

# 服务质量 (qos-scripts) 配置

这是UCI配置文件 `/etc/config/qos` 的文档。

警告：在“系统→启动”选项卡中启用 `qos` Initscript 之前，`luci-app-qos` 将不会启动，并且在 `Network→QoS` 下启用 `qos`

## 第

最小的QoS配置通常包括：

- 一个接口部分
- 一些规则将数据包分配到至少两个桶
- 配置的桶。

## 接口

每个接口都可以有自己的缓冲区。该 `interface` 节声明指定接口正在通信的连接的全局特性。本节中定义了以下选项：

### 配置界面 `dsl`

选项启用1

选项类组“默认”

期权开支1

选项上传512

选项下载4096

名称	类型	需要	默认	描述
<code>enabled</code>	布尔	是	1	启用/禁用QoS
<code>classgroup</code>	串	是	Default	指定 <code>classgroup</code> 用于此界面（请参阅下面的 <code>classgroup</code> 说明）
<code>overhead</code>	布尔	是	1	减少上传和下载比例，以防止链路饱和
<code>download</code>	整数	是	4096	下载限制 kilobits/second
<code>upload</code>	整数	是	512	上传限制 kilobits/second

# 规则

每个 `classify` 部分定义一组数据包以及该组所属的目标（即数据桶）。所有数据包共享指定的桶。

名称	类型	需要	默认	描述
<code>target</code>	桶	是	（没有）	四个默认值是： <code>Priority</code> 、 <code>Express</code> 、 <code>Normal</code> 和 <code>Bulk</code>
<code>proto</code>	串	没有	<code>0</code>	与此协议匹配的数据包属于目标中定义的存储桶
<code>srchost</code>	串	没有	（没有）	与此源主机（单个IP或CIDR表示法）匹配的数据包属于目标中定义的存储桶
<code>dsthost</code>	串	没有	（没有）	与此目标主机（单个IP或CIDR表示法）匹配的数据包属于目标中定义的存储桶
<code>ports</code>	整数	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>srcports</code>	整数	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>dstports</code>	整数	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>portrange</code>	整数	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>pktsize</code>	整数	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>tcpflags</code>	串	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>mark</code>	串	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>connbytes</code>	INT	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>tos</code>	串	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>dscp</code>	串	没有	（没有）	与此匹配的数据包属于目标中定义的数据桶
<code>direction</code>	串	没有	（没有）	匹配此流量方向（ <code>in</code> 或 <code>out</code> ）的数据包属于目标中定义的存储桶

注意：已经打破的“`layer7`”选项被r45425删除。

# Classgroup

因为我们可以有一个接口，我们可以有一个以上的类组。

```
config classgroup“Default”  
    选项类“Priority Express Normal Bulk”  
    选项默认“正常”
```

名称	类型	需要	默认	描述
classes	桶名称	是	(没有)	指定类的名称列表
default	桶名称	是	(没有)	定义哪个类被认为是默认的

## 类

每个Bucket都有自己的配置。

例：

```
config类“Normal”  
    选项packetize 1500  
    选项packetdelay 100  
    选项avgrate 10  
    选项优先5
```

名称	类型	需要	默认	描述
packetize	整数	是	(没有)	以字节为单位
packetdelay	整数	是	(没有)	在ms
maxsize	整数	是	(没有)	以字节为单位
avgrate	整数	是	(没有)	该类的平均速率，带宽百分比的值（此值用于计算vaues'Nx' 'tc ... hfsc rt m1 N1 d N2 m2 N3'）
limitrate	整数	没有	100	定义此类限制的可用带宽的百分比，以%
maxsize	整数	是	(没有)	以字节为单位
priority	整数	是	(没有)	在%

## 类（高级用户）

以下是每个 / etc / config / qos 类参数的未经验证的技术细分。资料来源: [http \(http://pastebin.com/YL55na2E\) : //pastebin.com/YL55na2E \(http://pastebin.com/YL55na2E\)](http://pastebin.com/YL55na2E)

```
###参数:
#
#maxsize:
#限制iptables规则中的包大小
#
#avgrate :(注: sum (avgrates) ~100)
#rt m1 = avgrate / sum (avgrate) * max_bandwidth
#rt m2 = avgrate * max_bandwidth / 100
#ls m1 = rt m1
#
#packetize&packetdelay :(仅当avgrate存在时才有效)
#rt d = max (packetdelay, "packetize to transfer"所需的时间) (较小的ps - >较小的d)
#ls d = rt d
#
# 优先:
#ls m2 = priority / sum (priority) * max_bandwidth
#
#limitrate:
#ul rate = limitrate * max_bandwidth / 100
```

## 快速入门指南

### 1. 安装qos-scripts包:

```
opkg安装qos-scripts
```

### 2. 使用UCI命令行的基本配置:

```
uci set qos.wan.upload = 1000 # 以kBits / s上传速度
uci set qos.wan.download = 16000 # 以kBits / s的速度下载
uci set qos.wan.enabled = 1
uci commit qos
```

### 3. 启动并查找错误输出和测试):

```
/etc/init.d/qos启动
```

### 4. 在每次启动时使脚本运行:

```
/etc/init.d/qos启用
```

## 故障排除

通过运行以下命令查看生成的流量控制qdisc设置:

```
tc qdisc
```

任何接口的默认值（即无QoS应用）的值如下所示:

```
qdisc fq_codel 0: dev eth0 root refcnt 2 limit 1024p streams 1024 quantum 300 targ  
et 5.0ms interval 100.0ms ecn
```

任何仅打印单个qdisc行的接口，显示与该行相同的设置（这是针对`dev eth0`），表示该接口没有QoS。

启用QoS的网络接口将打印多个qdisc行，每个对应于QoS类等。

如果打印的qdisc设置似乎不正确，则可以通过运行以下命令预览从OpenWRT / `etc / config / qos`生成的`tc`命令:


```
/usr/lib/qos/generate.sh 界面wan
```

（使用 / `etc / config / qos` 文件中给出的UCI接口名称替换 `wan`）。

这应该打印一系列用于设置QoS子系统的`insmod`和`tc`命令。您可以通过运行以下命令来调试运行这些命令导致的任何错误:

```
/usr/lib/qos/generate.sh interface wan | sh -x
```

的输出`/usr/lib/qos/generate.sh`通常自动执行作为其一部分`/etc/hotplug.d/iface/10-qos`。

 最后修改: 2016/10/21 17:24 通过bobafetthotmail

除非另有说明，本维基的内容将根据以下许可证获得许可: CC Attribution-Share Alike 4.0 International  
(<http://creativecommons.org/licenses/by-sa/4.0/>)