什么是 ACL

文档版本 02

发布日期 2020-11-16

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： https://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 什么是 ACL

1.1 引言

本文档简要介绍了什么是ACL。以下章节将详细阐述ACL的基本概念、ACL的常用匹配项及ACL的常用配置原则。

1.2 ACL 简介

定义

访问控制列表ACL（Access Control List）是由一条或多条规则组成的集合。所谓规则，是指描述报文匹配条件的判断语句，这些条件可以是报文的源地址、目的地址、端口号等。

ACL本质上是一种报文过滤器，规则是过滤器的滤芯。设备基于这些规则进行报文匹配，可以过滤出特定的报文，并根据应用ACL的业务模块的处理策略来允许或阻止该报文通过。

目的

随着网络的飞速发展，网络安全和网络服务质量QoS（Quality of Service）问题日益突出。

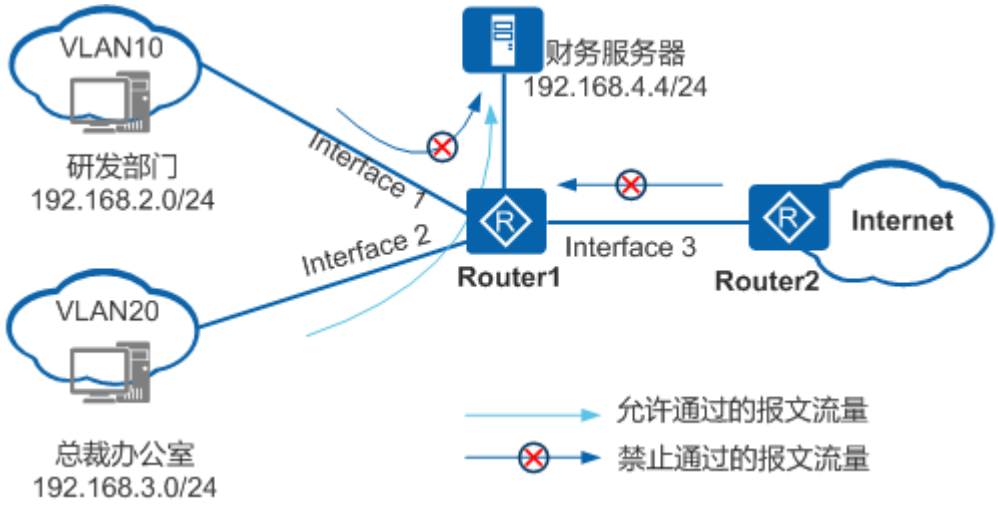
- 企业重要服务器资源被随意访问，企业机密信息容易泄露，造成安全隐患。
- Internet病毒肆意侵略企业内网，内网环境的安全性堪忧。
- 网络带宽被各类业务随意挤占，服务质量要求最高的语音、视频业务的带宽得不到保障，造成用户体验差。

以上种种问题，都对正常的网络通信造成了很大的影响。因此，提高网络安全性服务质量迫在眉睫。ACL就在这种情况下应运而生了。

通过ACL可以实现对网络中报文流的精确识别和控制，达到控制网络访问行为、防止网络攻击和提高网络带宽利用率的目的，从而切实保障网络环境的安全性和网络服务质量的可靠性。

图1-1是一个典型的ACL应用组网场景。

图 1-1 ACL 典型应用场景



- 某企业为保证财务数据安全，禁止研发部门访问财务服务器，但总裁办公室不受限制。实现方式：

  在Interface 1的入方向上部署ACL，禁止研发部门访问财务服务器的报文通过。Interface 2上无需部署ACL，总裁办公室访问财务服务器的报文默认允许通过。
- 保护企业内网环境安全，防止Internet病毒入侵。实现方式：

  在Interface 3的入方向上部署ACL，将病毒经常使用的端口予以封堵。

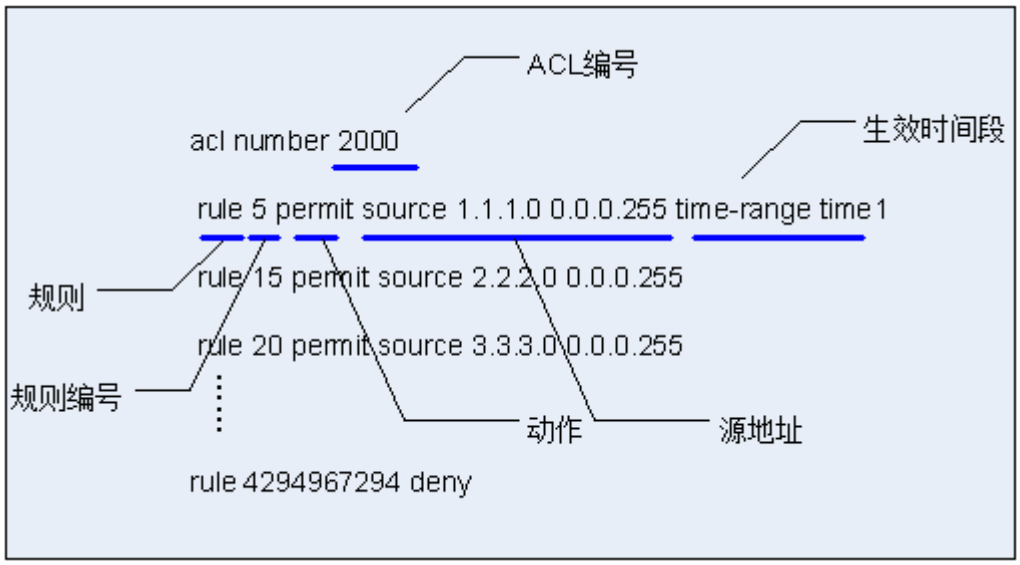
1.3 ACL 的基本原理

ACL由一系列规则组成，通过将报文与ACL规则进行匹配，设备可以过滤出特定的报文。

ACL 的组成

一条ACL的结构组成，如图1-2所示。

图 1-2 ACL 的结构组成



- **ACL编号**：用于标识ACL，表明该ACL是数字型ACL。

  根据ACL规则功能的不同，ACL被划分为基本ACL、高级ACL、二层ACL和用户ACL这几种类型，每类ACL编号的取值范围不同。关于每类ACL编号的详细介绍，请参见**1.4 ACL的分类**。

  除了可以通过ACL编号标识ACL，设备还支持通过名称来标识ACL，就像用域名代替IP地址一样，更加方便记忆。这种ACL，称为命名型ACL。

  命名型ACL实际上是“名字+数字”的形式，可以在定义命名型ACL时同时指定ACL编号。如果不指定编号，则由系统自动分配。例如，下面就是一个既有名字“deny-telnet-login”又有编号“3998”的ACL。

  ```
  #
  acl name deny-telnet-login 3998
   rule 0 deny tcp source 10.152.0.0 0.0.63.255 destination 10.64.0.97 0 destination-port eq
  telnet
   rule 5 deny tcp source 10.242.128.0 0.0.127.255 destination 10.64.0.97 0 destination-port eq
  telnet
  #
  ```

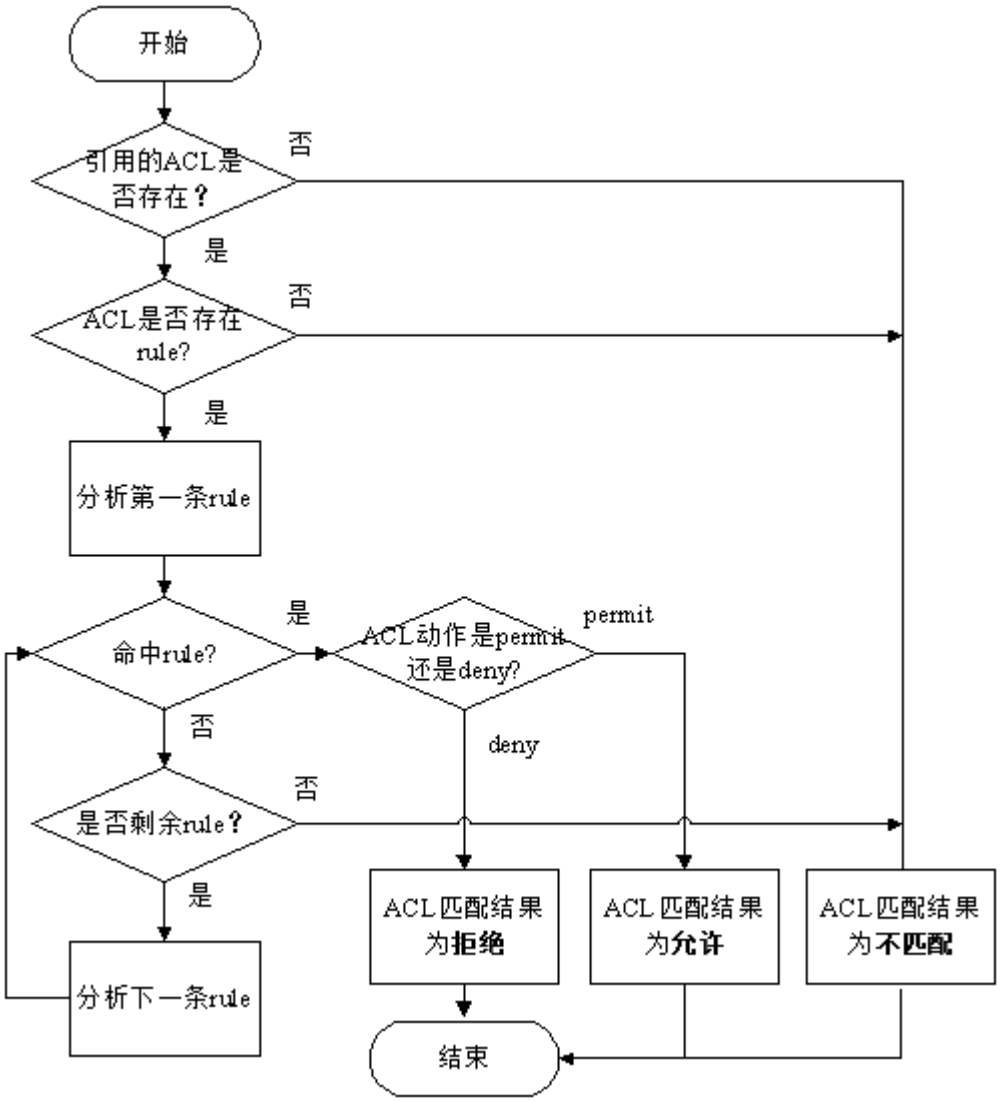
- **规则**：即描述报文匹配条件的判断语句。
  - **规则编号**：用于标识ACL规则。可以自行配置规则编号，也可以由系统自动分配。

    ACL规则的编号范围是0～4294967294，所有规则均按照规则编号从小到大进行排序。所以，**图1-2**中的rule 5排在首位，而规则编号最大的rule 4294967294排在末位。系统按照规则编号从小到大的顺序，将规则依次与报文匹配，一旦匹配上一条规则即停止匹配。
  - **动作**：包括permit/deny两种动作，表示允许/拒绝。
  - **匹配项**：ACL定义了极其丰富的匹配项。除了**图1-2**中的源地址和生效时间段，ACL还支持很多其他规则匹配项。例如，二层以太网帧头信息（如源MAC、目的MAC、以太帧协议类型）、三层报文信息（如目的地址、协议类型）以及四层报文信息（如TCP/UDP端口号）等。关于每种匹配项的详细介绍，请参见**1.8 ACL的常用匹配项**。

ACL 的匹配机制

设备将报文与ACL规则进行匹配时，遵循“一旦命中即停止匹配”的机制，如图1-3所示。

图 1-3 ACL 的匹配机制



首先系统会查找设备上是否配置了ACL。

- 如果ACL不存在，则返回ACL匹配结果为：不匹配。
- 如果ACL存在，则查找设备是否配置了ACL规则。
  - 如果规则不存在，则返回ACL匹配结果为：不匹配。
  - 如果规则存在，则系统会从ACL中编号最小的规则开始查找。
- 如果匹配上了permit规则，则停止查找规则，并返回ACL匹配结果为：匹配（允许）。
- 如果匹配上了deny规则，则停止查找规则，并返回ACL匹配结果为：匹配（拒绝）。

- 如果未匹配上规则，则继续查找下一条规则，以此循环。如果一直查到最后一条规则，报文仍未匹配上，则返回ACL匹配结果为：不匹配。

从整个ACL匹配流程可以看出，报文与ACL规则匹配后，会产生两种匹配结果：“匹配”和“不匹配”。

- 匹配（命中规则）：指存在ACL，且在ACL中查找到了符合匹配条件的规则。

  不论匹配的动作是“permit”还是“deny”，都称为“匹配”，而不是只是匹配上permit规则才算“匹配”。
- 不匹配（未命中规则）：指不存在ACL，或ACL中无规则，再或者在ACL中遍历了所有规则都没有找到符合匹配条件的规则。

  以上三种情况，都叫做“不匹配”。

1.4 ACL 的分类

基于 ACL 标识方法的划分

划分如下：

- 数字型ACL：传统的ACL标识方法。创建ACL时，指定一个唯一的数字标识该ACL。
- 命名型ACL：通过名称代替编号来标识ACL。

用户在创建ACL时可以为其指定编号，不同的编号对应不同类型的ACL，如**表1-1**所示。同时，为了便于记忆和识别，用户还可以创建命名型ACL，即在创建ACL时为其设置名称。命名型ACL，也可以是“名称 数字”的形式，即在定义命名型ACL时，同时指定ACL编号。如果不指定编号，系统则会自动为其分配一个数字型ACL的编号。

说明

命名型ACL一旦创建成功，便不允许用户再修改其名称。如果删除ACL名称，则表示删除整个ACL。

仅基本ACL与基本ACL6，以及高级ACL与高级ACL6，可以使用相同的ACL名称；其他类型ACL之间，不能使用相同的ACL名称。

基于对 IPv4 和 IPv6 支持情况的划分

划分如下：

- ACL4：通常直接叫做“ACL”，特指仅支持过滤IPv4报文的ACL。
- ACL6：又叫做“IPv6 ACL”，特指仅支持过滤IPv6报文的ACL。

以上两种ACL，以及既支持过滤IPv4报文又支持过滤IPv6报文的ACL，统一称做“ACL”。各类型ACL对IPv4和IPv6的支持情况，如**表1-1**所示。

基于 ACL 规则定义方式的划分

表1-1所示，基于ACL规则定义方式的划分如下。

表 1-1 基于 ACL 规则定义方式的 ACL 分类

分类	适用的IP版本	规则定义描述	编号范围
基本ACL	IPv4	仅使用报文的**源IP地址**、分片信息和生效时间段信息来定义规则。	2000～2999
高级ACL	IPv4	既可使用IPv4报文的**源IP地址**，也可使用**目的IP地址**、IP协议类型、ICMP类型、TCP源/目的端口、UDP源/目的端口号、生效时间段等来定义规则。	3000～3999
二层ACL	IPv4&IPv6	使用报文的**以太网帧头信息**来定义规则，如根据源MAC（Media Access Control）地址、目的MAC地址、二层协议类型等。	4000～4999
用户自定义ACL	IPv4&IPv6	使用**报文头、偏移位置、字符串掩码和用户自定义字符串**来定义规则，即以报文头为基准，指定从报文的第几个字节开始与字符串掩码进行“与”操作，并将提取出的字符串与用户自定义的字符串进行比较，从而过滤出相匹配的报文。	5000～5999
用户ACL	IPv4	既可使用IPv4报文的**源IP地址**，也可使用**目的IP地址**、IP协议类型、ICMP类型、TCP源端口/目的端口、UDP源端口/目的端口号等来定义规则。	6000～6031
基本ACL6	IPv6	可使用IPv6报文的**源IPv6地址**、分片信息和生效时间段来定义规则。	2000～2999
高级ACL6	IPv6	可以使用IPv6报文的**源IPv6地址**、**目的IPv6地址**、IPv6协议类型、ICMPv6类型、TCP源/目的端口、UDP源/目的端口号、生效时间段等来定义规则。	3000～3999

1.5 ACL 的步长设定

步长的含义

步长，是指系统自动为ACL规则分配编号时，每个相邻规则编号之间的差值。

系统为ACL中首条未手工指定编号的规则分配编号时，使用步长值作为该规则的起始编号；为后续规则分配编号时，则使用大于当前ACL内最大规则编号且是步长整数倍的最小整数作为规则编号。例如ACL中包含规则rule 5和rule 12，ACL（特指基本ACL、高

级ACL、二层ACL、用户ACL）的缺省步长为5，大于12且是5的倍数的最小整数是15，所以系统分配给新配置的规则的编号为15。

```
[Huawei-acl-basic-2001] display this
#
acl number 2001              //空ACL
#
return
[Huawei-acl-basic-2001] rule deny source 10.1.1.0 0.0.0.255 //配置首条不指定规则编号的规则
[Huawei-acl-basic-2001] display this
#
acl number 2001
 rule 5 deny source 10.1.1.0 0.0.0.255
#
return
[Huawei-acl-basic-2001] rule 12 deny source 10.2.2.0 0.0.0.255 //配置一条规则编号为12的规则
[Huawei-acl-basic-2001] display this
#
acl number 2001
 rule 5 deny source 10.1.1.0 0.0.0.255
 rule 12 deny source 10.2.2.0 0.0.0.255
#
return
[Huawei-acl-basic-2001] rule deny source 10.3.3.0 0.0.0.255 //再次配置一条不指定规则编号的规则
[Huawei-acl-basic-2001] display this
#
acl number 2001
 rule 5 deny source 10.1.1.0 0.0.0.255
 rule 12 deny source 10.2.2.0 0.0.0.255
rule 15 deny source 10.3.3.0 0.0.0.255
#
return
```

如果重新调整了步长值（例如调整为2），系统则会自动从当前步长值开始重新排列规则编号，规则编号变成2、4、6…。恢复步长值为缺省值后，系统则会立刻按照缺省步长重新调整规则编号，规则编号变成5、10、15…。

```
[Huawei-acl-basic-2001] display acl 2001
Basic ACL 2001, 3 rules
Acl's step is 5
 rule 5 deny source 10.1.1.0 0.0.0.255
 rule 12 deny source 10.2.2.0 0.0.0.255
 rule 15 deny source 10.3.3.0 0.0.0.255

[Huawei-acl-basic-2001] step 2   //配置步长值为2
[Huawei-acl-basic-2001] display acl 2001
Basic ACL 2001, 3 rules
Acl's step is 2
 rule 2 deny source 10.1.1.0 0.0.0.255
 rule 4 deny source 10.2.2.0 0.0.0.255
 rule 6 deny source 10.3.3.0 0.0.0.255

[Huawei-acl-basic-2001] undo step   //恢复步长值为缺省值
[Huawei-acl-basic-2001] display acl 2001
Basic ACL 2001, 3 rules
Acl's step is 5
 rule 5 deny source 10.1.1.0 0.0.0.255
 rule 10 deny source 10.2.2.0 0.0.0.255
 rule 15 deny source 10.3.3.0 0.0.0.255
```

步长的作用

设置步长的作用，在于方便后续在旧规则之间插入新的规则。

假设，一条ACL中，已包含了三条规则rule 5、rule 10、rule 15。如果希望源IP地址为10.1.1.3的报文也被拒绝通过，该如何处理？

```
rule 5 deny source 10.1.1.1 0  //表示拒绝源IP地址为10.1.1.1的报文通过
rule 10 deny source 10.1.1.2 0 //表示拒绝源IP地址为10.1.1.2的报文通过
rule 15 permit source 10.1.1.0 0.0.0.255 //表示允许源IP地址为10.1.1.0/24网段地址的报文通过
```

由于ACL匹配报文时遵循“**一旦命中即停止匹配**”的原则，所以源IP地址为10.1.1.1和10.1.1.2的报文，会在匹配上编号较小的rule 5和rule 10后停止匹配，从而被系统拒绝通过；而源IP地址为10.1.1.3的报文，则只会命中rule 15，从而得到系统允许通过。若想让源IP地址为10.1.1.3的报文也被拒绝通过，则必须为该报文配置一条新的deny规则。可以在rule 15之前插入一条新规则rule 11，这样源IP地址为10.1.1.3的报文，就会因先命中rule 11而被系统拒绝通过。插入rule 11后，该ACL的旧规则编号不受影响，且新的规则排序为rule 5、rule 10、rule 11、rule 15。

```
rule 5 deny source 10.1.1.1 0  //表示禁止源IP地址为10.1.1.1的报文通过
rule 10 deny source 10.1.1.2 0 //表示禁止源IP地址为10.1.1.2的报文通过
rule 11 deny source 10.1.1.3 0 //表示拒绝源IP地址为10.1.1.3的报文通过
rule 15 permit source 10.1.1.0 0.0.0.255 //表示允许源IP地址为10.1.1.0网段地址的报文通过
```

试想一下，如果这条ACL的规则间隔不是5，而是1（rule 1、rule 2、rule 3…），这时再想插入新的规则，就只能先删除已有的规则，然后再配置新规则，最后将之前删除的规则重新配置还原。

因此，为了避免上述操作造成的麻烦，ACL引入了步长的概念。通过设置ACL步长，使规则之间留有一定的空间，就可以轻松的在旧规则中插入新规则了。

1.6 ACL 的匹配顺序

一条ACL可以由多条“deny | permit”语句组成，每一条语句描述一条规则，这些规则可能存在重复或矛盾的地方。例如，在一条ACL中先后配置以下两条规则：

```
rule deny ip destination 10.1.0.0 0.0.255.255   //表示拒绝目的IP地址为10.1.0.0/16网段地址的报文通过
rule permit ip destination 10.1.1.0 0.0.0.255   //表示允许目的IP地址为10.1.1.0/24网段地址的报文通过，该网段地址范围小于10.1.0.0/16网段范围
```

其中，permit规则与deny规则是相互矛盾的。对于目的IP=10.1.1.1的报文，如果系统先将deny规则与其匹配，则该报文会被拒绝通过。相反，如果系统先将permit规则与其匹配，则该报文会得到允许通过。

因此，对于规则之间存在重复或矛盾的情形，报文的匹配结果与ACL的匹配顺序是息息相关的。

设备支持两种ACL匹配顺序：配置顺序（**config**模式）和自动排序（**auto**模式）。缺省的ACL匹配顺序是**config**模式。

配置顺序

配置顺序，即系统按照ACL规则编号从小到大的顺序进行报文匹配，规则编号越小越容易被匹配。

- 如果配置规则时指定了规则编号，则规则编号越小，规则插入位置越靠前，该规则越先被匹配。
- 如果配置规则时未指定规则编号，则由系统自动为其分配一个编号。该编号是一个大于当前ACL内最大规则编号且是步长整数倍的最小整数，因此该规则会被最后匹配。

自动排序

自动排序，是指系统使用“深度优先”的原则，将规则按照精确度从高到低进行排序，并按照精确度从高到低的顺序进行报文匹配。规则中定义的匹配项限制越严格，

规则的精确度就越高，即优先级越高，系统越先匹配。各类ACL的“深度优先”顺序匹配原则如**表1-2**所示。

关于**表1-2**中提到的IP地址通配符掩码、IP协议承载的协议类型、TCP/UDP端口号、二层协议类型通配符掩码、MAC地址通配符掩码等ACL匹配项的详细介绍，请参见**1.8 ACL的常用匹配项**。

表 1-2 “深度优先”匹配原则

ACL类型	匹配原则
基本ACL&ACL6	1. 先看规则中是否带VPN实例，带VPN实例的规则优先。 2. 再比较源IP地址范围，源IP地址范围小（IP地址通配符掩码中“0”位的数量多）的规则优先。 3. 如果源IP地址范围相同，则规则编号小的优先。
高级ACL&ACL6	1. 先看规则中是否带VPN实例，带VPN实例的规则优先。 2. 再比较协议范围，指定了IP协议承载的协议类型的规则优先。 3. 如果协议范围相同，则比较源IP地址范围，源IP地址范围小（IP地址通配符掩码中“0”位的数量多）的规则优先。 4. 如果协议范围、源IP地址范围相同，则比较目的IP地址范围，目的IP地址范围小（IP地址通配符掩码中“0”位的数量多）的规则优先。 5. 如果协议范围、源IP地址范围、目的IP地址范围相同，则比较四层端口号（TCP/UDP端口号）范围，四层端口号范围小的规则优先。 6. 如果上述范围都相同，则规则编号小的优先。
二层ACL	1. 先比较二层协议类型通配符掩码，通配符掩码大（协议类型通配符掩码中“1”位的数量多）的规则优先。 2. 如果二层协议类型通配符掩码相同，则比较源MAC地址范围，源MAC地址范围小（MAC地址通配符掩码中“1”位的数量多）的规则优先。 3. 如果源MAC地址范围相同，则比较目的MAC地址范围，目的MAC地址范围小（MAC地址通配符掩码中“1”位的数量多）的规则优先。 4. 如果源MAC地址范围、目的MAC地址范围相同，则规则编号小的优先。
用户ACL	1. 先比较协议范围，指定了IP协议承载的协议类型的规则优先。 2. 如果协议范围相同，则比较源IP地址范围。如果规则的源IP地址均为IP网段，则源IP地址范围小（IP地址通配符掩码中“0”位的数量多）的规则优先。 3. 如果协议范围、源IP地址范围相同，则比较目的IP地址范围。如果规则的目的IP地址均为IP网段，则目的IP地址范围小（IP地址通配符掩码中“0”位的数量多）的规则优先。 4. 如果协议范围、源IP地址范围、目的IP地址范围相同，则比较四层端口号（TCP/UDP端口号）范围，四层端口号范围小的规则优先。 5. 如果上述范围都相同，则规则编号小的优先。

在自动排序的ACL中配置规则时，不允许自行指定规则编号。系统能自动识别出该规则在这条ACL中对应的优先级，并为其分配一个适当的规则编号。

例如，在**auto**模式的高级ACL 3001中，先后配置以下两条规则：

```
rule deny ip destination 10.1.0.0 0.0.255.255  //表示拒绝目的IP地址为10.1.0.0/16网段地址的报文通过
rule permit ip destination 10.1.1.0 0.0.0.255  //表示允许目的IP地址为10.1.1.0/24网段地址的报文通过，该网段地址范围小于10.1.0.0/16网段范围
```

两条规则均没有带VPN实例，且协议范围、源IP地址范围相同，所以根据**表1-2**中高级ACL的深度优先匹配原则，接下来需要进一步比较规则的目的IP地址范围。由于permit规则指定的目的地址范围小于deny规则，所以permit规则的精确度更高，系统为其分配的规则编号更小。配置完上述两条规则后，ACL 3001的规则排序如下：

```
#
acl number 3001 match-order auto
 rule 5 permit ip destination 10.1.1.0 0.0.0.255
 rule 10 deny ip destination 10.1.0.0 0.0.255.255
#
```

此时，如果再插入一条新的规则rule deny ip destination 10.1.1.1 0（目的IP地址范围是主机地址，优先级高于以上两条规则），则系统将按照规则的优先级关系，重新为各规则分配编号。插入新规则后，ACL 3001新的规则排序如下：

```
#
acl number 3001 match-order auto
rule 5 deny ip destination 10.1.1.1 0
 rule 10 permit ip destination 10.1.1.0 0.0.0.255
 rule 15 deny ip destination 10.1.0.0 0.0.255.255
#
```

相比**config**模式的ACL，**auto**模式ACL的规则匹配顺序更为复杂，但是**auto**模式ACL有其独特的应用场景。例如，在网络部署初始阶段，为了保证网络安全性，管理员定义了较大的ACL匹配范围，用于丢弃不可信网段范围的所有IP报文。随着时间的推移，实际应用中需要允许这个大范围中某些特征的报文通过。此时，如果管理员采用的是**auto**模式，则只需要定义新的ACL规则，无需再考虑如何对这些规则进行排序避免报文被误丢弃。

1.7 ACL 的生效时间段

产生背景

ACL定义了丰富的匹配项，可以满足大部分的报文过滤需求。但需求是不断变化发展的，新的需求总是不断涌现。例如，某公司要求，在上班时间只允许员工浏览与工作相关的几个网站，下班或周末时间才可以访问其他互联网网站；再如，在每天20:00～22:00的网络流量的高峰期，为防止P2P、下载类业务占用大量带宽对其他数据业务的正常使用造成影响，需要对P2P、下载类业务的带宽进行限制。

基于时间的ACL过滤就是用来解决上述问题的。管理员可以根据网络访问行为的要求和网络的拥塞情况，配置一个或多个ACL生效时间段，然后在ACL规则中引用该时间段，从而实现在不同的时间段设置不同的策略，达到网络优化的目的。

生效时间段模式

在ACL规则中引用的生效时间段存在两种模式：

- 第一种模式——周期时间段：以星期为参数来定义时间范围，表示规则以一周为周期（如每周一的8至12点）循环生效。

  格式：**time-range** *time-name start-time* **to** *end-time* { *days* } &<1-7>

  - *time-name*：时间段名称，以英文字母开头的字符串。

  - *start-time* **to** *end-time*：开始时间和结束时间。格式为[小时:分钟] to [小时:分钟]。
  - *days*：有多种表达方式。
    - **Mon**、**Tue**、**Wed**、**Thu**、**Fri**、**Sat**、**Sun**中的一个或者几个的组合，也可以用数字表达，0表示星期日，1表示星期一，……6表示星期六。
    - **working-day**：从星期一到星期五，五天。
    - **daily**：包括一周七天。
    - **off-day**：包括星期六和星期日，两天。
- 第二种模式——绝对时间段：从某年某月某日的某一时间开始，到某年某月某日的某一时间结束，表示规则在这段时间范围内生效。

  格式：**time-range** *time-name* **from** *time1 date1* [**to** *time2 date2*]
  - *time-name*：时间段名称，以英文字母开头的字符串。
  - *time1*/*time2*：格式为[小时:分钟]。
  - *date1*/*date2*：格式为[YYYY/MM/DD]，表示年/月/日。

可以使用同一名称（*time-name*）配置内容不同的多条时间段，配置的各周期时间段之间以及各绝对时间段之间的交集将成为最终生效的时间范围。

例如，在ACL 2001中引用了时间段“test”，“test”包含了三个生效时间段：

```
#
time-range test 8:00 to 18:00 working-day
time-range test 14:00 to 18:00 off-day
time-range test from 00:00 2014/01/01 to 23:59 2014/12/31
#
acl number 2001
 rule 5 permit time-range test
```

- 第一个时间段，表示在周一到周五每天8:00到18:00生效，这是一个周期时间段。
- 第二个时间段，表示在周六、周日下午14:00到18:00生效，这是一个周期时间段。
- 第三个时间段，表示从2014年1月1日00:00起到2014年12月31日23:59生效，这是一个绝对时间段。

时间段“test”最终描述的时间范围为：2014年的周一到周五每天8:00到18:00以及周六和周日下午14:00到18:00。

1.8 ACL 的常用匹配项

设备支持的ACL匹配项种类非常丰富，其中最常用的匹配项包括以下几种。

生效时间段

格式：**time-range** *time-name*

所有ACL均支持根据生效时间段过滤报文。关于生效时间段的详细介绍，请参见**1.7 ACL的生效时间段**。

IP 承载的协议类型

格式：*protocol-number* | **icmp** | **tcp** | **udp** | **gre** | **igmp** | **ip** | **ipinip** | **ospf**

高级ACL支持基于协议类型过滤报文。常用的协议类型包括：ICMP（协议号1）、TCP（协议号6）、UDP（协议号17）、GRE（协议号47）、IGMP（协议号2）、IP（指任何IP层协议）、IPinIP（协议号4）、OSPF（协议号89）。协议号的取值可以是1～255。

例如，当设备某个接口下的用户存在大量的攻击者时，如果希望能够禁止这个接口下的所有用户接入网络，则可以通过指定协议类型为IP来屏蔽这些用户的IP流量来达到目的。配置如下：

```
rule deny ip //表示拒绝IP报文通过
```

再如，设备上打开透明防火墙功能后，在缺省情况下，透明防火墙会在域间丢弃所有入域间的报文，包括业务报文和协议报文。如果希望像OSPF这样的动态路由协议报文能正常通过防火墙，保证路由互通，这时，通过指定协议类型为OSPF即可解决问题。

```
rule permit ospf  //表示允许OSPF报文通过
```

源/目的 IP 地址及其通配符掩码

源IP地址及其通配符掩码格式：**source** { *source-address source-wildcard* | **any** }

目的IP地址及其通配符掩码格式：**destination** { *destination-address destination-wildcard* | **any** }

基本ACL支持根据源IP地址过滤报文，高级ACL不仅支持源IP地址，还支持根据目的IP地址过滤报文。

将源/目的IP地址定义为规则匹配项时，需要在源/目的IP地址字段后面同时指定通配符掩码，用来与源/目的IP地址字段共同确定一个地址范围。

IP地址通配符掩码与IP地址的反向子网掩码类似，也是一个32比特位的数字字符串，用于指示IP地址中的哪些位将被检查。各比特位中，“0”表示“检查相应的位”，“1”表示“不检查相应的位”，概括为一句话就是“检查0，忽略1”。但与IP地址子网掩码不同的是，子网掩码中的“0”和“1”要求必须连续，而通配符掩码中的“0”和“1”可以不连续。

通配符掩码可以为0，相当于0.0.0.0，表示源/目的地址为主机地址；也可以为255.255.255.255，表示任意IP地址，相当于指定**any**参数。

举一个IP地址通配符掩码的示例，当希望来自192.168.1.0/24网段的所有IP报文都能够通过，可以配置如下规则：

```
rule 5 permit ip source 192.168.1.0 0.0.0.255
```

规则中的通配符掩码为0.0.0.255，表示只需检查IP地址的前三组二进制八位数对应的比特位。因此，如果报文源IP地址的前24个比特位与参照地址的前24个比特位（192.168.1）相同，即报文的源IP地址是192.168.1.0/24网段的地址，则允许该报文通过。**表1-3**展示了该例的地址范围计算过程。

表 1-3 通配符掩码示例

项目	十进制等价值	二进制等价值
参照地址	192.168.1.0	11000000.10101000.00000001.00000000
通配符掩码	0.0.0.255	00000000.00000000.00000000.**11111111**

项目	十进制等价值	二进制等价值
确定的地址范围	192.168.1.* *表示0～255之间的整数	11000000.10101000.00000001.**xxxxxxxx** x既可以是0，也可以是1

更多的IP地址与通配符掩码共同确定的地址范围示例，详见**表1-4**。

表 1-4 IP 地址与通配符掩码共同确定的地址范围

IP地址	IP地址通配符掩码	确定的地址范围
0.0.0.0	255.255.255.255	任意IP地址
172.18.0.0	0.0.255.255	172.18.0.0/16网段的IP地址
172.18.5.2	0.0.0.0	仅172.18.5.2这一个主机地址
172.18.8.0	0.0.0.7	172.18.8.0/29网段的IP地址
172.18.8.8	0.0.0.7	172.18.8.8/29网段的IP地址
10.1.2.0	0.0.254.255（通配符掩码中的1和0不连续）	10.1.0.0/24～10.1.254.0/24网段之间且第三个字节为偶数的IP地址，如10.1.0.0/24、10.1.2.0/24、10.1.4.0/24、10.1.6.0/24等。

源/目的 MAC 地址及其通配符掩码

源MAC地址及其通配符掩码格式：**source-mac** *source-mac-address* [*source-mac-mask*]

目的地址及其通配符掩码格式：**destination-mac** *dest-mac-address* [*dest-mac-mask*]

仅二层ACL支持基于源/目的MAC地址过滤报文。

将源/目的MAC地址定义为规则匹配项时，可以在源/目的MAC地址字段后面同时指定通配符掩码，用来与源/目的MAC地址字段共同确定一个地址范围。

MAC地址通配符掩码的格式与MAC地址相同，采用十六进制数表示，共六个字节（48位），用于指示MAC地址中的哪些位将被检查。与IP地址通配符掩码不同的是，MAC地址通配符掩码各比特位中，1表示“检查相应的位”，0表示“不检查相应的位”。如果不指定通配符掩码，则默认掩码为ffff-ffff-ffff，表示检查MAC地址的每一位。

MAC地址与通配符掩码共同确定的地址范围示例，如**表1-5**所示。

表 1-5 MAC 地址与通配符掩码共同确定的地址范围

MAC地址	MAC地址通配符掩码	确定的地址范围
00e0-fc01-0101	0000-0000-0000	任意MAC地址
00e0-fc01-0101	ffff-ffff-ffff	仅00e0-fc01-0101这一个MAC地址
00e0-fc01-0101	ffff-ffff-0000	00e0-fc01-0000～00e0-fc01-ffff

VLAN 编号及其掩码

外层VLAN及其掩码格式：**vlan-id** *vlan-id* [*vlan-id-mask*]

内层VLAN及其掩码格式：**cvlan-id** *cvlan-id* [*cvlan-id-mask*]

二层ACL支持基于外层VLAN或内层VLAN编号过滤报文。

将VLAN编号定义为规则匹配项时，可以在VLAN编号字段后面同时指定VLAN掩码，用来与VLAN编号字段共同确定一个VLAN范围。

VLAN掩码的格式是十六进制形式，取值范围是0x0～0xFFF。如果不指定VLAN掩码，则默认掩码为0xFFF，表示检查VLAN编号的每一位。

VLAN编号与掩码共同确定的VLAN范围示例，如表1-6所示。

表 1-6 VLAN 编号及其掩码共同确定的 VLAN 范围

VLAN编号	VLAN掩码	确定的VLAN范围
10	0x000	任意VLAN
10	0xFFF	仅VLAN 10
10	0xFF0	VLAN 1～VLAN 10

TCP/UDP 端口号

源端口号格式：**source-port** { **eq** *port* | **gt** *port* | **lt** *port* | **range** *port-start port-end* }

目的端口号格式：**destination-port** { **eq** *port* | **gt** *port* | **lt** *port* | **range** *port-start port-end* }

在高级ACL中，当协议类型指定为TCP或UDP时，设备支持基于TCP/UDP的源/目的端口号过滤报文。

其中，TCP/UDP端口号的比较符含义如下：

- **eq** *port*：指定等于源/目的端口。
- **gt** *port*：指定大于源/目的端口。

- **lt** *port*：指定小于源/目的端口。
- **range** *port-start port-end*：指定源/目的端口的范围。*port-start*是端口范围的起始，*port-end*是端口范围的结束。

TCP/UDP端口号可以使用数字表示，也可以用字符串（助记符）表示。例如，**rule deny tcp destination-port eq 80**，可以用**rule deny tcp destination-port eq www**替代。常见TCP端口号及对应的字符串如表1-7所示，常见UDP端口号及对应的字符串如表1-8所示。

表 1-7 常见 TCP 端口号及对应的字符串

端口号	字符串	协议	说明
7	echo	Echo	Echo服务
9	discard	Discard	用于连接测试的空服务
13	daytime	Daytime	给请求主机发送日期和时间
19	CHARgen	Character generator	字符生成服务；发送无止境的字符流
20	ftp-data	FTP data connections	FTP数据端口
21	ftp	File Transfer Protocol(FTP)	文件传输协议（FTP）端口
23	telnet	Telnet	Telnet服务
25	smtp	Simple Mail Transport Protocol (SMTP)	简单邮件传输协议
37	time	Time	时间协议
43	whois	Nickname（WHOIS）	目录服务
49	tacacs	TAC Access Control System (TACACS)	用于基于TCP/IP验证和访问的访问控制系统（TACACS登录主机协议）
53	domain	Domain Name Service (DNS)	域名服务
70	gopher	Gopher	信息检索协议（互联网文档搜寻和检索）
79	finger	Finger	用于用户联系信息的Finger服务，查询远程主机在线用户等信息

端口号	字符串	协议	说明
80	www	World Wide Web (HTTP)	用于万维网（WWW）服务的超文本传输协议（HTTP），用于网页浏览
101	hostname	NIC hostname server	NIC机器上的主机名服务
109	pop2	Post Office Protocol v2	邮件协议-版本2
110	pop3	Post Office Protocol v3	邮件协议-版本3
111	sunrpc	Sun Remote Procedure Call (RPC)	SUN公司的远程过程调用（RPC）协议，用于远程命令执行，被网络文件系统（NFS）使用
119	nntp	Network News Transport Protocol (NNTP)	网络新闻传输协议，承载USENET通信
179	bgp	Border Gateway Protocol (BGP)	边界网关协议
194	irc	Internet Relay Chat (IRC)	互联网中继聊天（多线交谈协议）
512	exec	Exec (rsh)	用于对远程执行的进程进行验证
513	login	Login (rlogin)	远程登录
514	cmd	Remote commands	远程命令，不必登录的远程shell（rshell）和远程复制（rcp）
515	lpd	Printer service	打印机（lpr）假脱机
517	talk	Talk	远程对话服务和客户
540	uucp	Unix-to-Unix Copy Program	Unix到Unix复制服务
543	klogin	Kerberos login	Kerberos版本5（v5）远程登录
544	kshell	Kerberos shell	Kerberos版本5（v5）远程shell

表 1-8 常见 UDP 端口号及对应的字符串

端口号	字符串	协议	说明
7	echo	Echo	Echo服务
9	discard	Discard	用于连接测试的空服务
37	time	Time	时间协议
42	nameserver	Host Name Server	主机名服务
53	dns	Domain Name Service (DNS)	域名服务
65	tacacs-ds	TACACS-Database Service	TACACS数据库服务
67	bootps	Bootstrap Protocol Server	引导程序协议（BOOTP）服务端，DHCP服务使用
68	bootpc	Bootstrap Protocol Client	引导程序协议（BOOTP）客户端，DHCP客户使用
69	tftp	Trivial File Transfer Protocol (TFTP)	小文件传输协议
90	dnsix	DNSIX Security Attribute Token Map	DNSIX安全属性标记图
111	sunrpc	SUN Remote Procedure Call (SUN RPC)	SUN公司的远程过程调用（RPC）协议，用于远程命令执行，被网络文件系统（NFS）使用
123	ntp	Network Time Protocol (NTP)	网络时间协议，蠕虫病毒会利用
137	netbios-ns	NETBIOS Name Service	NETBIOS名称服务
138	netbios-dgm	NETBIOS Datagram Service	NETBIOS数据报服务
139	netbios-ssn	NETBIOS Session Service	NETBIOS会话服务
161	snmp	SNMP	简单网络管理协议
162	snmptrap	SNMPTRAP	SNMP陷阱
177	xdmcp	X Display Manager Control Protocol (XDMCP)	X显示管理器控制协议

端口号	字符串	协议	说明
434	mobilip-ag	MobileIP-Agent	移动IP代理
435	mobilip-mn	MobileIP-MN	移动IP管理
512	biff	Mail notify	异步邮件，可用来通知用户有邮件到达
513	who	Who	登录的用户列表
514	syslog	Syslog	UNIX系统日志服务
517	talk	Talk	远程对话服务器和客户端
520	rip	Routing Information Protocol	RIP路由协议

TCP 标志信息

格式：**tcp-flag** { **ack** | **established** | **fin** | **psh** | **rst** | **syn** | **urg** }*

在高级ACL中，当协议类型指定为TCP时，设备支持基于TCP标志信息过滤报文。

TCP报文头有6个标志位：

- URG(100000)：标识紧急指针有效
- ACK(010000)：标识确认序号有效
- PSH(001000)：标识接收方应该尽快将这个报文段上交给应用层
- RST(000100)：标识重建连接
- SYN(000010)：同步序号，用来发起一个连接
- FIN(000001)：标识发送方完成发送任务

TCP标志信息中的**established**，表示标志位为ACK(010000)或RST(000100)。

指定**tcp-flag**的ACL规则可以用来实现单向访问控制。假设，要求192.168.1.0/24网段用户可以主动访问192.168.2.0/24网段用户，但反过来192.168.2.0/24网段用户不能主动访问192.168.1.0/24。可通过在设备上连接192.168.2.0/24网段的接口入方向上，应用ACL规则来实现该需求。

由TCP建立连接和关闭连接的过程可知，只有在TCP中间连接过程的报文才会ACK=1或者RST=1。根据这个特点，配置如下两种ACL规则，允许TCP中间连接过程的报文通过，拒绝该网段的其他TCP报文通过，就可以限制192.168.2.0/24网段主动发起的TCP连接。

- 类型一：配置指定**ack**和**rst**参数的ACL规则
  ```
  rule 5 permit tcp source 192.168.2.0 0.0.0.255 tcp-flag ack  //允许ACK=1的TCP报文通过
  rule 10 permit tcp source 192.168.2.0 0.0.0.255 tcp-flag rst  //允许RST=1的TCP报文通过
  rule 15 deny tcp source 192.168.2.0 0.0.0.255  //拒绝该网段的其他TCP报文通过
  ```
- 类型二：配置指定**established**参数的ACL规则
  ```
  rule permit tcp source 192.168.2.0 0.0.0.255 tcp-flag established  // established表示ACK=1或者RST=1，表示允许TCP中间连接过程的报文通过
  rule deny tcp source 192.168.2.0 0.0.0.255     //拒绝该网段的其他TCP报文通过
  ```


IP 分片信息

格式：**none-first-fragment**

基本ACL和高级ACL支持基于IP分片信息过滤报文。

IP分片除了首片报文外，还有后续分片报文，又叫做非首片分片报文。仅首片分片报文携带四层信息（如TCP/UDP端口号等），后续分片报文均不携带。网络设备收到分片报文后，会判断其是否是最后一个分片报文。如果不是，则为其分配内存空间，以便于最后一个分片报文到达后完成重组。黑客可以利用这一点，向接收方设备发起分片报文攻击，始终不向接收方发送最后一个分片报文，使得接收方的内存得不到及时释放（接收方会启动一个分片重组的定时器，在定时器超时前如果无法完成重组，将向发送方发送ICMP重组超时差错报文；如果定时器超时后仍未完成重组，则丢弃已存储的分片报文）。在分片报文发送数量很多并且发送速度很快的情况下，接收方的内存很容易被占满，从而导致接收方没有足够的内存资源处理其他正常的业务。

为了解决这个问题，可以配置指定**none-first-fragment**匹配项的ACL规则来阻塞非首片分片报文，从而达到防范分片报文攻击的目的。

针对非分片报文、首片分片报文、非首片分片报文这三类报文，ACL的处理方式如**表 1-9**所示。

表 1-9 ACL 对 IP 分片报文的处理方式

规则包含的匹配项	非分片报文	首片分片报文	非首片分片报文
三层信息（如源/目的IP地址）	三层信息匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配	三层信息匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配	三层信息匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配
三层信息 + 四层信息（如TCP/UDP端口号）	三层和四层信息都匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配	三层和四层信息都匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配	不匹配，转下一条规则进行匹配
三层信息 + **none-first-fragment**	不匹配，转下一条规则进行匹配	不匹配，转下一条规则进行匹配	三层信息匹配上，则返回匹配结果（permit/deny）；未匹配上，则转下一条规则进行匹配

例如，ACL 3012中存在以下规则：

```
#
acl number 3012
 rule 5 deny tcp destination 192.168.2.2 0 none-first-fragment
 rule 10 permit tcp destination 192.168.2.2 0 destination-port eq www
 rule 15 deny ip
#
```


- 该报文是非分片报文或首片分片报文时：如果该报文的目的端口号是80（www对应的端口号是80），则报文与rule 10匹配，报文被允许通过；如果该报文的目的端口号不是80，则报文与rule 15匹配，报文被拒绝通过。
- 该报文是非首片分片报文时：该报文与rule 5匹配，报文被拒绝通过。

1.9 ACL 的常用配置原则

配置ACL规则时，可以遵循以下原则：

1. 如果配置的ACL规则存在包含关系，应注意严格条件的规则编号需要排序靠前，宽松条件的规则编号需要排序靠后，避免报文因命中宽松条件的规则而停止往下继续匹配，从而使其无法命中严格条件的规则。
2. 根据各业务模块ACL默认动作的不同，ACL的配置原则也不同。例如，在默认动作为permit的业务模块中，如果只希望deny部分IP地址的报文，只需配置具体IP地址的deny规则，结尾无需添加任意IP地址的permit规则；而默认动作为deny的业务模块恰与其相反。详细的ACL常用配置原则，如表1-10所示。

说明

以下rule的表达方式仅是示意形式，实际配置方法请参考各类ACL规则的命令行格式。

- **rule permit xxx**/**rule permit xxxx**：表示允许指定的报文通过，xxx/xxxx表示指定报文的标识，可以是源IP地址、源MAC地址、生效时间段等。xxxx表示的范围与xxx表示的范围是包含关系，例如xxx是某一个IP地址，xxxx可以是该IP地址所在的网段地址或**any**（表示任意IP地址）；再如xxx是周六的某一个时段，xxxx可以是双休日全天时间或一周七天全部时间。
- **rule deny xxx**/**rule deny xxxx**：表示拒绝指定的报文通过。
- **rule permit**：表示允许所有报文通过。
- **rule deny**：表示拒绝所有报文通过。

表 1-10 ACL 的常用配置原则

业务模块的ACL默认动作	permit所有报文	deny所有报文	permit少部分报文，deny大部分报文	deny少部分报文，permit大部分报文
permit	无需应用ACL	配置**rule deny**	需先配置**rule permit xxx**，再配置**rule deny xxxx**或**rule deny** **说明** 以上原则适用于报文过滤的情形。当ACL应用于流策略中进行流量监管或者流量统计时，如果仅希望对指定的报文进行限速或统计，则只需配置**rule permit xxx**。	只需配置**rule deny xxx**，无需再配置**rule permit xxxx**或**rule permit** **说明** 如果配置**rule permit**并在流策略中应用ACL，且该流策略的流行为behavior配置为**deny**，则设备会拒绝所有报文通过，导致全部业务中断。

业务模块的ACL默认动作	permit所有报文	deny所有报文	permit少部分报文，deny大部分报文	deny少部分报文，permit大部分报文
deny	• 路由和组播模块：需配置**rule permit** • 其他模块：无需应用ACL	• 路由和组播模块：无需应用ACL • 其他模块：需配置**rule deny**	只需配置**rule permit xxx**，无需再配置**rule deny xxxx**或**rule deny**	需先配置**rule deny xxx**，再配置**rule permit xxxx**或**rule permit**

举例：

- 例1：在流策略中应用ACL，使设备对192.168.1.0/24网段的报文进行过滤，拒绝192.168.1.2和192.168.1.3主机地址的报文通过，允许192.168.1.0/24网段的其他地址的报文通过。

  流策略的ACL默认动作为**permit**，该例属于“deny少部分报文，permit大部分报文”的情况，所以只需配置**rule deny xxx**。

  ```
  #
  acl number 2000
   rule 5 deny source 192.168.1.2 0
   rule 10 deny source 192.168.1.3 0
  #
  ```

- 例2：在流策略中应用ACL，使设备对192.168.1.0/24网段的报文进行过滤，允许192.168.1.2和192.168.1.3主机地址的报文通过，拒绝192.168.1.0/24网段的其他地址的报文通过。

  流策略的ACL默认动作为**permit**，该例属于“permit少部分报文，deny大部分报文”的情况，所以需先配置**rule permit xxx**，再配置**rule deny xxxx**。

  ```
  #
  acl number 2000
   rule 5 permit source 192.168.1.2 0
   rule 10 permit source 192.168.1.3 0
   rule 15 deny source 192.168.1.0 0.0.0.255
  #
  ```

- 例3：在Telnet中应用ACL，仅允许管理员主机（IP地址为172.16.105.2）能够Telnet登录设备，其他用户不允许Telnet登录。

  Telnet的ACL默认动作为**deny**，该例属于“permit少部分报文，deny大部分报文”的情况，所以只需配置**rule permit xxx**。

  ```
  #
  acl number 2000
   rule 5 permit source 172.16.105.2 0
  #
  ```

- 例4：在Telnet中应用ACL，不允许某两台主机（IP地址为172.16.105.3和172.16.105.4）Telnet登录设备，其他用户均允许Telnet登录。

  Telnet的ACL默认动作为**deny**，该例属于“deny少部分报文，permit大部分报文”的情况，所以需先配置**rule deny xxx**，再配置**rule permit**。

  ```
  #
  acl number 2000
   rule 5 deny source 172.16.105.3 0
   rule 10 deny source 172.16.105.4 0
  ```



```
 rule 15 permit
#
```

- 例5：在FTP中应用ACL，不允许用户在周六的00:00～8:00期间访问FTP服务器，允许用户在其他任意时间访问FTP服务器。

  FTP的ACL默认动作为**deny**，该例属于“deny少部分报文，permit大部分报文”的情况，所以需先配置**rule deny xxx**，再配置**rule permit xxxx**。

```
#
time-range t1 00:00 to 08:00 Sat
time-range t2 00:00 to 23:59 daily
#
acl number 2000
rule 5 deny time-range t1
rule 10 permit time-range t2
#
```

1.10 相关信息

如果您想了解ACL的更多信息及配置方法，可参考以下产品文档：

AR系列接入路由器 V200R010 ACL配置

S系列交换机 V200R013 ACL配置

CloudEngine 12800, 12800E V200R005C10 ACL配置指南

