某局点 高端路由器 部署IPOE V6触发配置后认证功能失效问题

QoS 认证 **林宇阳** 2023-06-25 发表

组网及说明

高端路由器（如CR16K-F、SR88-X等）部署为园区网接入认证核心设备，对园区内用户提供IPOE接入认证服务。

告警信息

无

问题描述

现场此前已部署IPOE认证功能，接口配置ip subscriber l2-connected enable命令使能双栈认证功能。
但由于此前园区内只要求进行IPv4地址族认证，因此设备的全局QOS中仅匹配了IPv4地址族的ACL，所以此时用户终端的IPv6地址族从BRAS获取地址（或前缀）后在前域上线，即使不进行认证访问IPv6网络资源也不受限制。
基于此情况，园区管理方要求增加IPv6访问的认证控制，于是现场在路由器的全局QOS配置中增加了IPv6地址的匹配条件，要求前域终端的IPv6地址族访问也能被限制。
但实际操作配置修改后，园区内认证功能失效，新接入用户无法触发WEB重定向认证，但无需认证即可直接访问网络。

过程分析

原设备通过全局QOS Policy中的各个CB对实现放通特定流量、重定向WEB界面和阻断其它前域流量功能，以官网配置案例为例：

配置入方向QoS策略web

[Device] qos policy web

为类指定对应的流行为，规则为对于用户组web中的用户：

允许目的地址为Portal服务器和内网服务器IP地址的报文通过；

对于目的端口为80（HTTP报文）和443（HTTPS报文）的报文重定向到CPU；

除上述报文外，其余报文均禁止通过。

[Device-qospolicy-web] classifier web_permit behavior web_permit

[Device-qospolicy-web] classifier neiwang behavior neiwang

[Device-qospolicy-web] classifier web_http behavior web_http

[Device-qospolicy-web] classifier web_https behavior web_https

[Device-qospolicy-web] classifier web_deny behavior web_deny

以其中之一举例，如变更操作前classifier **web_deny**相关配置为：

[Device] traffic classifier web_deny operator and

[Device-classifier-web_deny] if-match acl name ip

[Device-classifier-web_deny] quit

[Device] acl advanced name ip

[Device-acl-ipv4-adv-ip] rule 0 permit ip user-group web

[Device-acl-ipv4-adv-ip] quit

此时QOS能命中所有前域用户组web的非放通或http/https访问IPv4流量进行阻断。

但现场在qos中增加IPv6地址族配置后，classifier **web_deny**变成了

[Device] traffic classifier web_deny operator and

[Device-classifier-web_deny] if-match acl name ip

[Device-classifier-web_deny] if-match acl ipv6 name ip

[Device-classifier-web_deny] quit

这种情况下，由于classifier的操作符为and，即要求“指定类下的规则之间是逻辑与的关系，即数据包必须匹配全部规则才属于该类。”

而显然报文不可能同时既是IPv4报文又是IPv6报文，所以该classifier无法命中任何报文，QOS中其它classifier也同理，失去了命中前域报文的能力。

最终导致通过QOS实现的前域放通特定流量、重定向WEB界面和阻断其它流量功能全部失效。

解决方法

因为流量显然无法同时匹配双栈特征，所以classifier中只需满足任一if-match条件即应视为匹配命中，将Classifier的操作符从“and”改为“or”即可。“or：指定类下的规则之间是逻辑或的关系，即数据包只要匹配其中任何一个规则就属于该类。”

需注意：

1、Traffic Classifier创建时，如未命令指定操作符则缺省是“and”，若该Classifier计划由于双栈场景，建议在创建时就指定为“or”类型。

2、在修改已配置的Traffic Classifier时直接操作覆盖即可实现操作符由and改为or，举例：

```
[SR88X-UP-classifier-3587]dis th
#
traffic classifier 3587 operator and
 if-match acl 3587
#
return
[SR88X-UP-classifier-3587]exit
[SR88X-UP]traffic classifier 3587 operator or
[SR88X-UP-classifier-3587]dis th
#
traffic classifier 3587 operator or
 if-match acl 3587
#
return
```