

ACL 规则处理方法、装置、计算机设备和可读介质

技术领域

本公开涉及通信技术的包分类领域，具体涉及一种 ACL 规则处理方法、装置、计算机设备和可读介质。

背景技术

ACL (Access Control Lists, 访问控制列表) 硬件实现方案，分为 TCAM (ternary content addressable memory, 三态内容寻址存储器) 与 RAM (Random Access Memory, 随机存取存储器) 两类，其中 TCAM 实现方案利用 TCAM 硬件特性直接存储 ACL 规则。RAM 类方案通过高效的算法将 ACL 规则组织存储于 RAM 结构中，配合定制的转发面架构，通常具备较低的面积功耗实现代价。

由于 ACL 业务本身具备多匹配特性，因此在转发面架构中需要设计优先级仲裁模块，用于选取多条匹配规则中优先级最高的规则输出至后级，基于 RAM 实现的算法方案在规则优先级仲裁方面与 TCAM 实现方案有所区别，RAM 方案将规则过滤方法信息与规则对应的优先级信息共同存储于 RAM 规则中，在多规则匹配优先级仲裁时，基于规则存储对应的优先级高低来决策最终结果。而 TCAM 方案通常为通过命中规则存储地址的高低来决策最终结果。

发明内容

本公开提供一种 ACL 规则处理方法、装置、计算机设备和可读介质。

第一方面，本公开实施例提供一种 ACL 规则处理方法，应用于 ACL 规则处理装置，所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级，所述 ACL 处理装置内预设有关基线指针参数，所述优先级基线指针参数指向已存储的优先级最高的 ACL 规则，且所述优先级基线指针参数的值为所述已存储的优先级最高的 ACL 规则的相对优先级，所述方法包括：根据待存储 ACL 规则与已存储 ACL 规则的优先级关系，确定所述待存储 ACL 规则的存储地址；若所述存储地址中当前已存储有 ACL 规则，则根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址，确定搬移方向；至少根据所述搬移方向确定搬移的目标地址；根据所述搬移方向和目标地址逐条搬移 ACL 规则，直到空出所述存储地址，其中，每搬移一次 ACL 规则，更新当前搬移的 ACL 规则的相对优先级；将所述待存储 ACL 规则存储在所述存储地址，并根据所述存储地址记录所述待存储 ACL 规则的相对优先级。

另一方面，本公开实施例提供一种 ACL 规则处理方法，包括：根据待存储 ACL 规则与已存储 ACL 规则的优先级关系，确定待存储 ACL 规则的存储地址；若存储地址中当前已存储有 ACL 规则，则根据存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离存储地址最近的空洞地址和高地址方向上距离存储地址最近的空洞地址，确定搬移方向；

至少根据搬移方向确定搬移的目标地址；根据搬移方向和目标地址逐条搬移 ACL 规则，直到空出所述存储地址；将待存储 ACL 规则存储在所述存储地址，并根据存储地址记录待存储 ACL 规则的相对优先级。

又一方面，本公开实施例还提供一种 ACL 规则处理装置，所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级，所述 ACL 处理装置内预设有优先级基线指针参数，所述优先级基线指针参数指向已存储的优先级最高的 ACL 规则，且所述优先级基线指针参数的值为所述已存储的优先级最高的 ACL 规则的相对优先级；所述 ACL 规则处理装置包括：结果存储模块和优先级处理模块，所述优先级处理模块包括存储地址确定单元、搬移方向确定单元、目标地址确定单元、搬移单元和第一处理单元；所述存储地址确定单元被配置为根据待存储 ACL 规则与已存储 ACL 规则的优先级关系，确定所述待存储 ACL 规则的存储地址；所述搬移方向确定单元被配置为当所述存储地址中当前已存储有 ACL 规则时，根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址，确定搬移方向；所述目标地址确定单元被配置为至少根据所述搬移方向确定搬移的目标地址；所述搬移单元被配置为根据所述搬移方向和目标地址逐条搬移 ACL 规则，直到空出所述存储地址，其中，每搬移一次 ACL 规则，更新当前搬移的 ACL 规则的相对优先级；所述第一处理单元被配置为将所述待存储 ACL 规则存储在所述存储地址；所述结果存储模块被配置为根据所述存储地址记录所述待存储 ACL 规则的相对优先级。

又一方面，本公开实施例还提供一种计算机设备，包括：一个或多个处理器；存储装置，其上存储有一个或多个程序；当所述一个或多个程序被所述一个或多个处理器执行时，使得所述一个或多个处理器实现如前所述的 ACL 规则处理方法。

又一方面，本公开实施例还提供一种计算机可读介质，其上存储有计算机程序，其中，所述程序被执行时实现如前所述的 ACL 规则处理方法。

附图说明

图 1a 为本公开实施例提供的转发面 RAM 硬件存储示意图；

图 1b 为本公开实施例提供的控制面链表管理示意图；

图 2 为本公开实施例提供的 ACL 规则处理流程示意图之一；

图 3 为本公开实施例提供的确定搬移方向的流程示意图；

图 4 为本公开实施例提供的计算低地址方向上的搬移代价的流程示意图；

图 5 为本公开实施例提供的 ACL 规则处理流程示意图之二；

图 6 为本公开实施例提供的计算 ACL 规则的绝对优先级的流程示意图；

图 7 为本公开实施例提供的不修改优先级基线指针参数场景的 ACL 规则搬移示意图；

图 8 为本公开实施例提供的修改优先级基线指针参数场景的 ACL 规则搬移示意图；

图 9 为本公开实施例提供的 ACL 规则处理装置的结构示意图之一；

图 10 为本公开实施例提供的 ACL 规则处理装置的结构示意图之二；

图 11 为本公开实施例提供的 ACL 规则处理装置的结构示意图之三。

具体实施方式

在下文中将参考附图更充分地描述示例实施例，但是所述示例实施例可以以不同形式来体现且不当被解释为限于本文阐述的实施例。反之，提供这些实施例的目的在于使本公开透彻和完整，并将使本领域技术人员充分理解本公开的范围。

如本文所使用的，术语“和/或”包括一个或多个相关列举条目的任何和所有组合。

本文所使用的术语仅用于描述特定实施例，且不意欲限制本公开。如本文所使用的，单数形式“一个”和“该”也意欲包括复数形式，除非上下文另外清楚指出。还将理解的是，当本说明书中使用术语“包括”和/或“由……制成”时，指定存在所述特征、整体、步骤、操作、元件和/或组件，但不排除存在或添加一个或多个其他特征、整体、步骤、操作、元件、组件和/或其群组。

本文所述实施例可借助本公开的理想示意图而参考平面图和/或截面图进行描述。因此，可根据制造技术和/或容限来修改示例图示。因此，实施例不限于附图中所示的实施例，而是包括基于制造工艺而形成的配置的修改。因此，附图中例示的区具有示意性属性，并且图中所示区的形状例示了元件的区的具体形状，但并不旨在是限制性的。

除非另外限定，否则本文所用的所有术语(包括技术和科学术语)的含义与本领域普通技术人员通常理解的含义相同。还将理解，诸如那些在常用字典中限定的那些术语应当被解释为具有与其在相关技术以及本公开的背景下的含义一致的含义，且将不解释为具有理想化或过度形式上的含义，除非本文明确如此限定。

在用户动态更新 ACL 表的场景中，仅指定当前下发规则与已下发的某个规则的优先级高低关系，在 RAM 类算法实现方案中，受限转发面匹配特性，每条规则必须指定一项全局优先级值，用于最终结果仲裁，因此控制面需要对用户下发的规则进行隐式的优先级分配。业内常用的解决方案是将规则集顺序编码，初始下发的规则优先级置为 0 (假设 0 为最高优先级)，在某些场景下，后续下发的规则的优先级指定比某条已下发规则的优先级高，先将低于当前下发规则优先级的全部已下发规则的优先级的值加 1，从而空出一个优先级数值，将新下发规则的优先级置为该优先级数值。

在上述方案场景中，每次下发一条规则，最差情况下需要修改全部已下发规则优先级，即修改规则优先级的操作次数为 N，N 为已下发规则的总数量。对于容量较大的 ACL，修改规则优先级的操作次数较多，会增加 ACL 规则从配置到生效的时间，降低 ACL 规则的配置效率。

为此，本公开实施例提供了一种 ACL 规则处理方案。

本公开实施例提供的 ACL 规则处理方案涉及转发面与控制面两部分，转发面由硬件(如 RAM)实现，负责完成匹配报文的优先级存储与优先级仲裁功能，控制面使用软件实现，负责用户下发规则隐式优先级的分配与 RAM 硬件读写控制。

假设转发面 RAM 全局地址范围为 0-MAX_ADDR，MAX_ADDR 为 ACL 规则的最大存储地址，如图 1a 所示，转发面 RAM 中已存储的 ACL 规则按照控制面分配的相对优先级从小到大存储，低优先级值代表高优先级。转发面存储优先级基线指针参数 PRIO_BASE，优先级基线指针参数的初始值为 0，同时控制面软件全局存储并管理该优先级基线指针参数。各个 ACL 规则在转发面 RAM 中存储的相对优先级为其存储地址序号。

如图 1b 所示，控制面使用链表管理用户下发的规则，链表中头部节点(最左侧节点)对应最高优先级的 ACL 规则，每个链表节点中记录 ACL 规则与该 ACL 规则的相对优先级(即该

ACL 规则的存储地址)。

本公开实施例提供的 ACL 规则处理方法, 应用于 ACL 规则处理装置, 所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级, 所述 ACL 处理装置内预设有优先级基线指针参数 PRIO_BASE, 优先级基线指针参数指向已存储的优先级最高的 ACL 规则, 且优先级基线指针参数的值为已存储的优先级最高的 ACL 规则的相对优先级。

在一些实施例中, 所述 ACL 规则处理装置包括 RAM, 并且所述 ACL 规则的存储地址为其在 RAM 中的存储地址序号 (RAM 地址)。

如图 2 所示, 本公开实施例提供的 ACL 规则处理方法包括以下步骤 S11 至 S15。

在步骤 S11 中, 根据待存储 ACL 规则与已存储 ACL 规则的优先级关系, 确定待存储 ACL 规则的存储地址。

在本步骤中, 根据用户指定的待存储 ACL 规则与已存储 ACL 规则的优先级关系, 将待存储 ACL 规则插入到控制面链表中, 确定待存储 ACL 规则在 RAM 中的存储位置 (及存储地址), 记为 ins_pos。

在一些实施例中, 若待存储 ACL 规则的优先级小于已存储 ACL 规则的最低优先级 k, 说明待存储 ACL 规则的优先级最低, 则待存储 ACL 规则的存储地址为 (k+1), 即待存储 ACL 规则插入链表尾部 (最右侧)。若待存储 ACL 规则的优先级大于已存储 ACL 规则的最低优先级 k 且小于已存储 ACL 规则的最高优先级, 说明待存储 ACL 规则的优先级处于已存储 ACL 规则的优先级的中间位置, 则确定已存储 ACL 规则中小于待存储 ACL 规则的优先级的最大优先级 p, 并确定待存储 ACL 规则的存储地址为 p, 即待存储 ACL 规则插入链表中小于待存储 ACL 规则的优先级的最大优先级 p 的节点之前, 也就是说, 链表中待存储 ACL 规则紧邻的后继节点的优先级为比该待存储 ACL 规则的优先级小的各个优先级中的最大的优先级。

在步骤 S12 中, 若存储地址中当前已存储有 ACL 规则, 则根据存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离存储地址最近的空洞地址和高地址方向上距离存储地址最近的空洞地址, 确定搬移方向。

搬移方向包括低地址方向和高地址方向。设控制面管理优先级基线指针参数为 PRIO_BASE、ACL 规则的最大存储地址为 MAX_ADDR、低地址方向上距离存储地址最近的空洞地址为 m、高地址方向上距离存储地址最近的空洞地址为 n。

在本步骤中, 若存储地址 ins_pos 当前已存储有其他 ACL 规则, 则需要确定该 ACL 规则的搬移方向, 使该存储地址 ins_pos 空出, 以便存储待存储 ACL 规则。需要说明的是, 若存储地址 ins_pos 当前为空, 说明该存储地址 ins_pos 没有存储有其他 ACL 规则, 无需进行 ACL 规则搬移, 则直接将待存储 ACL 规则存储在存储地址 ins_pos。

在步骤 S13 中, 至少根据搬移方向确定搬移的目标地址。

目标地址记为 dst_pos。

在步骤 S14 中, 根据搬移方向和目标地址逐条搬移 ACL 规则, 直到空出存储地址。

需要说明的是, 每搬移一次 ACL 规则, 由于该 ACL 规则的存储地址发生了变化, 相应的, 就要更新当前搬移的 ACL 规则的相对优先级。也就是说, 每次搬移 ACL 规则之后, 修改控制面链表中记录的 ACL 规则的相对优先级值信息, 并将该 ACL 规则的更新后的相对优先级值同步写入 RAM 中。

在本步骤中, 若搬移方向为低地址方向, 则先将存储地址为 (dst_pos+1) 的 ACL 规则搬

移至存储地址为 dst_pos 的位置, 再将存储地址为 $(\text{dst_pos}+2)$ 的 ACL 规则搬移至存储地址为 $(\text{dst_pos}+1)$ 的位置, 以此类推, 直到空出存储地址 ins_pos (该存储地址即为待插入 ACL 规则的位置)。若搬移方向为高地址方向, 则先将存储地址为 $(\text{dst_pos}-1)$ ACL 规则搬移至存储地址为 dst_pos 的位置, 再将存储地址为 $(\text{dst_pos}-2)$ 的 ACL 规则搬移至存储地址为 $(\text{dst_pos}-1)$ 的位置, 以此类推, 直到空出存储地址 ins_pos 。

在步骤 S15 中, 将待存储 ACL 规则存储在存储地址, 并根据存储地址记录待存储 ACL 规则的相对优先级。

在本步骤中, 在存储地址 ins_pos 中插入待存储 ACL 规则, 并将该待存储 ACL 规则的相对优先级写入 RAM 中。

本公开实施例提供的 ACL 规则处理方法及装置, 根据待存储 ACL 规则与已存储 ACL 规则的优先级关系, 确定待存储 ACL 规则的存储地址; 若存储地址中当前已存储有 ACL 规则, 则根据存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离存储地址最近的空洞地址和高地址方向上距离存储地址最近的空洞地址, 确定搬移方向; 至少根据搬移方向确定搬移的目标地址; 根据搬移方向和目标地址逐条搬移 ACL 规则, 直到空出所述存储地址; 将待存储 ACL 规则存储在所述存储地址, 并根据存储地址记录待存储 ACL 规则的相对优先级。本公开实施例在全局维护优先级基线指针参数, 使其一直指向优先级最高的 ACL 规则, 在搬移已存储的 ACL 规则的过程中, 将优先级基线指针参数作为确定搬移方向的考量因素, 能够降低搬移代价, 搬移次数至少可以减少一半, 从而减少硬件操作次数, 提升 ACL 动态下表速率, 提高 ACL 规则更新效率。

在一些实施例中, 如图 2 所示, 所述根据存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离存储地址最近的空洞地址和高地址方向上距离存储地址最近的空洞地址, 确定搬移方向 (即步骤 S12) 包括以下步骤 S21 至 S23。

在步骤 S21 中, 根据存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址, 计算低地址方向的搬移代价。

计算低地址方向的搬移代价 cost_low 的具体实现方式后续结合图 4 再详细说明。

在步骤 S22 中, 根据存储地址和高地址方向上距离存储地址最近的空洞地址, 计算高地址方向的搬移代价。

在一些实施例中, 通过计算高地址方向上距离存储地址最近的空洞地址 n 与存储地址 ins_pos 之差, 得到高地址方向的搬移代价 cost_high , 即 $\text{cost_high}=n-\text{ins_pos}$ 。

在步骤 S23 中, 根据低地址方向的搬移代价和高地址方向的搬移代价, 确定搬移方向。

在一些实施例中, 所述根据低地址方向的搬移代价和高地址方向的搬移代价, 确定搬移方向, 包括: 确定低地址方向的搬移代价和高地址方向的搬移代价中的最小值, 将所述最小值对应的方向作为搬移方向。

需要说明的是, 步骤 S21 和步骤 S22 的执行顺序不限, 也可同步执行。

通过步骤 S21-S23 可以看出, 搬移代价反映了 ACL 规则搬移次数, 搬移代价越小, 说明搬移 ACL 规则的次数越少。通过比较低地址方向的搬移代价 cost_low 与高地址方向的搬移代价 cost_high 的大小, 选定最小值对应的搬移方向作为本次 ACL 规则的搬移方向, 可以减少搬移次数。

在一些实施例中, 如图 4 所示, 所述根据所述存储地址、优先级基线指针参数、ACL 规

则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址，计算低地址方向上的搬移代价（即步骤 S21），包括以下步骤 S121 至 S123：

在步骤 S121 中，根据存储地址、优先级基线指针参数和 ACL 规则的最大存储地址，计算第一搬移代价。

第一搬移代价 $cost_base$ 为从存储地址 ins_pos 搬移到优先级基线指针参数 $PRIO_BASE$ 指向的 ACL 规则地址的搬移代价。

在一些实施例中，若存储地址小于优先级基线指针参数的值，则将用于存储 ACL 规则的地址形成闭环，并向低地址方向环式计算第一搬移代价，即若 $ins_pos < PIR0_BASE$ ，则第一搬移代价 $cost_base = ACL$ 规则的最大存储地址 $MAX_ADDR - 优先级基线指针参数的值 PIR0_BASE + 2 + 存储地址 ins_pos$ 。也就是说，在 $ins_pos < PIR0_BASE$ 的情况下，需要向低地址方向环式计算第一搬移代价。

若存储地址大于或等于优先级基线指针参数的值，则向高地址方向计算第一搬移代价，即若 ins_pos （即 p ） $\geq PIR0_BASE$ ，则第一搬移代价 $cost_base = 存储地址 ins_pos - 优先级基线指针参数的值 PIR0_BASE + 1$ ，也就是说，在 ins_pos （即 p ） $\geq PIR0_BASE$ 的情况下，不需要环式计算第一搬移代价，而是直接向高地址方向计算第一搬移代价。

环式计算是指，在低地址方向从存储地址为 0 的位置跨越至地址最右侧位置形成地址环，根据该地址环进行计算。环式计算可以在计算第一搬移代价 $cost_base$ 时使用，也可以在计算第二搬移代价 $cost_m$ 时使用。

在步骤 S122 中，根据存储地址、低地址方向上距离存储地址最近的空洞地址和 ACL 规则的最大存储地址，计算第二搬移代价。

第二搬移代价 $cost_m$ 为从存储地址 ins_pos 搬移到低地址方向上距离存储地址 ins_pos 最近的空洞地址 m 的搬移代价。

在一些实施例中，若存储地址小于低地址方向上距离存储地址最近的空洞地址，则将用于存储 ACL 规则的地址形成闭环，并向低地址方向环式计算第二搬移代价。即若 $ins_pos < m$ ，则第二搬移代价 $cost_m = ACL$ 规则的最大存储地址 $MAX_ADDR + 1 - 低地址方向上距离存储地址最近的空洞地址 m + 存储地址 ins_pos$ 。也就是说，在 $ins_pos < m$ 的情况下，需要向低地址方向环式计算第二搬移代价。

若存储地址大于或等于低地址方向上距离存储地址最近的空洞地址，则向高地址方向计算第二搬移代价。即若 $ins_pos \geq m$ ，则第二搬移代价 $cost_m = 存储地址 ins_pos - 低地址方向上距离存储地址最近的空洞地址 m$ ，也就是说，在 $ins_pos \geq m$ 的情况下，不需要环式计算第二搬移代价，而是直接向高地址方向计算第二搬移代价。

由于优先级基线指针参数的设计，允许 ACL 规则搬移朝着低地址方向从存储地址为 0 的位置向空洞位置或优先级基线指针参数指向的位置搬移，从存储地址为 0 的位置搬移至 MAX_ADDR 位置的搬移代价为 1，因此当目的地址值（即 m 或 $PIR0_BASE$ ）大于原始地址值（ ins_pos ）时，低地址方向搬移代价为 $MAX_ADDR + 1 - 目的地址值 + 原始地址值$ 。需要说明的是，当计算第一搬移代价时，目的地址值实际上是 $PRIO_BASE - 1$ 。

在步骤 S123 中，确定第一搬移代价和第二搬移代价中的最小值，将所述最小值作为低地址方向上的搬移代价。

在本步骤中，低地址方向搬移代价 $= \min(cost_base, cost_m)$ 。

在一些实施例中,述至少根据搬移方向确定搬移的目标地址(即步骤S13),包括:若搬移方向为低地址方向,且低地址方向的搬移代价为第一搬移代价,则搬移的目标地址为优先级基线指针参数的值减1。若搬移方向为低地址方向,且低地址方向的搬移代价为第二搬移代价,则搬移的目标地址为低地址方向上距离所述存储地址最近的空洞地址。若搬移方向为高地址方向,则搬移的目标地址为高地址方向上距离存储地址最近的空洞地址。也就是说,若选定搬移方向为低地址方向,且 $\text{MIN}(\text{cost_base}, \text{cost_m}) = \text{cost_base}$,则 $\text{dst_pos} = \text{PIRO_BASE} - 1$ 。若选定搬移方向为低地址方向,且 $\text{MIN}(\text{cost_base}, \text{cost_m}) = \text{cost_m}$,则 $\text{dst_pos} = m$ 。若选定搬移方向为高地址方向,则 $\text{dst_pos} = n$ 。

在一些实施例中,若搬移方向为低地址方向,且低地址方向上的搬移代价为第一搬移代价,则在至少根据搬移方向确定搬移的目标地址之后,所述方法还包括以下步骤:将优先级基线指针参数的值减1。也就是说,在搬移方向为低地址方向,且 $\text{MIN}(\text{cost_base}, \text{cost_m}) = \text{cost_base}$ 的情况下,需要搬移优先级基线指针参数PIRO_BASE指向的ACL规则,相应的,在确定出搬移的目标地址之后,优先级基线指针参数PIRO_BASE的值也需要相应进行调整(仍然指向位置搬移后的优先级最高的ACL规则)。需要说明的是,在搬移方向为低地址方向,且低地址方向的搬移代价为第二搬移代价的情况下,以及,在搬移方向为高地址方向的情况下,优先级基线指针参数PIRO_BASE指向的ACL规则没有发生搬移,相应的,不用调整优先级基线指针参数PIRO_BASE的值。

需要说明的是,步骤S121和步骤S122的执行顺序不限,例如也可同步执行。

以上为存储ACL规则(写入RAM)的流程。以下结合图5详细说明删除ACL规则的流程。

如图5所示,所述ACL规则处理方法还可以包括以下步骤S31至S32。

在步骤S31中,从已存储ACL规则中删除待删除ACL规则。

在本步骤中,根据用户下发的删除ACL规则信息索引至控制面链表对应的链表节点,根据链表节点记录的优先级信息清空转发面RAM位置存储的ACL规则。

在步骤S32中,若待删除ACL规则的存储地址与优先级基线指针参数的值相等,则将优先级基线指针参数的值加1。

在一些实施例中,在步骤S32之后,所述ACL规则处理方法还可以包括:在控制面的链表中删除与已删除的ACL规则对应的链表节点。

RAM中存储的ACL规则的优先级为相对优先级,但是在业务报文转发阶段,是根据ACL规则的绝对优先级进行优先级仲裁,因此,需要将ACL规则的相对优先级转换为绝对优先级。因此,如图6所示,所述ACL规则处理方法还可以包括以下步骤S41至S42。

在步骤S41中,若接收到业务报文且业务报文中至少2个ACL规则,则确定命中的各ACL规则的相对优先级。

在步骤S42中,至少根据命中的各ACL规则的相对优先级和优先级基线指针参数,计算命中的各ACL规则的绝对优先级。

在一些实施例中,若命中的ACL规则的相对优先级 r_prio 大于或等于优先级基线指针参数PRIO_BASE的值,则根据命中的ACL规则的相对优先级 r_prio 和优先级基线指针参数PRIO_BASE的值,向高地址方向计算命中的ACL规则的绝对优先级 $prio$ 。即若 $r_prio \geq \text{PRIO_BASE}$,则 $prio = r_prio - \text{PRIO_BASE}$ 。也就是说,在 $r_prio \geq \text{PRIO_BASE}$ 的情况下,不需要环式计算绝对优先级,而是直接向高地址方向计算绝对优先级。

若命中的 ACL 规则的相对优先级 r_prio 小于优先级基线指针参数 $PRI0_BASE$ 的值, 则将用于存储 ACL 规则的地址形成闭环, 并根据命中的 ACL 规则的相对优先级 r_prio 、优先级基线指针参数 $PRI0_BASE$ 的值、ACL 规则的最大存储地址 MAX_ADDR , 向低地址方向环式计算命中的 ACL 规则的绝对优先级 $prio$ 。即若 $r_prio < PRI0_BASE$, 则 $prio = MAX_ADDR + 1 - PRI0_BASE + r_prio$, 也就是说, 在 $r_prio < PRI0_BASE$ 的情况下, 需要向低地址方向环式计算绝对优先级。

为了清楚说明本公开实施例的方案, 以下分别结合图 7 和图 8, 对不修改优先级基线指针参数场景和修改优先级基线指针参数场景下的 ACL 规则搬移过程进行详细说明。

如图 7 所示, 待存储 ACL 规则为 $rule8$, 其存储地址 ins_pos 为当前 $rule3$ 的存储位置, 优先级基线指针参数 $PRI0_BASE$ 指向 $rule0$, $rule4$ 后一个位置为高地址方向上距离存储地址最近的空洞地址 n , 低地址方向上无空洞, 因此低地址方向的搬移代价 $cost_low$ 即为 $cost_base = 4$ ($rule2$ 、 $rule1$ 、 $rule0$ 、 $null$), 高地址方向的搬移代价 $cost_high = 2$ ($rule4$ 、 $null$), 因此, 确定搬移方向为高地址方向。由于搬移方向为高地址方向, 因此不用修改优先级基线指针参数 $PRI0_BASE$ 的值。逐条将 $rule4$ 和 $rule3$ 向高地址方向搬移, 直到空出存储地址 ins_pos 的位置, 将 $rule8$ 存储在该位置。

如图 8 所示, 待存储 ACL 规则为 $rule8$, 其存储地址 ins_pos 为当前 $rule3$ 的存储位置, 优先级基线指针参数 $PRI0_BASE$ 指向 $rule0$, $rule7$ 后一个位置为高地址方向上距离存储地址最近的空洞地址 n , 低地址方向上无空洞, 因此低地址方向的搬移代价 $cost_low$ 即为 $cost_base = 4$ ($rule2$ 、 $rule1$ 、 $rule0$ 、 $null$), 高地址方向的搬移代价 $cost_high = 5$ ($rule4$ 、 $rule5$ 、 $rule6$ 、 $rule7$ 、 $null$), 因此, 确定搬移方向为低地址方向, 且低地址方向的搬移代价 $cost_low$ 与 $cost_base$ 相等 (均为 4), 则将优先级基线指针参数 $PRI0_BASE$ 的值减 1。确定目标地址为 dst_pos 为优先级基线指针参数 $PRI0_BASE$ 当前指向的位置, 逐条将 $rule0$ 、 $rule1$ 、 $rule2$ 、 $rule3$ 向低地址方向搬移, 其中, $rule0$ 搬移至 $PRI0_BASE$ 当前指向的位置, 直到空出存储地址 ins_pos 的位置, 将 $rule8$ 存储在该位置。

基于相同的技术构思, 本公开实施例还提供一种 ACL 规则处理装置, 如图 9 所示, 包括: 结果存储模块 101 和优先级处理模块 102, 结果存储模块 101 用于存储 ACL 规则对应的相对优先级; 优先级处理模块 102 被配置为接收结果存储模块 101 输出的多条 ACL 规则对应的相对优先级, 进行优先级处理, 得到绝对优先级, 并将绝对优先级与对应的 ACL 规则及其结果发送至优先级比较模块。

在一些实施例中, 所示 ACL 规则处理装置还包括优先级比较模块 103, 优先级比较模块 103, 用于比较其输入的多个绝对优先级, 并输出优先级最高的 ACL 规则对应的结果。

结合图 9 和图 10 所示, 所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级, 所述 ACL 处理装置内预设有优先级基线指针参数, 所述优先级基线指针参数指向已存储的优先级最高的 ACL 规则, 且所述优先级基线指针参数的值为所述已存储的优先级最高的 ACL 规则的相对优先级。

优先级处理模块 102 包括存储地址确定单元 1021、搬移方向确定单元 1022、目标地址确定单元 1023、搬移单元 1024 和第一处理单元 1025, 存储地址确定单元 1021 被配置为根据待存储 ACL 规则与已存储 ACL 规则的优先级关系, 确定所述待存储 ACL 规则的存储地址。

搬移方向确定单元 1022 被配置为当所述存储地址中当前已存储有 ACL 规则时, 根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址, 确定搬移方向。

目标地址确定单元 1023 被配置为至少根据所述搬移方向确定搬移的目标地址。

搬移单元 1024 被配置为根据所述搬移方向和目标地址逐条搬移 ACL 规则, 直到空出所述存储地址, 其中, 每搬移一次 ACL 规则, 更新当前搬移的 ACL 规则的相对优先级。

第一处理单元 1025 被配置为将所述待存储 ACL 规则存储在所述存储地址。

结果存模块 101 被配置为根据所述存储地址记录所述待存储 ACL 规则的相对优先级。

在一些实施例中, 搬移方向确定单元 1022 被配置为根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址, 计算低地址方向的搬移代价; 根据所述存储地址和高地址方向上距离所述存储地址最近的空洞地址, 计算高地址方向的搬移代价; 根据所述低地址方向的搬移代价和所述高地址方向的搬移代价, 确定搬移方向。

在一些实施例中, 搬移方向确定单元 1022 被配置为确定所述低地址方向的搬移代价和所述高地址方向的搬移代价中的最小值, 将所述最小值对应的方向作为搬移方向。

在一些实施例中, 搬移方向确定单元 1022 被配置为计算所述高地址方向上距离所述存储地址最近的空洞地址与所述存储地址之差, 得到高地址方向的搬移代价。

在一些实施例中, 搬移方向确定单元 1022 被配置为根据所述存储地址、优先级基线指针参数和 ACL 规则的最大存储地址, 计算第一搬移代价, 所述第一搬移代价为从所述存储地址搬移到所述优先级基线指针参数指向的 ACL 规则地址的搬移代价; 根据所述存储地址、低地址方向上距离所述存储地址最近的空洞地址和 ACL 规则的最大存储地址, 计算第二搬移代价, 所述第二搬移代价为从所述存储地址搬移到所述低地址方向上距离所述存储地址最近的空洞地址的搬移代价; 确定所述第一搬移代价和所述第二搬移代价中的最小值, 将所述最小值作为低地址方向上的搬移代价。

在一些实施例中, 搬移方向确定单元 1022 被配置为, 当所述存储地址小于所述优先级基线指针参数的值时, 将用于存储 ACL 规则的地址形成闭环, 并向低地址方向环式计算第一搬移代价; 当所述存储地址大于或等于所述优先级基线指针参数的值时, 向高地址方向计算第一搬移代价。

在一些实施例中, 搬移方向确定单元 1022 被配置为, 当所述存储地址小于低地址方向上距离所述存储地址最近的空洞地址时, 将用于存储 ACL 规则的地址形成闭环, 并向低地址方向环式计算第二搬移代价; 当所述存储地址大于或等于低地址方向上距离所述存储地址最近的空洞地址时, 向高地址方向计算第二搬移代价。

在一些实施例中, 目标地址确定单元 1023 被配置为, 当所述搬移方向为低地址方向, 且低地址方向的搬移代价为第一搬移代价时, 搬移的目标地址为优先级基线指针参数的值减 1; 当所述搬移方向为低地址方向, 且低地址方向的搬移代价为第二搬移代价时, 搬移的目标地址为低地址方向上距离所述存储地址最近的空洞地址; 当所述搬移方向为高地址方向时, 搬移的目标地址为高地址方向上距离所述存储地址最近的空洞地址。

在一些实施例中, 目标地址确定单元 1023 还被配置为, 当所述搬移方向为低地址方向, 且低地址方向上的搬移代价为所述第一搬移代价时, 在至少根据所述搬移方向确定搬移的目

标地址之后，将所述优先级基线指针参数的值减 1。

在一些实施例中，存储地址确定单元 1021 被配置为，当待存储 ACL 规则的优先级小于已存储 ACL 规则的最低优先级 k 时，所述待存储 ACL 规则的存储地址为 (k+1)；当待存储 ACL 规则的优先级大于已存储 ACL 规则的最低优先级 k 且小于已存储 ACL 规则的最高优先级时，确定已存储 ACL 规则中小于待存储 ACL 规则的优先级的最大优先级 p，并确定所述待存储 ACL 规则的存储地址为 p。

在一些实施例中，第一处理单元 1025 还被配置为从已存储 ACL 规则中删除待删除 ACL 规则；当所述待删除 ACL 规则的存储地址与所述优先级基线指针参数的值相等时，将所述优先级基线指针参数的值加 1。

在一些实施例中，如图 11 所示，优先级处理模块 102 还包括第二处理单元 1026，第二处理单元 1026 被配置为，当接收到业务报文且所述业务报文中至少 2 个 ACL 规则时，确定命中的各 ACL 规则的相对优先级，至少根据所述命中的各 ACL 规则的相对优先级和所述优先级基线指针参数，计算所述命中的各 ACL 规则的绝对优先级。

在一些实施例中，第二处理单元 1026 被配置为，当命中的 ACL 规则的相对优先级大于或等于所述优先级基线指针参数的值时，根据所述命中的 ACL 规则的相对优先级和所述优先级基线指针参数的值，向高地址方向计算所述命中的 ACL 规则的绝对优先级；当命中的 ACL 规则的相对优先级小于所述优先级基线指针参数的值时，将用于存储 ACL 规则的地址形成闭环，并根据所述命中的 ACL 规则的相对优先级、所述优先级基线指针参数的值、所述 ACL 规则的最大存储地址，向低地址方向环式计算命中的 ACL 规则的绝对优先级。

本公开实施例还提供了一种计算机设备，该计算机设备包括：一个或多个处理器以及存储装置；其中，存储装置上存储有一个或多个程序，当上述一个或多个程序被上述一个或多个处理器执行时，使得上述一个或多个处理器实现如前述各实施例所提供的 ACL 规则处理方法。

本公开实施例还提供了一种计算机可读介质，其上存储有计算机程序，其中，该计算机程序被执行时实现如前述各实施例所提供的 ACL 规则处理方法。

本领域普通技术人员可以理解，上文中所公开方法中的全部或某些步骤、装置中的功能模块/单元可以被实施为软件、固件、硬件及其适当的组合。在硬件实施方式中，在以上描述中提及的功能模块/单元之间的划分不一定对应于物理组件的划分；例如，一个物理组件可以具有多个功能，或者一个功能或步骤可以由若干物理组件合作执行。某些物理组件或所有物理组件可以被实施为由处理器，如中央处理器、数字信号处理器或微处理器执行的软件，或者被实施为硬件，或者被实施为集成电路，如专用集成电路。这样的软件可以分布在计算机可读介质上，计算机可读介质可以包括计算机存储介质（或非暂时性介质）和通信介质（或暂时性介质）。如本领域普通技术人员公知的，术语计算机存储介质包括在用于存储信息（诸如计算机可读指令、数据结构、程序模块或其他数据）的任何方法或技术中实施的易失性和非易失性、可移除和不可移除介质。计算机存储介质包括但不限于 RAM、ROM、EEPROM、闪存或其他存储器技术、CD-ROM、数字多功能盘（DVD）或其他光盘存储、磁盒、磁带、磁盘存储或其他磁存储装置、或者可以用于存储期望的信息并且可以被计算机访问的任何其他的介质。此外，本领域普通技术人员公知的是，通信介质通常包含计算机可读指令、数据结构、程序

模块或者诸如载波或其他传输机制之类的调制数据信号中的其他数据，并且可包括任何信息递送介质。

本文已经公开了示例实施例，并且虽然采用了具体术语，但它们仅用于并仅应当被解释为一般说明性含义，并且不用于限制的目的。在一些实例中，对本领域技术人员显而易见的是，除非另外明确指出，否则可单独使用与特定实施例相结合描述的特征、特性和/或元素，或可与其他实施例相结合描述的特征、特性和/或元件组合使用。因此，本领域技术人员将理解，在不脱离由所附的权利要求阐明的本发明的范围的情况下，可进行各种形式和细节上的改变。

权 利 要 求 书

1、一种 ACL 规则处理方法，应用于 ACL 规则处理装置，所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级，所述 ACL 处理装置内预设有优先级基线指针参数，所述优先级基线指针参数指向已存储的优先级最高的 ACL 规则，且所述优先级基线指针参数的值为所述已存储的优先级最高的 ACL 规则的相对优先级，所述方法包括：

根据待存储 ACL 规则与已存储 ACL 规则的优先级关系，确定所述待存储 ACL 规则的存储地址；

若所述存储地址中当前已存储有 ACL 规则，则根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址，确定搬移方向；

至少根据所述搬移方向确定搬移的目标地址；

根据所述搬移方向和目标地址逐条搬移 ACL 规则，直到空出所述存储地址，其中，每搬移一次 ACL 规则，更新当前搬移的 ACL 规则的相对优先级；以及

将所述待存储 ACL 规则存储在所述存储地址，并根据所述存储地址记录所述待存储 ACL 规则的相对优先级。

2、如权利要求 1 所述的方法，其中，所述根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址，确定搬移方向，包括：

根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址，计算低地址方向的搬移代价；

根据所述存储地址和高地址方向上距离所述存储地址最近的空洞地址，计算高地址方向的搬移代价；以及

根据所述低地址方向的搬移代价和所述高地址方向的搬移代价，确定搬移方向。

3、如权利要求 2 所述的方法，其中，所述根据所述低地址方向的搬移代价和所述高地址方向的搬移代价，确定搬移方向，包括：

确定所述低地址方向的搬移代价和所述高地址方向的搬移代价中的最小值，将所述最小值对应的方向作为搬移方向。

4、如权利要求 2 所述的方法，其中，所述根据所述存储地址和高地址方向上距离所述存储地址最近的空洞地址，计算高地址方向的搬移代价，包括：

计算所述高地址方向上距离所述存储地址最近的空洞地址与所述存储地址之差，得到高地址方向的搬移代价。

5、如权利要求 2 所述的方法，其中，所述根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址，计算低地址方向上

的搬移代价, 包括:

根据所述存储地址、优先级基线指针参数和 ACL 规则的最大存储地址, 计算第一搬移代价, 所述第一搬移代价为从所述存储地址搬移到所述优先级基线指针参数指向的 ACL 规则地址的搬移代价;

根据所述存储地址、低地址方向上距离所述存储地址最近的空洞地址和 ACL 规则的最大存储地址, 计算第二搬移代价, 所述第二搬移代价为从所述存储地址搬移到所述低地址方向上距离所述存储地址最近的空洞地址的搬移代价; 以及

确定所述第一搬移代价和所述第二搬移代价中的最小值, 将所述最小值作为低地址方向上的搬移代价。

6、如权利要求 5 所述的方法, 其中, 所述根据所述存储地址、优先级基线指针参数和 ACL 规则的最大存储地址, 计算第一搬移代价, 包括:

若所述存储地址小于所述优先级基线指针参数的值, 则将用于存储 ACL 规则的地址形成闭环, 并向低地址方向环式计算第一搬移代价;

若所述存储地址大于或等于所述优先级基线指针参数的值, 则向高地址方向计算第一搬移代价。

7、如权利要求 5 所述的方法, 其中, 所述根据所述存储地址、低地址方向上距离所述存储地址最近的空洞地址和 ACL 规则的最大存储地址, 计算第二搬移代价, 包括:

若所述存储地址小于低地址方向上距离所述存储地址最近的空洞地址, 则将用于存储 ACL 规则的地址形成闭环, 并向低地址方向环式计算第二搬移代价;

若所述存储地址大于或等于低地址方向上距离所述存储地址最近的空洞地址, 则向高地址方向计算第二搬移代价。

8、如权利要求 5 所述的方法, 其中, 所述至少根据所述搬移方向确定搬移的目标地址, 包括:

若所述搬移方向为低地址方向, 且低地址方向的搬移代价为第一搬移代价, 则搬移的目标地址为优先级基线指针参数的值减 1;

若所述搬移方向为低地址方向, 且低地址方向的搬移代价为第二搬移代价, 则搬移的目标地址为低地址方向上距离所述存储地址最近的空洞地址;

若所述搬移方向为高地址方向, 则搬移的目标地址为高地址方向上距离所述存储地址最近的空洞地址。

9、如权利要求 8 所述的方法, 其中, 若所述搬移方向为低地址方向, 且低地址方向上的搬移代价为所述第一搬移代价, 则在至少根据所述搬移方向确定搬移的目标地址之后, 所述方法还包括: 将所述优先级基线指针参数的值减 1。

10、如权利要求 1 所述的方法, 其中, 所述根据待存储 ACL 规则与已存储 ACL 规则的优先级关系, 确定所述待存储 ACL 规则的存储地址, 包括:

若待存储 ACL 规则的优先级小于已存储 ACL 规则的最低优先级 k ，则所述待存储 ACL 规则的存储地址为 $(k+1)$ ；

若待存储 ACL 规则的优先级大于已存储 ACL 规则的最低优先级 k 且小于已存储 ACL 规则的最高优先级，则确定已存储 ACL 规则中小于待存储 ACL 规则的优先级的最大优先级 p ，并确定所述待存储 ACL 规则的存储地址为 p 。

11、如权利要求 1 所述的方法，还包括：

从已存储 ACL 规则中删除待删除 ACL 规则；并且

若所述待删除 ACL 规则的存储地址与所述优先级基线指针参数的值相等，则将所述优先级基线指针参数的值加 1。

12、如权利要求 1-11 任一项所述的方法，还包括：

若接收到业务报文且所述业务报文中命中至少 2 个 ACL 规则，则确定命中的各 ACL 规则的相对优先级，至少根据所述命中的各 ACL 规则的相对优先级和所述优先级基线指针参数，计算所述命中的各 ACL 规则的绝对优先级。

13、如权利要求 12 所述的方法，其中，所述至少根据所述命中的各 ACL 规则的相对优先级和所述优先级基线指针参数，计算所述命中的各 ACL 规则的绝对优先级，包括：

若命中的 ACL 规则的相对优先级大于或等于所述优先级基线指针参数的值，则根据所述命中的 ACL 规则的相对优先级和所述优先级基线指针参数的值，向高地址方向计算所述命中的 ACL 规则的绝对优先级；

若命中的 ACL 规则的相对优先级小于所述优先级基线指针参数的值，则将用于存储 ACL 规则的地址形成闭环，并根据所述命中的 ACL 规则的相对优先级、所述优先级基线指针参数的值、所述 ACL 规则的最大存储地址，向低地址方向环式计算命中的 ACL 规则的绝对优先级。

14、一种 ACL 规则处理装置，其中，所述 ACL 规则处理装置内 ACL 规则的存储地址为 ACL 规则的相对优先级，所述 ACL 处理装置内预设有优先级基线指针参数，所述优先级基线指针参数指向已存储的优先级最高的 ACL 规则，且所述优先级基线指针参数的值为所述已存储的优先级最高的 ACL 规则的相对优先级；所述 ACL 规则处理装置包括：结果存储模块和优先级处理模块，所述优先级处理模块包括存储地址确定单元、搬移方向确定单元、目标地址确定单元、搬移单元和第一处理单元；

所述存储地址确定单元被配置为根据待存储 ACL 规则与已存储 ACL 规则的优先级关系，确定所述待存储 ACL 规则的存储地址；

所述搬移方向确定单元被配置为当所述存储地址中当前已存储有 ACL 规则时，根据所述存储地址、优先级基线指针参数、ACL 规则的最大存储地址、低地址方向上距离所述存储地址最近的空洞地址和高地址方向上距离所述存储地址最近的空洞地址，确定搬移方向；

所述目标地址确定单元被配置为至少根据所述搬移方向确定搬移的目标地址；

所述搬移单元被配置为根据所述搬移方向和目标地址逐条搬移 ACL 规则，直到空出所述存储地址，其中，每搬移一次 ACL 规则，更新当前搬移的 ACL 规则的相对优先级；

所述第一处理单元被配置为将所述待存储 ACL 规则存储在所述存储地址；并且
所述结果存模块被配置为根据所述存储地址记录所述待存储 ACL 规则的相对优先级。

15、一种计算机设备，包括：

一个或多个处理器；

存储装置，其上存储有一个或多个程序；其中，

当所述一个或多个程序被所述一个或多个处理器执行时，使得所述一个或多个处理器实现如权利要求 1-13 任一项所述的 ACL 规则处理方法。

16、一种计算机可读介质，其上存储有计算机程序，其中，所述程序被执行时实现如权利要求 1-13 任一项所述的 ACL 规则处理方法。