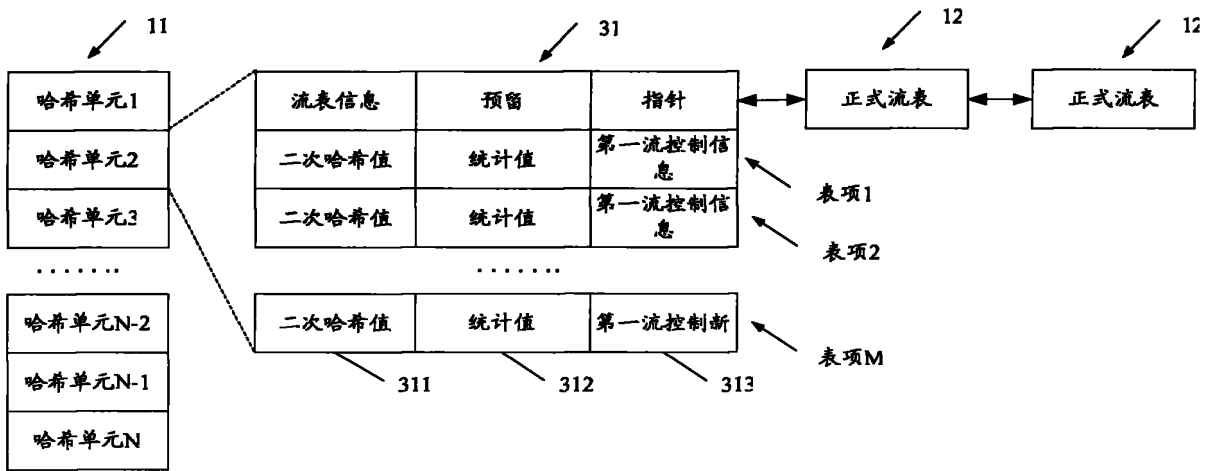


图 3

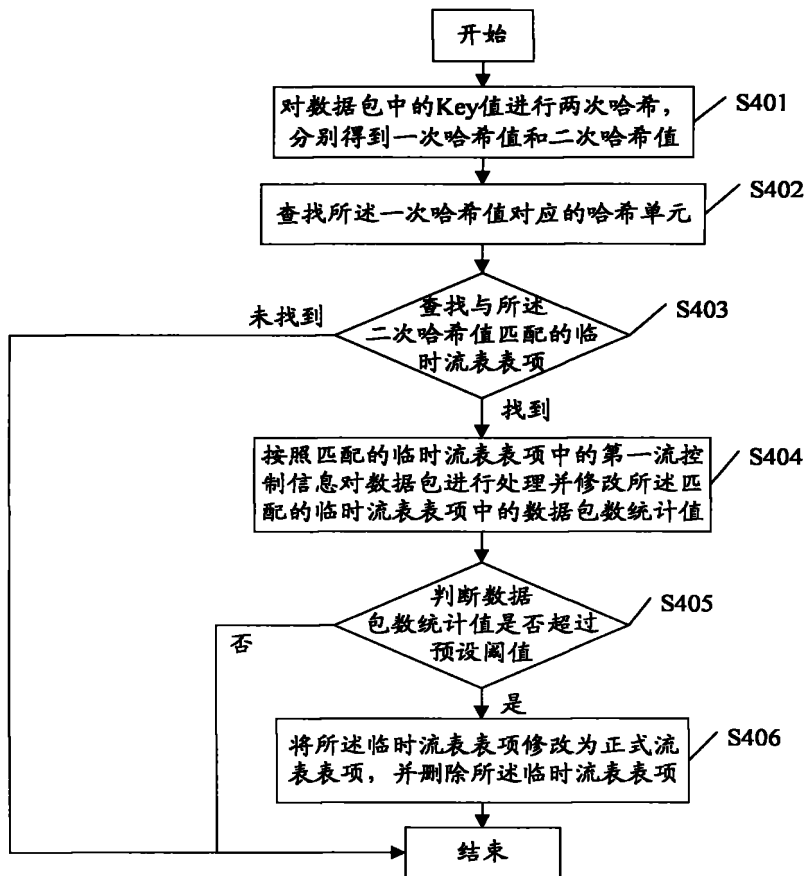


图 4

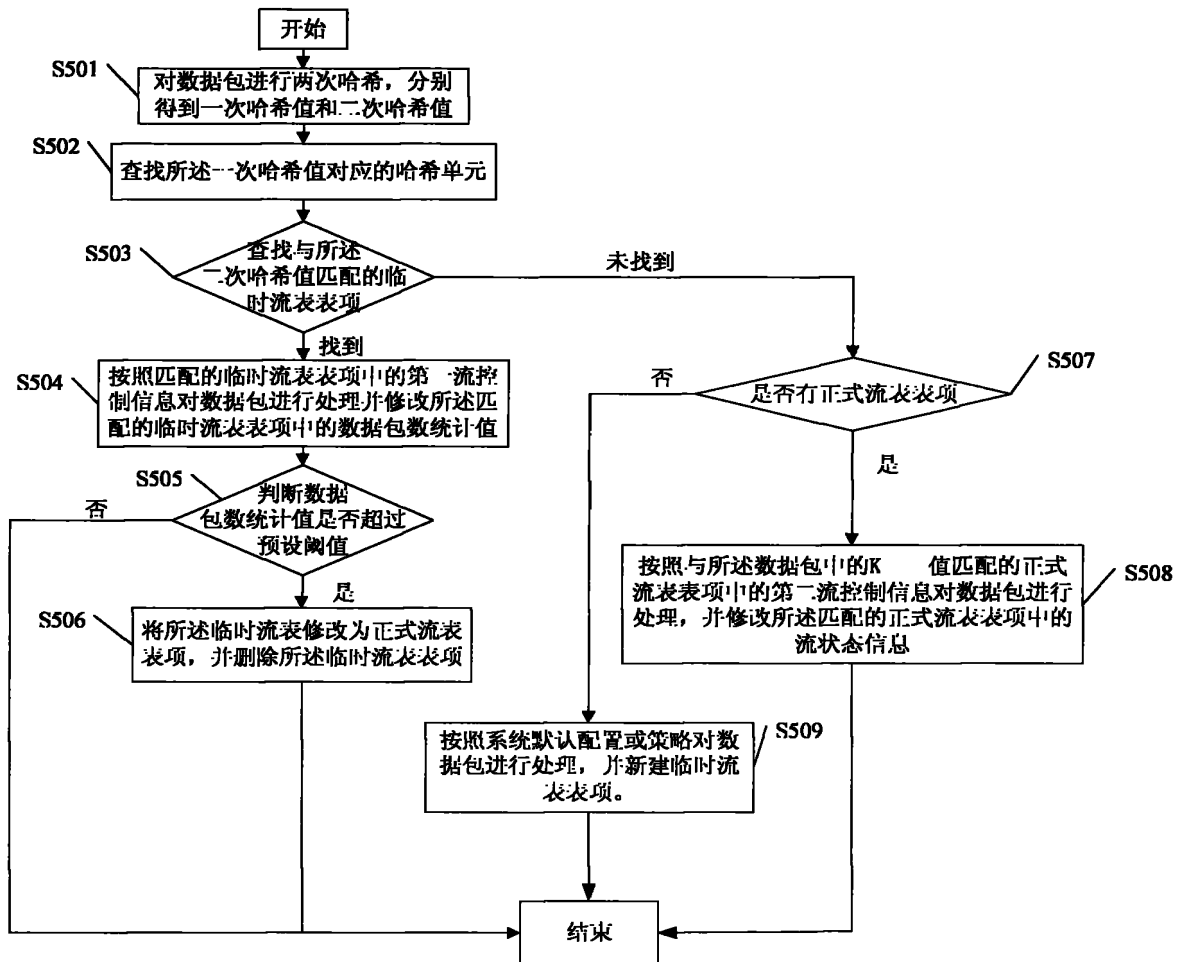


图 5

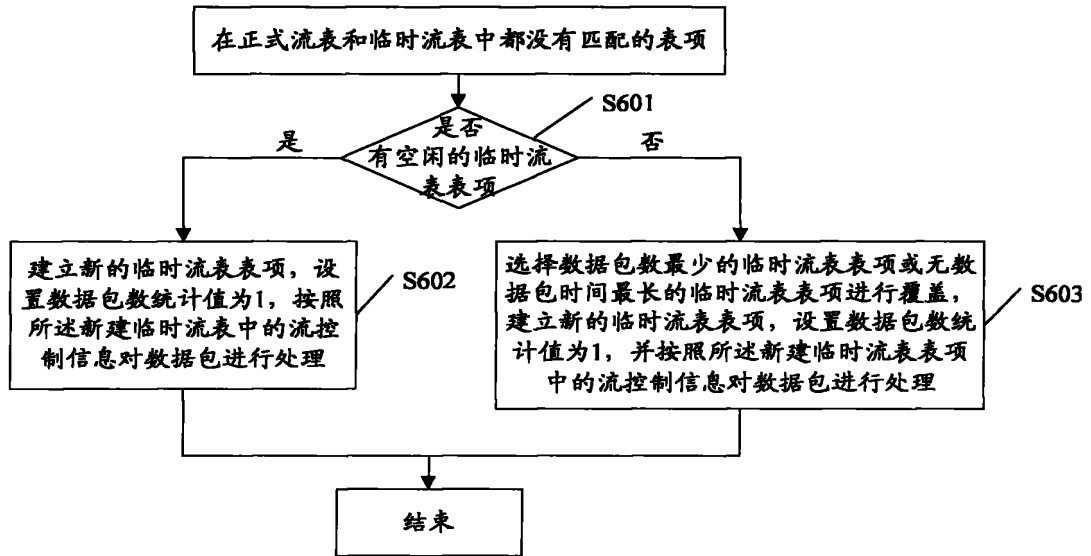


图 6

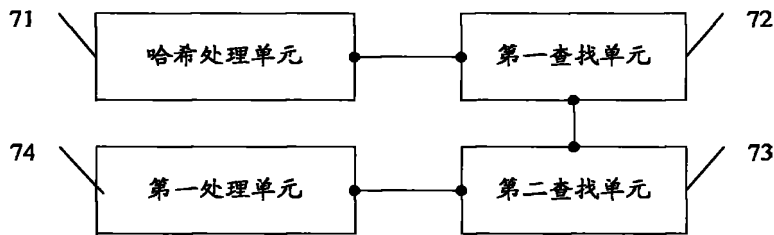


图 7

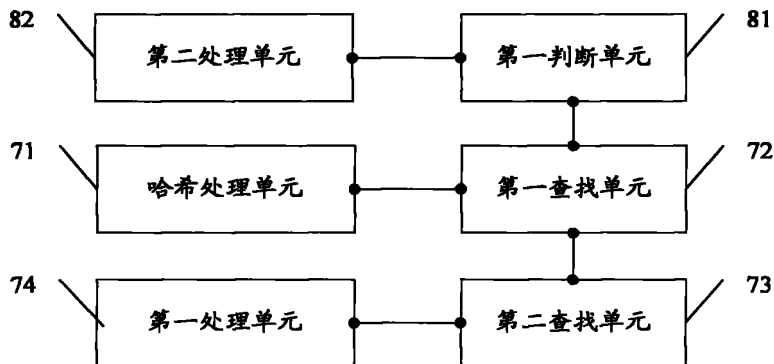


图 8

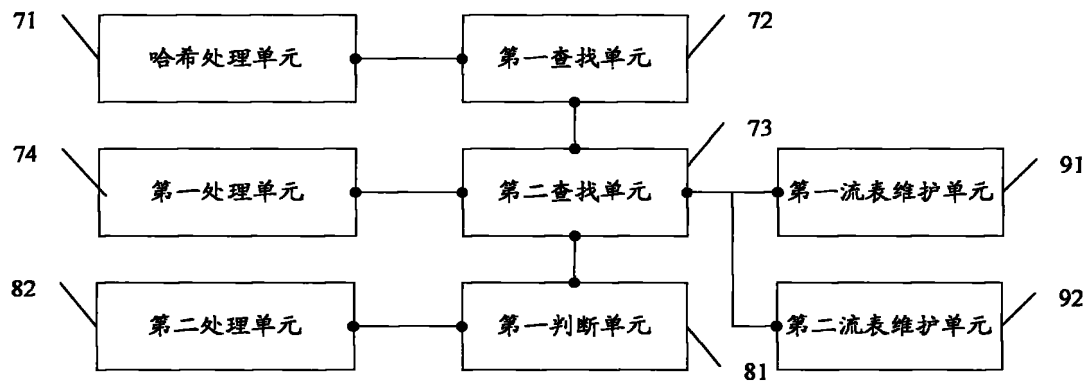


图 9

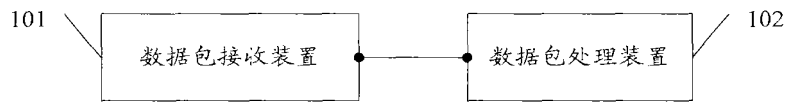


图 10