



(12) 发明专利

(10) 授权公告号 CN 110417609 B

(45) 授权公告日 2021.02.09

(21) 申请号 201810387680.5

G06F 9/48 (2006.01)

(22) 申请日 2018.04.26

审查员 牛爽

(65) 同一申请的已公布的文献号

申请公布号 CN 110417609 A

(43) 申请公布日 2019.11.05

(73) 专利权人 中移(苏州)软件技术有限公司

地址 215163 江苏省苏州市高新区科灵路  
78号

专利权人 中国移动通信集团有限公司

(72) 发明人 何丹

(74) 专利代理机构 北京同达信恒知识产权代理

有限公司 11291

代理人 郭润湘

(51) Int. Cl.

H04L 12/26 (2006.01)

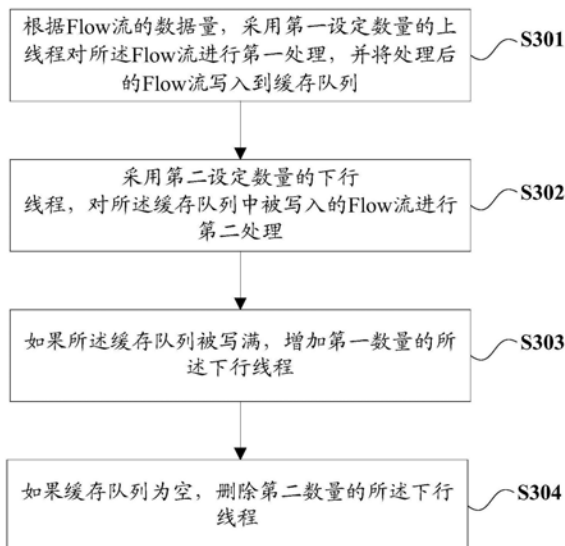
权利要求书3页 说明书16页 附图4页

(54) 发明名称

一种网络流量的统计方法、装置、电子设备  
及存储介质

(57) 摘要

本发明提供了一种网络流量的统计方法、装置、电子设备及存储介质,该方法包括:根据Flow流的数据量,采用第一设定数量的上行线程对该Flow流进行第一处理,并将处理后Flow流写入到缓存队列;采用第二设定数量的下行线程,对缓存队列中被写入的Flow流进行第二处理,每一个上行线程与每一个下行线程的处理能力相同;如果缓存队列被写满,增加第一数量的下行线程;如果缓存队列为空,删除第二数量下行线程。在本发明中,采用线程组对Flow流进行同步处理,并通过判断缓存队列是否被写满或者为空,增加或者减少下行线程的数量,对线程组的处理能力进行动态调控,克服了现有技术中各处理步骤之间处理能力固化、缺乏协同配合的问题。



1. 一种网络流量的统计方法,其特征在于,所述方法包括:

根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;

采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;

如果所述缓存队列被写满,增加第一数量的所述下行线程;

如果所述缓存队列为空,删除第二数量的所述下行线程;

其中,所述上行线程为解析线程,所述下行线程为归并线程;

当所述上行线程为解析线程,所述下行线程为归并线程时,所述方法还包括:

获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;

根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;

根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

2. 如权利要求1所述的方法,其特征在于,所述增加第一数量的所述下行线程之后,所述方法还包括:

将所述缓存队列的缓存队列长度增加第一预设单位长度;

所述删除第二数量的所述下行线程之后,所述方法还包括:

将所述缓存队列的缓存队列长度减小第二预设单位长度。

3. 如权利要求1所述的方法,其特征在于,所述删除第二数量的所述下行线程包括:

根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

4. 如权利要求1所述的方法,其特征在于,所述方法还包括:

根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;

根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

5. 一种电子设备,其特征在于,所述电子设备包括:第一处理器和第一存储器;

所述第一处理器,用于读取所述第一存储器中的程序,执行下列过程:根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;如果所述缓存队列被写满,增加第一数量的所述下行线程;如果所述缓存队列为空,删除第二数量的所述下行线程;

其中,所述上行线程为解析线程,所述下行线程为归并线程;

所述第一处理器,还用于当所述上行线程为解析线程,所述下行线程为归并线程时,获

取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

6.如权利要求5所述的电子设备,其特征在于,增加第一数量的所述下行线程之后,所述第一处理器还用于将所述缓存队列的缓存队列长度增加第一预设单位长度;删除第二数量的所述下行线程之后,所述第一处理器还用于将所述缓存队列的缓存队列长度减小第二预设单位长度。

7.如权利要求5所述的电子设备,其特征在于,所述第一处理器删除第二数量的所述下行线程时,具体用于根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

8.如权利要求5所述的电子设备,其特征在于,所述第一处理器,还用于根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

9.一种网络流量的统计装置,其特征在于,所述装置包括:

第一处理模块,用于根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;

第二处理模块,用于采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;

第一增加模块,用于如果所述缓存队列被写满,增加第一数量的所述上行线程;

第二删除模块,用于如果所述缓存队列为空,删除第二数量的所述下行线程;

其中,所述上行线程为解析线程,所述下行线程为归并线程;

当所述上行线程为解析线程,所述下行线程为归并线程时;

所述第一处理模块,用于获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;

所述第二处理模块,用于根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

10.一种电子设备,其特征在于,所述电子设备包括:第二处理器、通信接口、第二存储器和通信总线,其中,第二处理器,通信接口,第二存储器通过通信总线完成相互间的通信;

所述第二存储器中存储有计算机程序,当所述程序被所述第二处理器执行时,使得所

述第二处理器执行权利要求1-4任一项所述方法的步骤。

11.一种计算机可读存储介质,其特征在于,其存储有可由电子设备执行的计算机程序,当所述程序在所述电子设备上运行时,使得所述电子设备执行权利要求1-4任一项所述方法的步骤。

## 一种网络流量的统计方法、装置、电子设备及存储介质

### 技术领域

[0001] 本发明涉及网络技术领域,尤其涉及一种网络流量的统计方法、装置、电子设备及存储介质。

### 背景技术

[0002] 交换机的NetFlow技术是一种数据交换方式,其工作原理是:NetFlow利用标准的交换模式处理数据流的第一个IP包数据,生成NetFlow缓存,随后同样的数据基于缓存信息在同一个数据流中进行传输,不再匹配相关的访问控制等策略,NetFlow缓存同时包含了随后数据流的统计信息。因此,交换机的NetFlow技术可以实现对高速转发的IP数据流进行采集。

[0003]

源IP	目的IP	源端口	网络名	协议类型	大小	默认
10.12.112.67	10.10.242.24	64421	Others	17	6844200	1
211.241.112.64	10.10.242.24	64751	As10	17	1459000	1
10.10.242.16	10.10.242.24	64741	Others	17	4501	1

[0004] 表1

[0005] NetFlow的技术框架为:交换机对内部的流量按照配置频率采集一个个Flow流,其中,一个Flow流定义为在一个源IP地址和目的IP地址间传输的单向数据包流,将此Flow流通过用户数据协议UDP包进行打包发送到收集装置,由收集装置收集Flow流,并进行解析,得到每一个Flow流的会话级视图,该会话级视图包括网络流量的七元组,即源IP、目的IP、端口号等数据,再由归并装置对上述网络流量的会话级视图进行归并计算,得到如表1所示的网络流量的会话级视图,以实现网络信息监测和统计。其中,现有技术进行流量统计的流程如图1中所示。

[0006] 由于业内的现有技术对Flow流的接收和解析都被简单地整合到收集步骤中,因此收集交换机采集的Flow流的时候,往往忽视了交换机发送Flow流的发送能力和收集装置对Flow流的接收能力及解析能力的同步匹配问题,导致比如当Flow流被大规模的发送到收集装置,而此时收集装置没有达到解析此Flow流的能力,而导致收集装置来不及接收新的Flow流,最终导致交换机采集的Flow流被丢弃,从而影响整体的统计精度。同样相反的,若交换机发送Flow流的频率缓慢,收集装置的收集能力被设计很大,则会造成收集装置一边收集Flow流,一边等待收集,从而导致收集装置计算能力的相对空置和浪费。

[0007] 另外,业内的现有技术对网络流量的会话级视图进行归并计算的过程如图2所示,即先获取每个Flow流的会话级视图,对每个Flow流的会话级视图进行归并计算,得到表1中的会话级视图,然后在表1中的会话级视图进行分离,得到以IP为主体的流量统计,如流量数据等,最后根据统计周期,得到统计周期内的流量以及流速等数据。例如,对统计表1中源IP地址10.12.112.67的IP地址进行归并计算时,需要先从表1中得到10.12.112.67的流量数据及统计周期,然后根据流量数据及统计周期,即可计算出10.12.112.67在统计周期内的流速,而流量的统计则为每个统计周期内流量数据的累积。由于在归并计算过程中,是

分两步对流量及流速进行统计,因此会造成得到的局域网内的网络流量统计与流速统计相对滞后。

[0008] 因此,现有技术中在进行流量统计时存在因各处理步骤之间处理能力固化,而导致的各处理步骤之间的配合效率低、缺乏协同配合,以至于对流量和流速的监测统计不精确,流量统计与流速统计相对滞后的问题。

### 发明内容

[0009] 本发明提供一种网络流量的统计方法、装置、电子设备及存储介质,用以解决现有技术中在进行流量统计时存在的各处理步骤之间处理能力固化、配合效率低、缺乏协同配合的问题。

[0010] 为了解决上述问题,依据本发明的一个方面,提供了一种网络流量的统计方法,所述方法包括:

[0011] 根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;

[0012] 采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;

[0013] 如果所述缓存队列被写满,增加第一数量的所述下行线程;

[0014] 如果所述缓存队列为空,删除第二数量的所述下行线程。

[0015] 通过上述的描述可知,在本发明中,采用线程组的方式对Flow流进行处理,即上行线程对Flow流进行第一处理之后,会将处理后的Flow流先写入到缓存队列中,然后再由下行线程对写入到缓存队列中的Flow流进行第二处理。其中,由于上行线程与下行线程同时进行,因此可实现上行线程与下行线程之间的高效协同。另由于每一个所述上行线程与每一个所述下行线程的处理能力相同,因此通过增加或者减少下行线程的数量,即可对下行线程的处理能力进行调整。其中在对下行线程的处理能力进行调整时,可通过判断缓存队列是否被写满或者为空,相应的增加或者减少下行线程的数量,以实现根据上行线程对下行线程处理能力的要求,对下行线程的处理能力进行自动化调整,保证下行线程与上行线程在处理能力上相匹配,并最终实现各线程之间动态的处理能力以匹配各线程之间的高效协同,克服现有技术中各处理步骤之间处理能力固化、配合效率低、缺乏协同配合的问题。

[0016] 具体地,所述上行线程为接收线程,下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程。

[0017] 即当上行线程为接收线程,下行线程为解析线程时,通过判断缓存队列是否被写满或者为空,相应的增加或者减少解析线程的数量,以实现解析线程的解析能力进行调整。当所述上行线程为解析线程,所述下行线程为归并线程,通过判断缓存队列是否被写满或者为空,相应的增加或者减少归并线程的数量,以实现归并线程的归并计算能力进行调整,从而保证整个线程组在处理能力上相匹配。

[0018] 具体地,所述增加第一数量的所述下行线程之后,所述方法还包括:

[0019] 将所述缓存队列的缓存队列长度增加第一预设单位长度;

[0020] 所述删除第二数量的所述下行线程之后,所述方法还包括:

[0021] 将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0022] 由上述描述可知,在增加或者减少下行线程的数量时,需要判断缓存队列是否被写满或者为空,若缓存队列的缓存队列长度太小,则会对下行线程的处理能力造成局限,若缓存队列的缓存队列长度太大,则不会对下行线程的处理能力起到良好的调控作用,所以缓存队列的缓存队列长度可在一定程度上决定下行线程处理能力的大小。因此通过增加或者减小缓存队列的缓存对列长度,可优化对线程组的处理能力进行动态调控的效果,进一步保证下行线程与上行线程在处理能力上相匹配,更为高效的协同配合完成流量或流速的统计工作。

[0023] 具体地,所述删除第二数量的所述下行线程包括:

[0024] 根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0025] 为了便于根据上行线程对下行线程处理能力的需求,对下行线程的数量进行相应的增加或者减少,可根据下行线程的启动时间进行排序,当缓存队列被写满,需要删除下行线程时,可先从启动时间较晚的下行线程开始删除,以免造成整个线程组的混乱。

[0026] 具体地,当所述上行线程为解析线程,所述下行线程为归并线程时,所述方法还包括:

[0027] 获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;

[0028] 根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;

[0029] 根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0030] 通过采用哈希函数对源IP地址进行归并计算,因此可以对每一个源IP地址的流量数据进行统计,进而结合预设的统计周期,可以计算出每一个源IP地址的流速,提高了统计的精确度。

[0031] 具体地,所述方法还包括:

[0032] 根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;

[0033] 根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0034] 通过采用哈希函数对第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,可以对相同源IP地址、目的IP地、源端口、目的端口以及协议类型的流量,在预设的统计周期内进行统计和更新,以获取统计级别的统计数据,保证了流量统计的精确性。

[0035] 其中,由于采用哈希函数对源IP地址进行归并计算,与采用哈希函数对第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算是同时进行的,因此通过提供上述的双Hash归并统计机制,实现了同步计算会话级统计和以IP为主体的流量流速统计,不但提高了对流量和流速的统计精度,而且与现有技术相比,不会拆分

两步对流量及流速进行统计,所以还解决了现有技术中对流量和流速的统计相对滞后的问题。

[0036] 依据本发明的另一个方面,还提供了一种电子设备,包括:第一处理器和第一存储器;

[0037] 所述第一处理器,用于读取所述第一存储器中的程序,执行下列过程:根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;如果所述缓存队列被写满,增加第一数量的所述下行线程;如果所述缓存队列为空,删除第二数量的所述下行线程。

[0038] 具体地,所述上行线程为接收线程,下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程。

[0039] 具体地,增加第一数量的所述下行线程之后,所述第一处理器还用于将所述缓存队列的缓存队列长度增加第一预设单位长度;删除第二数量的所述下行线程之后,所述第一处理器还用于将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0040] 具体地,所述第一处理器删除第二数量的所述下行线程时,具体用于根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0041] 具体地,所述第一处理器,还用于当所述上行线程为解析线程,所述下行线程为归并线程时,获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0042] 具体地,所述第一处理器,还用于根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0043] 依据本发明的另一方面,还提供了一种网络流量的统计装置,所述装置包括:

[0044] 第一处理模块,用于根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;

[0045] 第二处理模块,用于采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;

[0046] 第一增加模块,用于如果所述缓存队列被写满,增加第一数量的所述下行线程;

[0047] 第一删除模块,用于如果所述缓存队列为空,删除第二数量的所述下行线程。

[0048] 依据本发明的另一个方面,还提供了一种电子设备,所述电子设备包括:第二处理



器、通信接口、第二存储器和通信总线,其中,第二处理器,通信接口,第二存储器通过通信总线完成相互间的通信;

[0049] 所述第二存储器中存储有计算机程序,当所述程序被所述第二处理器执行时,使得所述第二处理器执行上述网络流量的统计方法的步骤。

[0050] 依据本发明的另一个方面,还提供了一种计算机可读存储介质,其存储有可由电子设备执行的计算机程序,当所述程序在所述电子设备上运行时,使得所述电子设备执行上述网络流量的统计方法的步骤。

[0051] 本发明的有益效果:

[0052] 本发明提供了一种网络流量的统计方法、装置、电子设备及存储介质,该方法包括:根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;如果所述缓存队列被写满,增加第一数量的所述下行线程;如果所述缓存队列为空,删除第二数量的所述下行线程。由上述可知,在本发明中,采用线程组的方式对Flow流进行处理,即上行线程对Flow流进行第一处理之后,会将处理后的Flow流先写入到缓存队列中,然后再由下行线程对写入到缓存队列中的Flow流进行第二处理。其中,上行线程与下行线程同时进行,因此可实现上行线程与下行线程之间的高效协同,提高上行线程与下行线程之间的配合效率。由于每一个上行线程与每一个下行线程的处理能力相同,因此通过相应的增加或者减少下行线程的数量,即可对下行线程的处理能力进行调整。其中,当缓存队列被写满,则表示上行线程的能力大于下行线程的能力,需要增加下行线程的数量;而当缓存队列为空时,则表示上行线程的能力小于下行线程的能力,需要减少下行线程的数量。因此可实现根据上行线程对下行线程处理能力的需求,对下行线程的处理能力进行动态调控,并最终实现各线程之间动态的处理能力以匹配各线程之间的高效协同,解决了现有技术中各处理步骤之间处理能力固化,各处理步骤之间的配合效率低、缺乏协同配合的问题。

## 附图说明

[0053] 图1为现有技术中网络流量统计的过程示意图;

[0054] 图2为现有技术中归并计算的过程示意图;

[0055] 图3为本发明实施例中网络流量的统计方法的流程示意图;

[0056] 图4为本发明实施例中上行线程与下行线程数据交互过程的示意图;

[0057] 图5为本发明实施例中哈希算法的流程图;

[0058] 图6为本发明实施例中电子设备的示意图之一;

[0059] 图7为本发明实施例中网络流量的统计装置的结构框图;

[0060] 图8为本发明实施例中网络流量的统计的整体流程图;

[0061] 图9为本发明实施例中电子设备的示意图之二。

## 具体实施方式

[0062] 为了使本发明的目的、技术方案和优点更加清楚,下面将结合附图对本发明作进一步地详细描述,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

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它实施例，都属于本发明保护的范围。

[0063] 实施例一

[0064] 由于现有技术中，在进行流量与流速的统计时，由于各个步骤之间的处理能力被固化，各处理步骤之间的配合效率低、缺乏协同配合，因此在统计过程中，常常会出现对Flow流来不及解析的丢包情况，或者一边等待接收，一边解析，导致资源浪费的情况。

[0065] 因此，为解决上述技术问题，本发明的实施例提供了一种网络流量的统计方法，其中，需要说明的是，本发明实施例提供的网络流量的统计方法应用于一种电子设备上。图3为本发明实施例中提供的网络流量的统计方法的流程图，如图3所示，该方法包括以下步骤：

[0066] 步骤S301：根据Flow流的数据量，采用第一设定数量的上行线程对所述Flow流进行第一处理，并将处理后Flow流写入到缓存队列；

[0067] 步骤S302、采用第二设定数量的下行线程，对所述缓存队列中被写入的Flow流进行第二处理，其中，每一个所述上行线程与每一个所述下行线程的处理能力相同。

[0068] 由上述可知，在本发明实施中，采用线程组的方式对Flow流进行处理，其中，在上行线程对Flow流进行第一处理之后，会先将处理的Flow流写入到缓存队列中，而在下行线程在对缓存队列中写入的Flow流进行第二处理时，需要从缓存队列中获取Flow流，因此上行线程和下行线程之间，并不直接进行数据交互。由于缓存队列可对处理后Flow流进行存储，因此可在一定程度上避免Flow流的丢包情况，提高流量统计的精确度。另由于上行线程包括多个线程，且下行线程也包括多个线程，所以上行线程与下行线程之间必定存在同时进行的线程，因此可保证上行线程与下行线程同时进行，以实现上行线程与下行线程之间的高效协同，提高上行线程与下行线程之间的配合效率。

[0069] 步骤S303：如果所述缓存队列被写满，增加第一数量的所述下行线程。

[0070] 由于每一个所述上行线程与每一个所述下行线程的处理能力相同，且每一个Flow流在缓存队列中占用一个单位长度，因此如果缓存队列被写满，则说明此时上行线程的处理能力大于下行线程的处理能力，需要增加并启动第一数量的下行线程，以保证下行线程的处理能力与上行线程的处理能力相匹配。同时还可以防止Flow流从缓存队列中溢出，出现丢包现象，而导致对流量的统计不精准。

[0071] 步骤S304：如果所述缓存队列为空，删除第二数量的所述下行线程。

[0072] 同样的，由于每一个所述上行线程与每一个所述下行线程的处理能力相同，Flow流在缓存队列中占用一个单位长度，如果缓存队列为空，则说明此时上行线程的处理能力小于下行线程的处理能力，需要关闭并删除第二数量的下行线程，以保证下行线程的处理能力与上行线程的处理能力相匹配。同时还可以避免出现一边等待接收，一边解析，导致资源浪费的情况。

[0073] 因此，在本发明实施例中，可根据上行线程对下行线程处理能力的的需求，对下行线程的数量进行相应的增加或者减少，以实现下行线程组的处理能力进行动态调控，解决了现有技术中各处理步骤之间处理能力固化的问题。同时还通过各线程之间动态的处理能力以匹配各线程之间的高效协同，解决了现有技术中各处理步骤之间的配合效率低、缺乏协同配合的问题。

[0074] 其中,需要说明的是,第一数量与第二数量可以相同,也可以不同,而第一数量与第二数量的具体数值,可根据实际情况进行设定,在本发明实施例中,对比并不进行具体限定。

[0075] 其中,由于所述上行线程为接收线程,下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程,为使本发明实施例阐述更加清晰,则以上行线程为接收线程,下行线程为解析线程为例,并结合图4进行具体说明。

[0076] 当上行线程为接收线程时,接收线程的数量是根据待接收Flow流的数据量进行设定的,因此,不会出现Flow流来不及接收的情况,在一定程度上保证了对流量或流速进行统计的精确性。其中,在本发明实施例中,接收线程主要采用recvfrom()函数不断的循环监听获取交换机发送的Flow流,其中,recvfrom()函数用于从已连接的套接口上接收数据,并捕获发送源的地址。在本发明实施例中,套接口即为交换机的输出口。

[0077] 当下行线程为解析线程时,则解析线程需要从缓存队列中获取已写入的Flow流,并对该Flow流进行解析。由于,每一个接收线程与每一个解析线程的处理能力相同,且每个线程可提供一个单位的处理能力,因此线程数量的多少即可代表该线程处理能力的大小。因此通过判断缓存队列是否为写满或者为空,即可判断出当前解析线程的解析能力与接收线程的接收能力是否匹配。其中,当缓存队列被写满时,表示当前解析线程的解析能力小于接收线程的接收能力,需要相应的增加解析线程的数量;当缓存队列为空时,表示当前解析线程的解析能力大于接收线程的接收能力,需要相应的减少解析线程的数量,以保证接收线程的处理能力与解析线程的处理能力相匹配。

[0078] 另由于解析线程与接收线程并不直接进行信息交互,而是将接收到的Flow流写入到缓存队列中,且解析线程的处理能力可根据接收线程的需求进行自动调整,因此可保证被写入到缓存队列中的Flow流,不会因解析线程的处理能力较小,来不及对其进行解析而发生溢出现象,导致统计数据不精确,也不会因解析线程的处理能力较小,而发生解析资源的浪费。

[0079] 具体地,图4为本发明实施例中上行线程与下行线程数据交互过程的示意图,当上行线程为接收线程,下行线程为解析线程时,则接收线程即为图4中的生产者,不断地向缓存列中塞入从交换机抓取的Flow流,而解析线程即为图4中的消费者,不断地从缓存队列中提取Flow流。其中,需要说明的是,在本发明实施例中,每一个Flow流在缓存队列中占用一个单位长度,且接收线程与解析线程之间共享缓存队列中的缓存队列长度,因此当接收线程从交换机抓取到Flow流,并放入到缓存队列中时,缓存队列的剩余缓存长度则会减小,而当解析线程从缓存队列中提取Flow流,并进行解析时,缓存队列的剩余缓存长度则会增加,所以缓存队列中的剩余缓存长度是一个动态变化过程。因此,通过判断缓存队列是否被写满或者为空,即可判断出解析线程的处理能力与接收线程的处理能力是否匹配,从而可实现对解析线程组的处理能力进行动态调控,以克服现有技术中接收步骤与解析步骤之间处理能力固化、缺乏协同配合的问题。

[0080] 同样的,当上行线程为解析线程,下行线程为归并线程时,也适用上述的调控方法和图4中数据交互的过程,以实现归并线程的处理能力进行调控。即解析线程即为图4中的生产者,不断地向缓存列中塞入对Flow流进行解析后生成的数据,而归并线程即为图4中的消费者,不断地从缓存队列中提取对Flow流进行解析后生成的数据。在该过程中,缓

存队列中的剩余缓存长度也是一个动态变化过程。因此,通过判断缓存队列是否为写满或者为空,也能够判断出归并线程的处理能力与解析线程的处理能力是否匹配,并相应的增加或者减少归并线程的数量,以达到归并线程的处理能力与解析线程的处理能力相匹配的目的,以克服现有技术中归并步骤与解析步骤之间处理能力固化的问题。

[0081] 由上述可知,在本发明实施例中,对Flow流进行接收、解析以及归并的过程中,可通过下行线程感应上行线程对下行线程处理能力的需求,达到下行线程的处理能力与上行线程的处理能力相匹配的目的,并通过启动上行线程和下行线程之间的同步机制,以实现各步骤之间的高效同步配合,最终通过各线程之间动态的处理能力以匹配各线程之间的高效协同,高效地完成流量和流速的统计任务,克服了现有技术中因各处理步骤之间处理能力固化,而导致的各处理步骤之间的配合效率低、缺乏协同配合的问题。

[0082] 实施例二

[0083] 由上述实施例可知,由于在增加或者减少下行线程的数量时,需要判断缓存队列是否被写满或者为空,而若缓存队列的缓存队列长度太小,则会对上行线程与下行线程处理能力造成一定的局限,而若缓存队列的缓存队列长度太大,则不会对下行线程的处理能力起到良好的调控作用,因此缓存队列的缓存队列长度可在一定程度上决定下行线程处理能力的大小。

[0084] 为优化对线程组的处理能力进行动态调控的效果,进一步保证下行线程与上行线程在处理能力上相匹配,在本发明实施例中,增加第一数量的所述下行线程之后,所述方法还包括:将所述缓存队列的缓存队列长度增加第一预设单位长度;删除第二数量的所述下行线程之后,所述方法还包括:将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0085] 其中,需要说明的是,在本发明实施例中,每一个Flow流在缓存队列中占用一个单位长度,且缓存队列的缓存队列长度的阈值范围为0至N。在判断缓存队列是否被写满或者为空时,需要在上行线程将Flow流写入到缓存队列后,或者下行线程从缓存队列中提取Flow流后,判断当前的已占用的缓存队列长度是否在阈值范围0至N内。若当前的已占用的缓存队列长度达到最大值N,则说明缓存队列被写满;若当前的已占用的缓存队列长度为最小值0,则说明缓存队列为空。

[0086] 由于增加下行线程的数量后,已经增加了下行线程的处理能力,此时,若缓存队列长度增加第一预设单位长度,且第一预设单位长度为1个单位长度,那么此时缓存队列的缓存队列长度的阈值范围变更为0至(N+1),所以只有当前的已占用的缓存队列长度达到(N+1)时,缓存队列才能被写满,因此相当于进一步增加了下行线程的处理能力。

[0087] 而由于减少下行线程的数量后,已经减小了下行线程的处理能力,此时,若缓存队列长度减小第二预设单位长度,且第二预设单位长度为1个单位长度,那么此时缓存队列的缓存队列长度的阈值范围变更为0至(N-1),所以只要当前的已占用的缓存队列长度达到(N-1)时,缓存队列就已经被写满,因此相当于进一步减小了下行线程的处理能力。

[0088] 因此,在本发明实施例中,通过相应的增加或者减小缓存队列的缓存队列长度,可进一步保证下行线程和上行线程在处理能力上相匹配,以实现通过高效的协同配合完成流量和流速的统计任务。

[0089] 其中,需要说明的是,第一预设单位长度与第二预设单位长度可以相同,也可以不同,而第一预设单位长度与第二预设单位长度的具体数值,可根据实际情况进行设定,在本

发明实施例中,对比并不进行具体限定。

#### [0090] 实施例三

[0091] 由于每一个下行线程的启动时间不同,因此在删除第二数量的所述下行线程包括:

[0092] 根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0093] 其中,通过根据每一条下行线程的启动时间对每一条下行线程进行排序,可便于对下行线程进行管理,当缓存队列被写满,需要删除第二数量的下行线程时,可以先从启动时间较晚的下行线程开始删除,以免造成整个线程组的混乱。同样的,当增加第一数量的下行线程时,则将最新启动的下行线程作为启动时间较晚的下行线程。当然可以理解的是,在本发明实施例中,并不仅仅限定于上述的管理方式。

[0094] 即在本发明实施例中,根据每一条下行线程的启动时间,分别将每一个下行线程的顺序标记为1、2、3...i,若上行线程的处理能力大于下行线程的处理能力,则按下行线程的顺序1、2、3...i,由1至i依次启动并增加;若上行线程的处理能力小于下行线程的处理能力,则按下行线程的顺序i、i-1...1,由i至1依次关闭并删除。

#### [0095] 实施例四

[0096] 在上述实施例的基础上,当所述上行线程为解析线程,所述下行线程为归并线程时,即采用第一设定数量的解析线程对Flow流进行解析之后,生成Flow流的第一会话级视图,并将Flow流的第一会话级视图写入到缓存队列中,归并线程需从缓存队列中获取每一个Flow流对应的第一会话级视图,并对每一个Flow流对应的第一会话级视图进行归并统计。其中,与每一个Flow流对应的第一会话级视图在缓存队列中占用一个单位长度。

[0097] 其中,归并线程在对每一个Flow流对应的第一会话级视图进行归并统计时,具体通过哈希函数进行归并计算。由于第一会话级视图包括网络流量的七元组,即源IP、目的IP、端口号等数据,因此哈希函数可将第一会话级视图中至少一个数据元素的关键字作为自变量key,并对该数据元素进行归并计算,并通过一定的函数关系,计算出对应的数值,该数值即为该数据元素在哈希表中的对应的地址。

[0098] 具体地,在本发明实施例中,为实现对每一个源IP地址的流量和流速进行统计,所述方法包括:

[0099] 获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;

[0100] 根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;

[0101] 该函数的含义为,用key可取的数值,乘以预设质数7,然后除以第一预设表长后得到的余数,即为该数值在哈希表中的对应的存储地址。其中预设质数并不限定为整数7,需要理解的是,在本发明实施例中,对预设质数的取值并不进行具体限定。

[0102] 根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0103] 由上述可知,在本发明实施例中,是对源IP地址进行归并计算,则需要将源IP地址

作为自变量key,而源IP地址为如10.12.112.67的形式,因此在根据上述哈希函数进行计算时,需先对源IP地址的形式进行变更,比如将10.12.112.67中的点去掉,变更为数值101211267,当然这仅为本发明实施例中所列举的其中一种对源IP地址的变更方式,其他可得到与源IP地址对应的数值的方式,也在本发明实施例的保护范围内。在对源IP地址的形式进行变更后得到的数值,根据上述哈希函数,先乘以预设质数7,然后除以第一预设表长后得到的余数,即为该源IP地址在第一哈希表中对应的存储地址,在该存储地址对应的位置可存储该源IP地址的流量数据,然后根据预设周期,计算出该源IP地址的流速,并对源IP地址在每一个所述预设统计周期内的流量数据进行更新,以实现对该源IP地址的流量和流速的统计。

[0104] 上述为对一个源IP地址进行归并计算的过程,而对多个源IP地址进行归并计算时,其流程图如图5所示。

[0105] 具体地,在解析得到第一会话级视图后,对该第一会话级视图中的第一源IP地址,按上述归并计算的过程,确定出在第一源IP地址在第一哈希表中对应的存储地址,即通过上述哈希函数计算得到的第一余数。而在解析得到第二会话级视图后,对该第二会话级视图中的第二源IP地址,按上述归并计算的过程,通过上述哈希函数计算得到的第二余数,此时,需要判断第一余数与第二余数是否相同,即图5中所示的第一哈希表内按地址扫描,如果第一余数与第二余数不同,即地址无匹配,则新增地址,该新增的地址即为第二源IP地址在第一哈希表中对应的存储地址,而如果第一余数与第二余数相同,即地址有匹配,则需判断第一源IP地址与第二源IP地址中包括的数据信息是否相同,即第一哈希表内链接第一源IP地址中包括的数据信息,按七元组中包括的数据信息进行扫描,如果数据信息不同,即数据信息无匹配,则需对第二余数采用二次探测在哈希的方式,直到第二源IP地址在第一哈希表中与第一源IP地址在第一哈希表中对应的存储地址不同,新增地址,该新增的地址即为第二源IP地址在第一哈希表中对应的存储地址,而如果数据信息相同,即数据信息匹配,则对存储地址对应的位置中存储的第一源IP地址的流量数据进行更新,并根据预设周期,计算出该第一源IP地址的流速,以实现该源IP地址的流量和流速的统计。

[0106] 通过本发明实施例可知,本发明实施例中针对每一个源IP地址的流量和流速进行统计时,由于在整个过程中,不会出现丢包现象,且与现有技术中从会话级视图中分离得到每一源IP地址的流量后,在计算流速的方法相比,是通过上述哈希函数对每一个源IP地址的流量的统计,因此获取到统计结果更加精确,从而计算获取的周期内的流速也更加精确。

[0107] 实施例五

[0108] 依据上述实施例可知,对Flow流进行解析之后生成第一会话级视图包括网络流量的七元组,即源IP、目的IP、端口号等数据,为实现对第一会话级视图中包括的流量数据进行统计,所述方法还包括:

[0109] 根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数。

[0110] 其中,本发明实施例中,哈希函数与上述实施例中哈希函数的含义相同,故在本发明实施例中,不在赘述。

[0111] 根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会

话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0112] 也是就是说,本发明实施例是针对第一会话级视图中源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,并对源IP地址、目的IP地址、源端口、目的端口以及协议类型按上述列举的变更方式,得到源IP地址、目的IP地址、源端口、目的端口以及协议类型进行变更后相加得到的数值,即 $key = flow(ip1) + flow(ip2) + flow(s1) + flow(s2) + flow(xy)$ ,其中,ip1为源IP地址,ip2为目标IP地址,s1为源端口,s2为目标端口,xy为协议类型,根据上述哈希函数,确定第一会话级视图在第二哈希表中对应的存储地址,并在存储该地址对应的位置存储第一会话级视图中的数据信息,然后再根据预设周期,对第一会话级视图中的流量进行更新,以实现每个Flow流进行统计级别的统计数据。

[0113] 其中,对获取的每个Flow流解析得到的第一会话级视图进行归并计算的流程,与上述实施例中图5所示的对多个源IP地址进行归并计算的流程相同,故在本发明实施例中,不在赘述。

[0114] 由于本发明实施例中对第一会话级视图进行归并计算的流程,与上述实施例中对源IP地址进行归并计算的流程同步进行,因此可同时生成关于对每一个源IP地址的流量和流速进行统计的第一哈希表,以及关于对每一个第一会话级视图中的流量数据进行统计的第二哈希表,分别同时实现对第一会话级视图进行统计级别的流量统计和以源IP地址为主体的流量和流速的统计。因此通过提供上述的双Hash归并统计机制,实现了同步计算会话级统计和以IP为主体的流量流速统计,与现有技术中先对第一会话级视图进行归并计算,并在归并计算的结果中得到以源IP地址为主体的流量和流速的统计方法相比,不但提高了对流量和流速的统计精度,而且不会拆分两步对流量及流速进行统计,因此还解决了现有技术中对流量和流速的统计相对滞后的问题。

[0115] 实施例六

[0116] 依据本发明实施例的另一个方面,还提供了一种电子设备,图6为本发明实施例提供的一种电子设备,所述电子设备包括:第一处理器61和第一存储器62。

[0117] 在图6中,总线架构可以包括任意数量的互联的总线和桥,具体由第一处理器61代表的一个或多个第一处理器61和第一存储器62代表的各种电路链接在一起。总线架构还可以将诸如外围设备、稳压器和功率管理电路等之类的各种其他电路链接在一起,这些都是本领域所公知的,因此,本文不再对其进行进一步描述。第一处理器61负责管理总线架构和通常的处理,第一存储器62可以存储第一处理器61在执行操作时所使用的数据。

[0118] 可选的,第一处理器61可以是CPU(中央处理器)、ASIC(Application Specific Integrated Circuit,专用集成电路)、FPGA(Field-Programmable Gate Array,现场可编程门阵列)或CPLD(Complex Programmable Logic Device,复杂可编程逻辑器件)。

[0119] 其中,所述第一处理器61,用于读取所述第一存储器62中的程序,执行下列过程:根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;如果所述缓存队列被写满,增加第一数量的所述下行线程;如果所述缓存队列为空,删除第二数量的所述下行线程。

[0120] 具体地,所述上行线程为接收线程,下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程。

[0121] 具体地,增加第一数量的所述下行线程之后,所述第一处理器61还用于将所述缓存队列的缓存队列长度增加第一预设单位长度;删除第二数量的所述下行线程之后,所述第一处理器61还用于将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0122] 具体地,所述第一处理器61删除第二数量的所述下行线程时,具体用于根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0123] 具体地,所述第一处理器61,还用于当所述上行线程为解析线程,所述下行线程为归并线程时,获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0124] 具体地,所述第一处理器61,还用于根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0125] 实施例七

[0126] 依据本发明实施例的另一个方面,还提供一种网络流量的统计装置,图7为本发明实施例提供的一种网络流量的统计装置的结构框图,所述装置包括:

[0127] 第一处理模块701,用于根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;

[0128] 第二处理模块702,用于采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处理能力相同;

[0129] 第一增加模块703,用于如果所述缓存队列被写满,增加第一数量的所述下行线程;

[0130] 第二删除模块704,用于如果所述缓存队列为空,删除第二数量的所述下行线程。

[0131] 具体地,所述上行线程为接收线程,所述下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程。

[0132] 如图7所示,所述装置还包括:

[0133] 第二增加模块705,用于增加第一数量的所述下行线程之后,将所述缓存队列的缓存队列长度增加第一预设单位长度;

[0134] 第二删除模块706,用于删除第二数量的所述下行线程之后,将所述缓存队列的缓存队列长度减小第二预设单位长度。



[0135] 具体地,第二删除模块704具体用于,根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0136] 具体地,当所述上行线程为解析线程,所述下行线程为归并线程时,

[0137] 第一处理模块701用于,获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;

[0138] 第二处理模块702用于,根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0139] 具体地,当所述上行线程为解析线程,所述下行线程为归并线程时,

[0140] 第二处理模块702还用于,根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7 \bmod (\text{len}))$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0141] 其中,图8为本发明实施例中进行网络流量的统计的整体流程图,当上行线程为接收线程,下行线程为解析线程时,则第一处理模块701和第二处理模块702分别对应图8中的接收模块和解析模块,当所述上行线程为解析线程,所述下行线程为归并线程时,则第一处理模块701和第二处理模块702分别对应图8中的解析模块和归并模块。

[0142] 其中,接收模块采用接收线程组,接收交换机发送的Flow流,并将接收到的Flow流写入到缓存队列;解析模块采用解析线程组从缓存队列中获取Flow流,并进行解析;归并模块采用归并线程组,对解析Flow流后生成的第一会话级视图进行归并统计,并采用哈希函数实现同时对第一会话级视图进行统计级别的流量统计和以源IP地址为主体的流量和流速的统计。其中,P1V1表示一个中间的信号量,以实现上行线程对下行线程的协同控制。即该信号量需要与0或N进行大小比较,当该信号量为0时,需要减小下行线程的数量,当该信号量为N时,需要增加下行线程的数量,以实现对下行线程的处理能力的大小进行自适应调整,以保证上行线程和下行线程在处理能力上的匹配,并以高效的协同配合完成流量和流速的统计任务。

[0143] 实施例八

[0144] 依据本发明的另一个方面,还提供了一种电子设备,图9为本发明实施例提供的一种电子设备,所述电子设备包括:第二处理器91、通信接口92、第二存储器93和通信总线94,其中,第二处理器91,通信接口92,第二存储器93通过通信总线94完成相互间的通信;

[0145] 所述第二存储器93中存储有计算机程序,当所述程序被所述第二处理器91执行时,使得所述第二处理器91执行下列过程:

[0146] 根据Flow流的数据量,采用第一设定数量的上行线程对所述Flow流进行第一处理,并将处理后的Flow流写入到缓存队列;采用第二设定数量的下行线程,对所述缓存队列中被写入的Flow流进行第二处理,其中,每一个所述上行线程与每一个所述下行线程的处

理能力相同;如果所述缓存队列被写满,增加第一数量的所述下行线程;如果所述缓存队列为空,删除第二数量的所述下行线程。

[0147] 具体地,所述上行线程为接收线程,下行线程为解析线程;和/或所述上行线程为解析线程,所述下行线程为归并线程。

[0148] 具体地,增加第一数量的所述下行线程之后,所述第二处理器91还用于将所述缓存队列的缓存队列长度增加第一预设单位长度;删除第二数量的所述下行线程之后,所述第二处理器91还用于将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0149] 具体地,所述第二处理器91删除第二数量的所述下行线程时,具体用于根据每个下行线程的启动时间,确定启动时间较晚的第二数量的目标下行线程,删除所述目标下行线程。

[0150] 具体地,所述第二处理器91,还用于当所述上行线程为解析线程,所述下行线程为归并线程时,获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图;根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址进行归并计算,生成关于所述源IP地址的第一哈希表,其中key是哈希函数的自变量,len为第一预设表长,7为预设质数;根据预设统计周期以及所述第一哈希表,计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速,并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0151] 具体地,所述第二处理器91,还用于根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ,对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算,生成关于所述第一会话级视图的第二哈希表,其中key是哈希函数的自变量,len为第二预设表长,7为预设质数;根据预设统计周期以及所述第二哈希表,对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0152] 上述各实施例中的电子设备提到的通信总线94可以是外设部件互连标准(Peripheral Component Interconnect, PCI)总线或扩展工业标准结构(Extended Industry Standard Architecture, EISA)总线等。该通信总线94可以分为地址总线、数据总线、控制总线等。为便于表示,图中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

[0153] 通信接口92,用于上述电子设备与其他设备之间的通信。

[0154] 第二存储器93可以包括随机存取存储器(Random Access Memory, RAM),也可以包括非易失性存储器(Non-Volatile Memory, NVM),例如至少一个磁盘存储器。可选地,第二存储器93还可以是至少一个位于远离前述处理器的存储装置。

[0155] 上述第二处理器91可以是通用处理器,包括中央处理器、网络处理器(Network Processor, NP)等;还可以是数字信号处理器(Digital Signal Processing, DSP)、专用集成电路、现场可编程门阵列或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。

[0156] 实施例九

[0157] 依据本发明的另一个方面,还提供了一种计算机可读存储介质,其存储有可由电子设备执行的计算机程序,当所述程序在所述电子设备上运行时,使得所述电子设备执行

以下过程：

[0158] 根据Flow流的数据量，采用第一设定数量的上行线程对所述Flow流进行第一处理，并将处理后的Flow流写入到缓存队列；采用第二设定数量的下行线程，对所述缓存队列中被写入的Flow流进行第二处理，其中，每一个所述上行线程与每一个所述下行线程的处理能力相同；如果所述缓存队列被写满，增加第一数量的所述下行线程；如果所述缓存队列为空，删除第二数量的所述下行线程。

[0159] 具体地，所述上行线程为接收线程，所述下行线程为解析线程；和/或所述上行线程为解析线程，所述下行线程为归并线程。

[0160] 具体地，所述增加第一数量的所述下行线程之后，所述电子设备还用于将所述缓存队列的缓存队列长度增加第一预设单位长度；所述删除第二数量的所述下行线程之后，所述电子设备还用于将所述缓存队列的缓存队列长度减小第二预设单位长度。

[0161] 具体地，所述删除第二数量的所述下行线程时，所述电子设备还用于，根据每个下行线程的启动时间，确定启动时间较晚的第二数量的目标下行线程，删除所述目标下行线程。

[0162] 具体地，当所述上行线程为解析线程，所述下行线程为归并线程时，所述电子设备还用于：

[0163] 获取所述解析线程对Flow流进行解析之后生成的所述Flow流的第一会话级视图；

[0164] 根据哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ，对所述第一会话级视图中的源IP地址进行归并计算，生成关于所述源IP地址的第一哈希表，其中key是哈希函数的自变量，len为第一预设表长，7为预设质数；

[0165] 根据预设统计周期以及所述第一哈希表，计算出所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流速，并对所述第一哈希表中每一个所述源IP地址在每一个所述预设统计周期内的流量数据进行更新。

[0166] 具体地，当所述上行线程为解析线程，所述下行线程为归并线程时，所述电子设备还用于：

[0167] 根据所述哈希函数 $\text{Hash}(\text{key}) = (\text{key} * 7) \bmod (\text{len})$ ，对所述第一会话级视图中的源IP地址、目的IP地址、源端口、目的端口以及协议类型进行归并计算，生成关于所述第一会话级视图的第二哈希表，其中key是哈希函数的自变量，len为第二预设表长，7为预设质数；

[0168] 根据预设统计周期以及所述第二哈希表，对所述第二哈希表中每一个所述第一会话级视图在每一个所述预设统计周期内的流量数据进行更新。

[0169] 上述实施例中的计算机可读存储介质可以是电子设备中的处理器能够存取的任何可用介质或数据存储设备，包括但不限于磁性存储器如软盘、硬盘、磁带、磁光盘(MO)等、光学存储器如CD、DVD、BD、HVD等、以及半导体存储器如ROM、EPROM、EEPROM、非易失性存储器(NAND FLASH)、固态硬盘(SSD)等。

[0170] 尽管已描述了本发明的优选实施例，但本领域内的技术人员一旦得知了基本创造性概念，则可对这些实施例作出另外的变更和修改。所以，所附权利要求意欲解释为包括优选实施例以及落入本发明范围的所有变更和修改。

[0171] 显然，本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样，倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围

之内,则本发明也意图包含这些改动和变型在内。

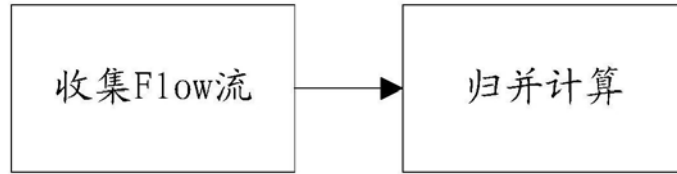


图1

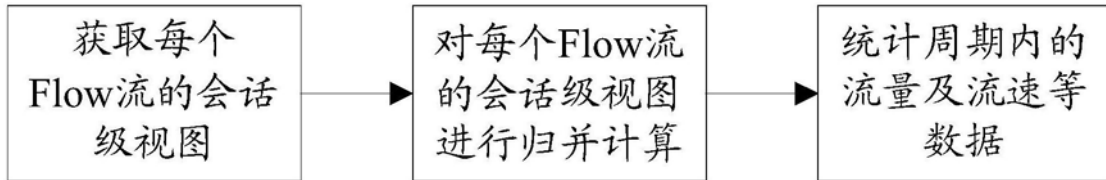


图2

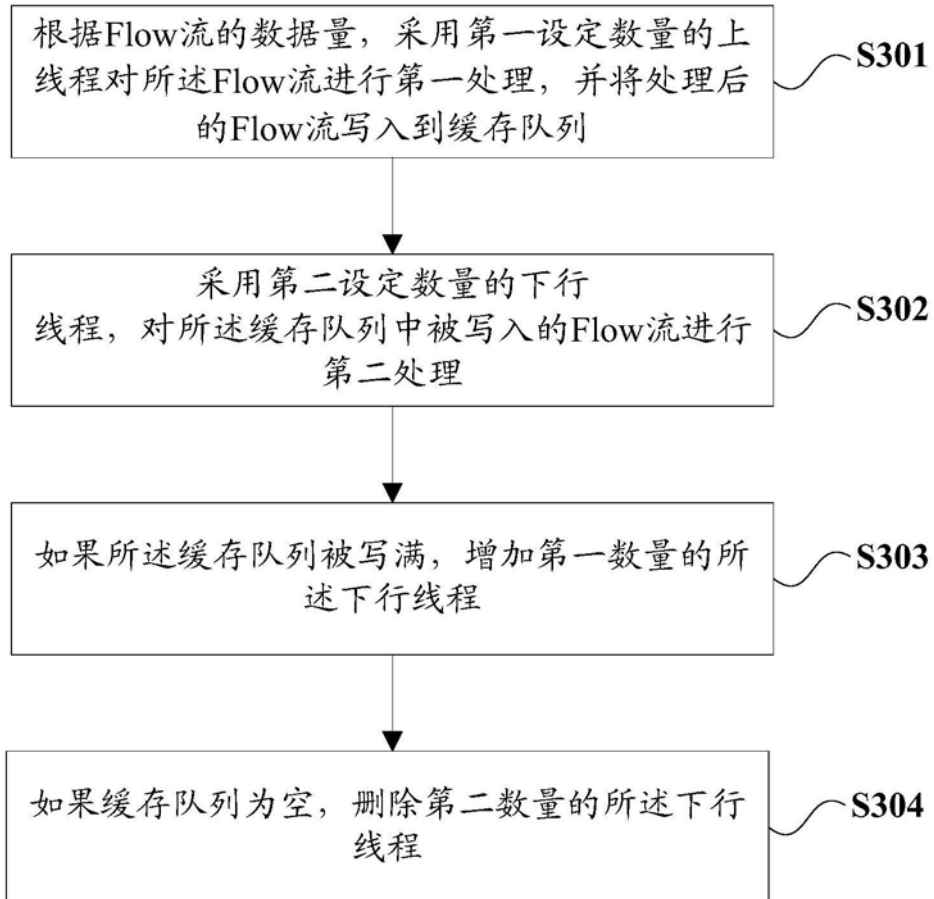


图3

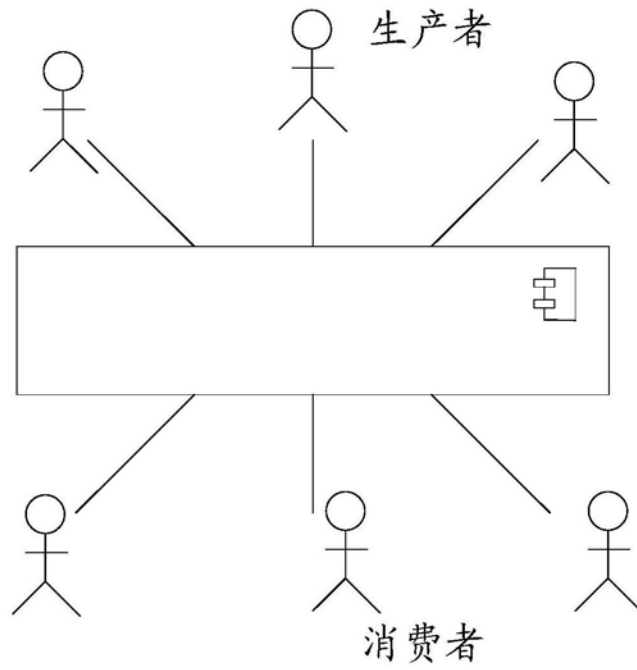


图4

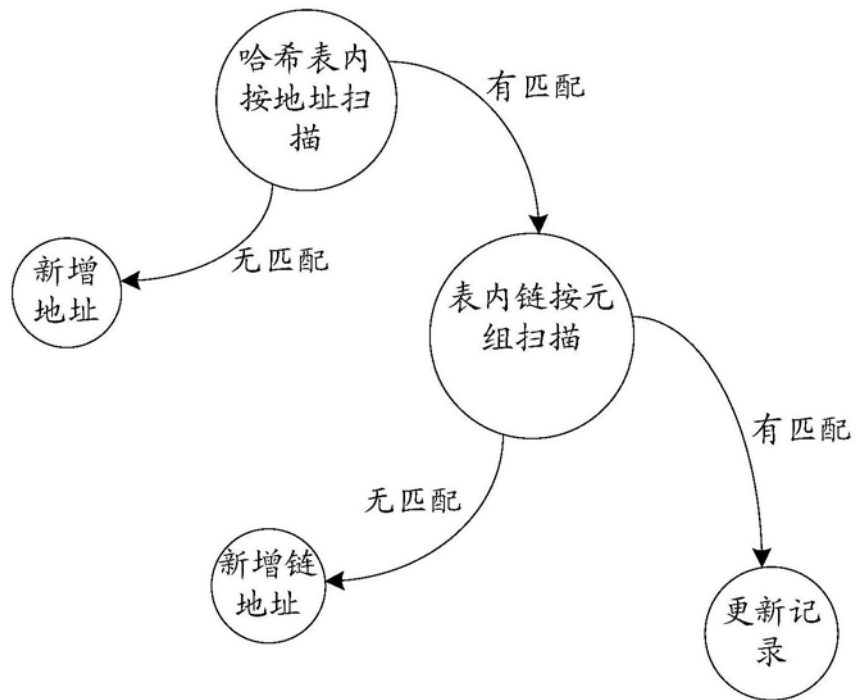


图5

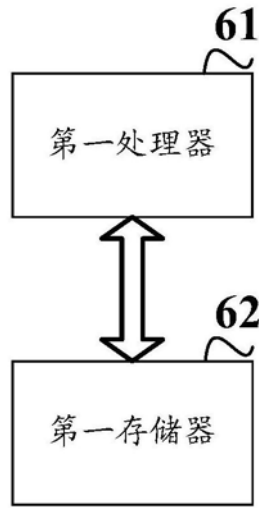


图6

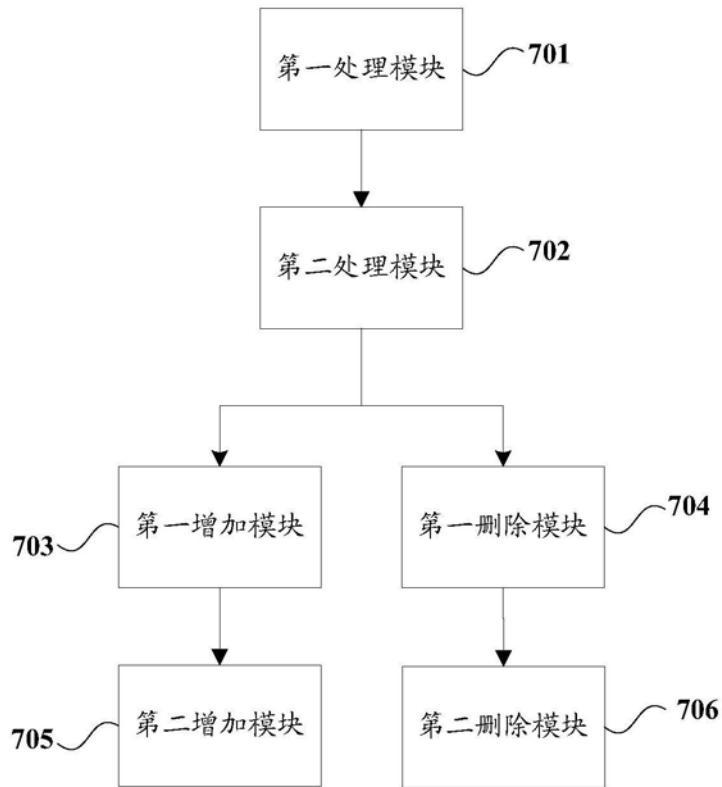


图7

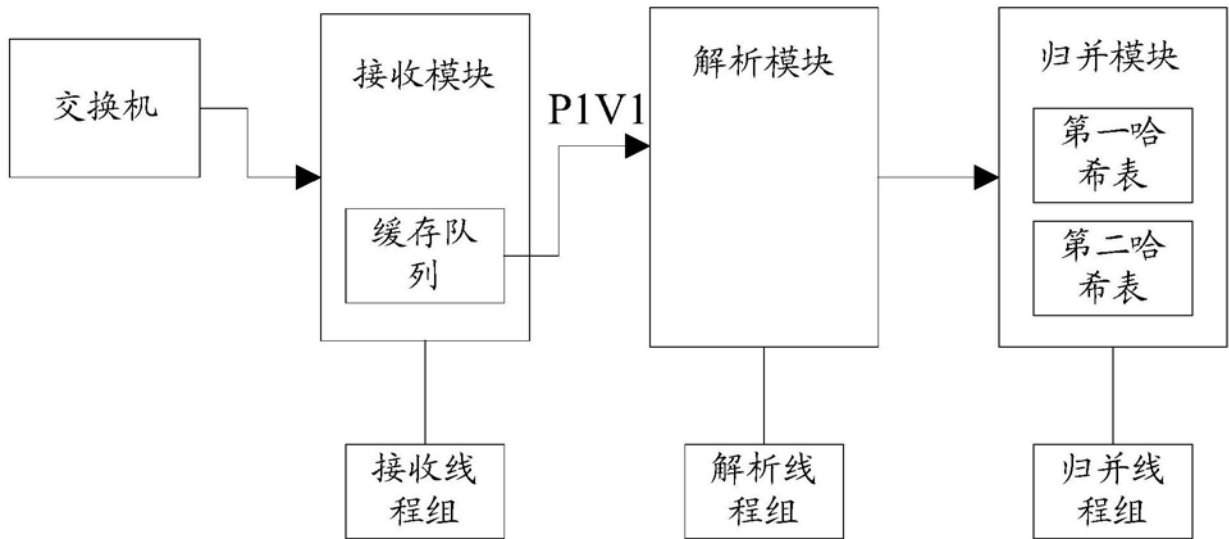


图8

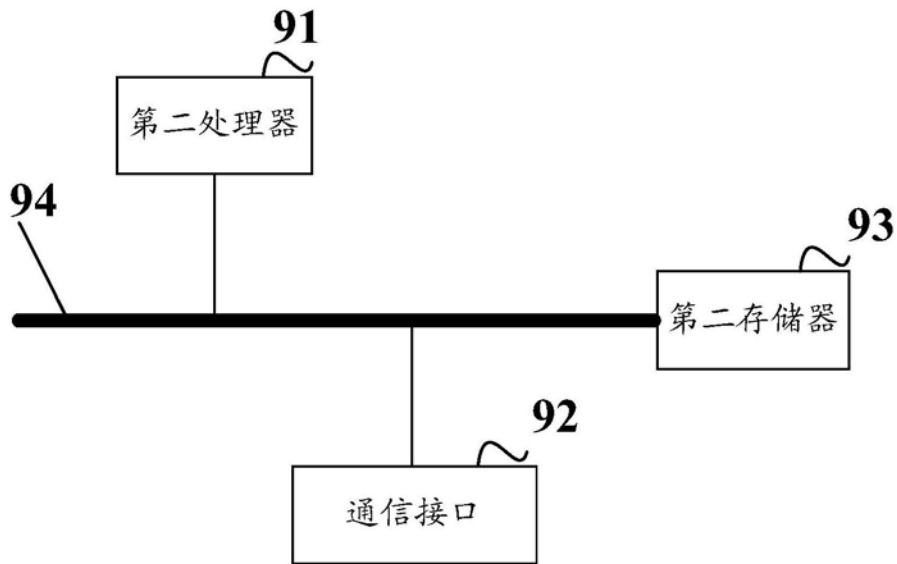


图9