

部署及操作指导

Kilolink 聚合平台 V5(linux 系统)

(2021-12 版)

目录

1 Kilolink 聚合平台部署	1
1.1 服务器环境准备	1
1.2 登录服务器	2
1.3 部署聚合服务	3
第一步：安装 docker 容器	3
第二步：获取聚合部署文件	3
第三步：给文件 “klnkserver.sh” 执行权限	4
第四步：执行文件 “klnkserver.sh” 开始部署聚合服务	4
第五步：配置聚合服务公网 IP 地址	5
1.4 登录验证	6
2 部署常见问题及解决方法	8
2.1 如果部署新版本后无法登录服务器	8
2.2 如果在部署过程中遇到错误提示	8
2.3 在部署 kilolink 过程中，安装 docker 长期未响应。	9
2.4 服务器按照 kilolink 手册部署完成后，使用 IP 加端口无法登录服务器都的 web	9
2.5 客户之前使用旧版本 kilolink，然后升级到新版本的 kilolink 后，P1/P2 升级至最新固件 (4.7.2519) 后依旧无法正常连接 kilolink	10
2.6 安装完成后，登录服务器报以下错误：	10
2.7 使用新版 kiloview 部署成功后，且可以通过 IP+端口登录，但是 P1/P2 无法正确连接到 kilolink 服务器。	10

2.8	Kilolink 如何修改默认登陆 web 端口.....	11
3	kilolink 聚合功能操作指导.....	13
3.1	聚合平台登录.....	13
3.2	设备连接聚合平台.....	14
	第一步：平台添加设备，并生成授权码.....	14
	第二步：设备注册到平台.....	14
3.3	聚合平台管理.....	16
	3.3.1 在线设备.....	16
	3.3.2 设备管理.....	17
	3.3.3 用户列表.....	21
3.4	其他.....	22

1 Kilolink 聚合平台部署

1.1 服务器环境准备

(1) 硬件环境

处理器：英特尔酷睿 i3 CPU 或更高

硬盘：64G 硬盘或更高

内存：4GB RAM 或更高

(2) 软件环境

操作系统：Linux64 位操作系统 (Ubuntu 18.04+ / Debian 9+)

(3) 网络环境

IP 地址：1 个公网 IP 地址

带宽：根据视频编码码率规划，如：编码码率 4Mbps,则需规划 8Mbps 的带宽

端口：服务器需使用以下端口，如果服务器的网络环境中存在防火墙，需开放相应端口

端口	协议
82	TCP
3478	TCP+UDP
60000	UDP
60001	UDP
5000-5100	TCP+UDP
30000-30050	TCP+UDP



注意

- (1) 考虑到服务器的硬件及维护成本，以及后续云平台的版本更新，建议租赁云服务提供商的服务器，如：阿里云、腾讯云、华为云等；
- (2) 服务器将使用多个端口。如果部署期间存在防火墙，则需要打开防火墙中打开相关端口。Kiloview 使用 AES256 加密来确保实时流媒体的安全性。

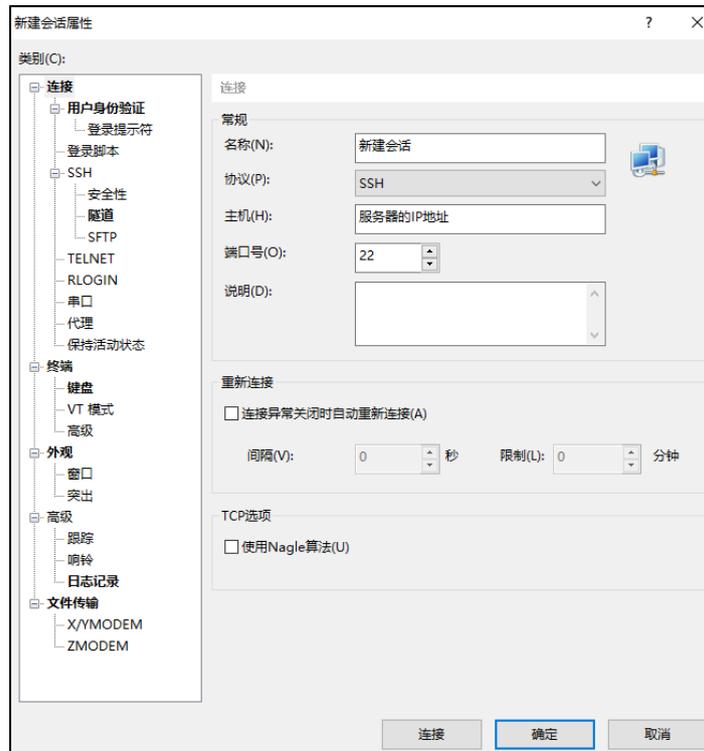
1.2 登录服务器

采用远程终端软件登录服务器，建议采用 Xshell 或者 PuTTY 工具。

Xshell 下载链接：<https://www.netsarang.com/zh/xshell-download/>

PuTTY 下载地址：<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

(1) 下载并安装后，在新建会话属性中输入服务器 IP 地址，采用 SSH 协议通信，端口号默认为 22，输入完成后，点击[确定]按钮；



(2) 在弹出的对话框中输入用户名和密码，普通用户需要 sudo 获取管理权限，或以 root 身份登录。使用命令: sudo su -

1.3 部署聚合服务

第一步：安装 docker 容器

在终端窗口中输入：`curl -fsSL https://get.docker.com | bash`

```
root@kiloview:/# curl -fsSL https://get.docker.com | sh

# Executing docker install script, commit: 93d2499759296ac1f9c510605fef85052a2c32be

+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq apt-transport-https ca-certificates curl >/dev/null
+ sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" | gpg --dearmor --yes -o /usr/share/keyrings/docker-archive-keyring.gpg
+ sh -c echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu focal stable" > /etc/apt/sources.list.d/docker.list
+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq --no-install-recommends docker-ce-cli docker-scanner-plugin docker-ce >/dev/null
+ version_gte 20.10
+ [ -z ]
+ return 0
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq docker-ce-rootless-extras >/dev/null
+ sh -c docker version
Client: Docker Engine - Community
 Version:           20.10.11
 API version:       1.41
 Go version:        go1.16.9
 Git commit:        dea9396
 Built:             Thu Nov 18 00:37:06 2021
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:           20.10.11
  API version:       1.41 (minimum version 1.12)
  Go version:        go1.16.9
  Git commit:        847da18
  Built:             Thu Nov 18 00:35:15 2021
  OS/Arch:           linux/amd64
  Experimental:     false
 containerd:
  Version:           1.4.12
  GitCommit:        7b11cfaabd73bb80907dd23182b9347b4245eb5d
 runc:
  Version:           1.0.2
  GitCommit:        v1.0.2-0-g52b36a2
 docker-init:
  Version:           0.19.0
  GitCommit:        de40ad0
```

第二步：获取聚合部署文件

使用命令：`wget http://firmware.kiloview.com.cn/Tools/.server/klkserver_cn.sh -O`

[klkserver.sh](http://firmware.kiloview.com.cn/Tools/.server/klkserver_cn.sh)

```
root@kiloview:~# wget http://firmware.kiloview.com.cn/Tools/.server/klnkserver_cn.sh -O klnkserver.sh
--2021-12-13 05:36:06-- http://firmware.kiloview.com.cn/Tools/.server/klnkserver_cn.sh
Resolving firmware.kiloview.com.cn (firmware.kiloview.com.cn)... 172.67.141.173, 104.21.46.201, 2606:4700:3031::ac43:8dad, ...
Connecting to firmware.kiloview.com.cn (firmware.kiloview.com.cn)|172.67.141.173|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24506 (24K) [application/octet-stream]
Saving to: 'klnkserver.sh'

klnkserver.sh                               100%
[=====]
=====>] 23.93K  147KB/s  in 0.2s

2021-12-13 05:36:08 (147 KB/s) - 'klnkserver.sh' saved [24506/24506]
```

第三步：给文件“klnkserver.sh”执行权限

使用命令：chmod +x klnkserver.sh

```
root@kiloview:~# chmod +x klnkserver.sh
```

第四步：执行文件“klnkserver.sh”开始部署聚合服务

使用命令：./klnkserver.sh

```
root@kiloview:/# ./klnkserver.sh
Docker 运行中

Kiloview KlnkServer 一键管理脚本 [v0.0.3]
1. 安装 KlnkServer
2. 更新 KlnkServer
3. 卸载 KlnkServer

当前状态：未安装

请输入数字 [1-3]: 1
```



注意

当前状态：显示“未安装”，直接选择数字“1”回车开始安装；显示“已安装并已启动”，如安装方法不是使用该部署方式，则需要对以前部署的进行删除，再按照该方法进行部署。如上次使用了该方法进行部署，则之后只需要选择数字“2”进行更新即可，更新后聚合服务器所有的配置将不会丢失。

第五步：配置聚合服务公网 IP 地址

选择数字“1”开始安装，网络模式可选择“host/bridge”后回车进入下一步；“IP 或域名”填入服务器公网 IP 或者域名，进入下一步开始自动拉去聚合镜像文件进行安装。

出现“创建容器成功”和“运行容器成功”则表示按照完成，可以开始登录聚合平台。

```
root@kiloview:/# ./Klnkserver.sh
Docker 运行中

Kiloview KlnkServer 一键管理脚本 [v0.0.3]
1. 安装 KlnkServer
2. 更新 KlnkServer
3. 卸载 KlnkServer

当前状态：未安装

请输入数字 [1-3]: 1
开始安装
/data/目录已存在
[信息] 开始设置 容器运行方式...
请输入容器网络模式: (host/bridge): host

容器网络模式 : host

[信息] 开始设置 服务器公网IP (设备连接)...
请输入用户配置中要显示的 服务器IP或域名 (当服务器有多个IP时, 可以指定用户配置中显示的IP或者域名)
(默认自动检测外网IP):192.168.22.216

IP或域名 : 192.168.22.216

是否使用默认配置: (Y/n): y
[信息] 选择默认配置!
-----
Web 端口: 82 TCP
聚合平台IP: 192.168.22.216
聚合工作端口: 60000 UDP
```

是否使用默认配置：建议选择“Y”使用默认端口及配置进行安装，如选择“N”，则根据提示对每个端口及配置进行手动输入。

```
是否使用默认配置: (Y/n): y
[信息] 选择默认配置!
-----
Web 端口: 82 TCP
聚合平台IP: 192.168.22.216
聚合工作端口: 60000 UDP
聚合转发服务端口: 60001 UDP
转发/流服务最小端口: 30000 TCP+UDP
转发/流服务最大端口: 30050 TCP+UDP
TUN服务器工作最小端口: 5000 TCP+UDP
TUN服务器工作最大端口: 5100 TCP+UDP
TUN服务器端口: 3478 + 3479 TCP+UDP
TUN服务器用户名: admin
TUN服务器密码: kiloview123456
TUN服务器REALM: demo
-----
```

1.4 登录验证

在浏览器（推荐使用 google）地址栏中输入“服务器 IP 地址:82”，回车后显示聚合服务器的登录界面，默认用户名和密码均为：admin.



注意

- (1) 服务器登录的默认端口号已经调整为 82;
- (2) 支持聚合服务的 P 系列编码器，需更新到 4.7.2519 及以上版本;
- (3) 设备端的端口号仍为 60000.

KILOVIEW®

Login

admin

.....

Login

1.5 聚合服务更新和卸载

执行命令：`./klnkserver.sh`，选择数字“2”开始拉取新的镜像文件更新，更新过程会自动删除旧的容器，并使用新的镜像文件重新安装。更新不会删除服务器上的配置。

```
root@kiloview:~# ./klnkserver.sh
Docker 运行中

Kiloview KlnkServer 一键管理脚本 [v0.0.3]

1. 安装 KlnkServer
2. 更新 KlnkServer
3. 卸载 KlnkServer

当前状态：已安装 并 已启动

请输入数字 [1-3]: 2
[信息] 拉取最新版本 kiloview/klnkserver 镜像...
Using default tag: latest
latest: Pulling from kiloview/klnkserver
Digest: sha256:72c741356f8f5a61f490f50639ca0ac0bf13336d11f06ac57e3b0f6567989feb
Status: Image is up to date for kiloview/klnkserver:latest
docker.io/kiloview/klnkserver:latest
[信息] 删除旧容器...
klnkserver
[信息] 重新运行镜像...
45145cdd51e91eb4cf2896d03d74faaf710b8ac06c8f496ee8464e0c545ef509
[信息] 创建容器成功
klnkserver
[信息] 运行容器成功
```

执行命令：`./klnkserver.sh`，选择数字“3”，开始卸载聚合容器，卸载后聚合服务的所有配置并未被删除，重新安装后，所有配置自动恢复到新的聚合服务页面。

```
root@kiloview:~# ./klnkserver.sh
Docker 运行中

Kiloview KlnkServer 一键管理脚本 [v0.0.3]

1. 安装 KlnkServer
2. 更新 KlnkServer
3. 卸载 KlnkServer

当前状态：已安装 并 已启动

请输入数字 [1-3]: 3
删除容器成功
```



注意

无论使用更新或者是卸载功能，聚合服务的配置信息都不会被删除，重新安装更新后自动恢复。

2 部署常见问题及解决方法

2.1 如果部署新版本后无法登录服务器

解决方法:

采用服务器 IP+端口号, 即服务器 IP:82 登录尝试 (之前的登录端口为 81)

设备注册时填写的端口仍然为 60000

建议通过 history 命令查看执行过的命令, 以更好的分析问题, 查询具体命令, 可通过 history |grep 命令进行查询。

2.2 如果在部署过程中遇到错误提示

解决方法:

请先检查操作系统版本, 目前仅支持 Linux64 位操作系统 Linux64-bit operating system (Ubuntu 18.04+ / Debian 9+)

(1) 查询 linux 位数: getconf LONG_BIT

```
Last login: Wed Oct 13 21:13:00 2021 from  
ubuntu@VM-4-5-ubuntu:~$ getconf LONG_BIT  
64  
ubuntu@VM-4-5-ubuntu:~$
```

(2) 查询 linux 版本号: cat /proc/version

```
ubuntu@VM-4-5-ubuntu:~$ cat /proc/version  
Linux version 5.4.0-77-generic (buildd@lgw01-amd64-028) (gcc version 9.3.0 (Ubuntu 9.3.0-17ubuntu1~20.04))  
#86-Ubuntu SMP Thu Jun 17 02:35:03 UTC 2021  
ubuntu@VM-4-5-ubuntu:~$
```

2.3 在部署 kilolink 过程中，安装 docker 长期未响应。

解决方法：

docker 安装过程较为缓慢，需要耐心等待。可以使用命令 “docker version” 查看以及确认是否安装成功 docker。

2.4 服务器按照 kilolink 手册部署完成后，使用 IP 加端口无法登录服务器都的 web

解决方法：

检查部署 PLATFORMIP=103.134.224.34 时，配置的 IP 地址的是否正确

步骤 a 先进入到 docker: docker exec -it klnkserver bash

步骤 b 打开配置文件: vi /data/configs/platform.conf

步骤 C 修改 server IP 为正确的服务器 IP 地址

步骤 d 重启 Nginx: /usr/local/openresty/nginx/sbin/nginx -s reload

```
{
  serverip = "192.168.22.230",
  klnkport = 60000,
  klnkfwport = 60001,
  fwd_minport = 30000,
  fwd_maxport = 30050,
  tunip= "192.168.22.230",
  tunport = 3478,
  tunuser = "amin",
  tunpass = "kiloview123456",
  tunrealm = "demo",
}
~
~
~
```

2.5 客户之前使用旧版本 kilolink，然后升级到新版本的 kilolink 后，P1/P2 升级至最新固件 (4.7.2519) 后依旧无法正常连接 kilolink

解决方法：

建议删除容器和镜像后重新部署。(旧版 kilolink 的配置文件默认存在/data 下，因此建议直接删除 /data)

2.6 安装完成后，登录服务器报以下错误：



解决办法：

(1) 重新载入 nginx 的 sbin 目录：

```
docker exec -it klnkserver /usr/local/openresty/nginx/sbin/nginx -s reload
```

(2) 退出容器：exit

(3) 重启容器：docker restart klnkserver (通过 docker ps -a 查看容器名称)

2.7 使用新版 kiloview 部署成功后，且可以通过 IP+端口登录，但是 P1/P2 无法正确连接到 kilolink 服务器。

解决方法：

(1) 首先检查设备固件版本是否更新到最新版本

(下载地址地址：<https://www.kiloview.com/cn/support/download/>)

(2) 设备端信息是否填写正确，P1/P2 连接 kilolink 端口为 60000。检查服务器信息以及注册码是否正确

(3) 尝试在 kilolink 上删除注册信息，重新进行注册测试。

(4) 查看服务器相应端口是否打开；

服务器需使用以下端口，如果服务器的网络环境中存在防火墙，需开放相应端口。

端口	协议
82	TCP
3478	TCP+UDP
60000	UDP
60001	UDP
5000-5100	TCP+UDP
5000-5100	TCP+UDP
30000-30050	TCP+UDP

2.8 Kilolink 如何修改默认登陆 web 端口

解决方法：修改 kilolink 登录端口

步骤 a 进入容器: `docker exec -it klnkserver bash`

步骤 b 打开文件: `vi /usr/local/openresty/nginx/conf/nginx.conf`

将文件中的 `server -listen` 修改为 `8081`,然后保存退出。

步骤 c 重启 Nginx: `/usr/local/openresty/nginx/sbin/nginx -s reload`

```
keepalive_timeout 65;
charset utf-8,gbk;
#gzip on;

lua_package_path "/usr/local/openresty/lualib/?.lua;/usr/local/openresty/nginx/lua/?.lua;/data/web/api/?.lua;/usr/share/lua/5
a;";

server {
    listen 8081;
    server_name localhost;
    client_max_body_size 50m;

    location ~* /QuickBoard/([0-9\\.]*)(.+) {
        proxy_set_header Host $http_host;
        proxy_set_header X-Forward-For $remote_addr;
        proxy_set_header platform this-is-made-by-kiloview-for-platform-login;
        proxy_pass http://$1/$2?$args;
        add_header Access-Control-Allow-Origin *;
    }
    location ~* /SettingPage/([0-9\\.]*)(.+) {
        proxy_set_header Host $http_host;
        proxy_set_header X-Forward-For $remote_addr;
        proxy_set_header platform this-is-made-by-kiloview-for-platform-login;
        proxy_pass http://$1/$2?$args;
        add_header Access-Control-Allow-Origin *;
    }

    location / {
        root /data/web/html;
        try_files $uri $uri/ @router;
        index index.html index.htm;
        add_header Access-Control-Allow-Origin *;
    }

    location @router {
        rewrite ^.*$ /index.html last;
    }

    location ^~ /firmwares/ {
        root /data;
    }
}

:wq
```

```
root@kiloview-virtual-machine:/# docker exec -it klnkserver bash
root@kiloview-virtual-machine:/# cd /usr/local/openresty/nginx/conf
root@kiloview-virtual-machine:/usr/local/openresty/nginx/conf# vi nginx.conf

[1]+  Stopped                  vi nginx.conf
root@kiloview-virtual-machine:/usr/local/openresty/nginx/conf# vi nginx.conf
root@kiloview-virtual-machine:/usr/local/openresty/nginx/conf# /usr/local/openresty/nginx/sbin/nginx -s reload
root@kiloview-virtual-machine:/usr/local/openresty/nginx/conf#
```



注意

修改完成后，kilolink 的登录方式为:IP:8081 端口

3 kilolink 聚合功能操作指导

3.1 聚合平台登录

在浏览器中打开 `http://服务器 IP:82/` ,登录到聚合平台。缺省登录设备的用户名是：**admin**，密码是：**admin**。



注意

- 为保证信息安全，建议您首次登录后、立即修改密码！
- 聚合服务器登录端口默认为 82。

通过平台左下角的“系统状态”检查“版本信息”，确保聚合服务平台更新到最新的版本。



3.2 设备连接聚合平台

第一步：平台添加设备，并生成授权码

进入“设备管理”页面，点击“添加设备”。在弹出的对话框填写对应参数和生成授权码。



说明

- **序列号**：登录设备 WEB 页面，序列号在左下角系统状态栏，序列号由一串 9 位数字组成。
- **名称**：可配置任意中英文、数字和符合组合。
- **授权码**：点击“生成授权码”按钮，系统自动生成一串字母和数字组合的授权码，授权码用于设备注册到平台时填写。
- **所属用户**：将添加的设备分配给某用户可见，管理账号下展示所有的设备。
- **私密**：选择私密后，添加的设备将仅在所属用户和管理账号（admin）可见。

第二步：设备注册到平台

登录设备的 WEB 管理页面，选择“网络和服务器设置” - “连接到聚合服务器”，填入对应的参数信息，开启服务并保存配置。



说明

为保证您的使用效果，请确保您的设备固件和对应连接的 KiloLink Server 已更新至最新版本。
获取最新固件下载地址，请访问网址 <https://www.kiloview.com/cn/support/download/>，在筛选列表中，选择“视

频编码器” > “P1/P2”，并在“程序”标签中找到最新的固件下载。

获取最新 KiloLink Server 版本，请参考：https://www.kiloview.com/cn/support/docs/p2/4g_aggregation/

- **服务器地址**：聚合服务器的 IP 地址，可支持域名。
- **服务端口**：聚合服务器登录管理页面的端口，该端口默认为 60000。
- **授权码**：聚合服务器添加设备时生成的授权码。

网络和服务设置

修改主机名称

网络接口管理

便捷化管理功能

Web服务设置

Onvif服务设置

Telnet服务设置

ARP解析设置

静态路由设置

连接到聚合服务器

串口和PTZ

语音对讲

快速复位

连接状态：正在连接

聚合配置

开启聚合服务：是

服务器地址：43.128.30.176

服务端口：60000

授权码：2ZL98LJPM5

连接超时：15 秒

聚合接口： 默认以太网口
 3G/4G设备 1
 3G/4G设备 2
 默认WIFI

保存参数



注意

聚合端口只有四个选项：默认 WIFI、3G/4G 设备 1、3G/4G 设备 2 和默认以太网口。如插入 4G USB 上网卡，上网卡存在两种上网模式，一种是 modem 模式，一种是以太网卡模式。Modem 模式下，网卡设备会被识别为 3G/4G 设备 3 或者 3G/4G 设备 4；以太网卡模式下网卡设备会被识别为 USB 网络连接 1 或者 USB 网络连接 2，聚合端口里的选项也会相应增加。

当聚合状态下显示“已连接”，此时设备成功连接到聚合服务器，设备通过选择的聚合网络接口传输。

手机通过左上角的二维码可以登录到聚合状态和配置页面，可关闭和配置聚合参数。

连接到聚合服务器



刷新

聚合状态

- 连接状态：已连接
- 聚合链路：eth0 | wlan0
- 发送统计：1.04M | - Byte
- 接收统计：350K | - Byte
- 往返时间：26.0 | - ms
- 丢包率：0.0 | - %
- 发送速率：15K | - bps
- 接收速率：9K | - bps



说明

- **聚合链路**: 显示选择的所有聚合网络链路。
- **发送统计**: 统计对应每个网络链路实际上传的数据。
- **往返时间**: 每个网络链路两端的往返时间, 时间越长可能造成聚合后的延迟越高。
- **丢包率**: 对应每个网络链路的丢包情况, 当丢包率过大可能造成视频传输不流畅等异常。
- **发送速率**: 对应每个网络链路的发送速率, 总速率相加等于实际配置的编码码率。

3.3 聚合平台管理

3.3.1 在线设备

该列表展示当前用户下所有成功连接聚合服务的设备, 统计聚合状态下接收和发送的数据总量。



说明

未在该列表的设备为离线设备, 表示设备未成功注册或者设备未开机注册到聚合平台, 所有已经添加的设备都可以在设备管理列表中展示。

3.3.2 设备管理

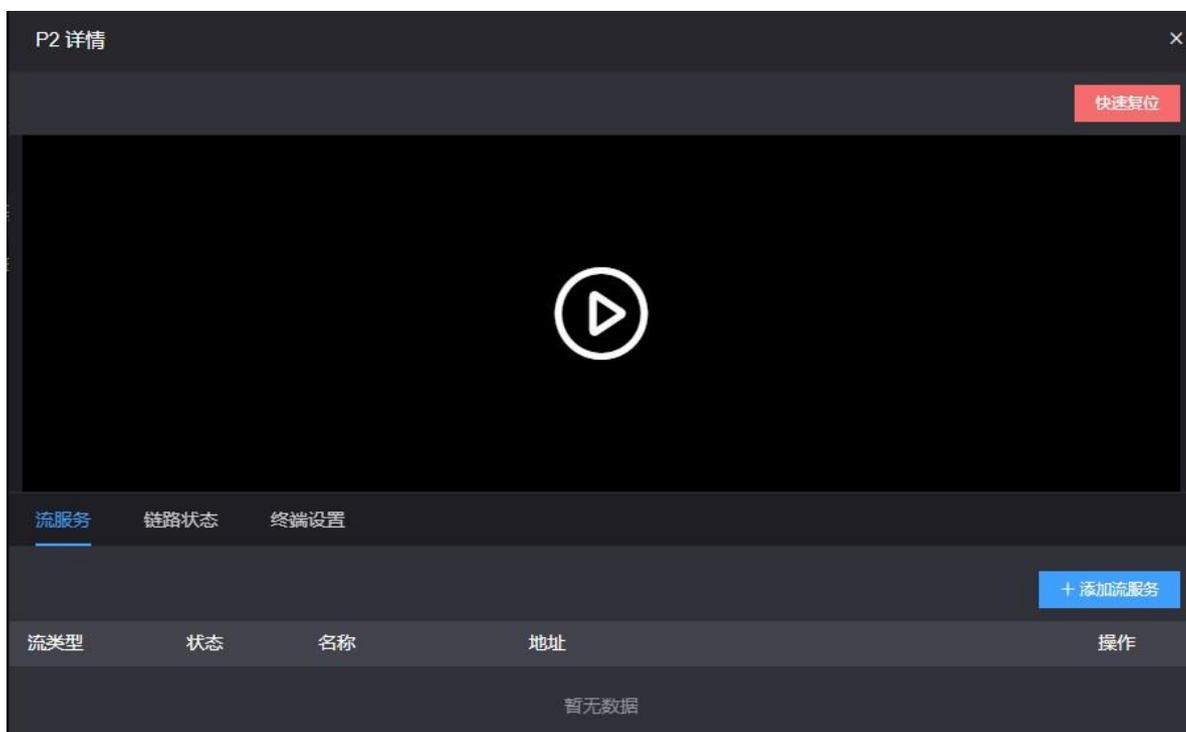
设备管理列表展示当前用户所属设备，包含在线和离线设备，对在线设备进行画面预览、端口映射、设备编码设置和流服务配置。



状态	名称	序列号	授权码	所属用户	NET IN	NET OUT	操作
在线	KV	181201032	HLY5WO8IGI	admin	2.19MB	387.00KB	[操作图标]
离线	Soheto	201230057	PYO6MIBUT6	Soheto	0Byte	0Byte	[操作图标]
离线	12	202005284	5Y5COTKODU	admin	0Byte	0Byte	[操作图标]

① 视频预览

点击在线设备所在行的任意区域，在页面的右侧弹出设备流服务和视频预览框，可预览设备实时的视频画面、流服务配置、链路状态和终端设置。



P2 详情

快速复位

流服务 链路状态 终端设置

+ 添加流服务

流类型	状态	名称	地址	操作
暂无数据				



注意

- 视频预览拉取视频是设备的子码流，必须保证在设备端开启了子码流，否则将无法收到设备的视频画面。
- 默认视频预览是停止不拉流状态，点击中间的播放键开启视频预览。

② 流服务



说明

- 通过该流服务进行推流，所有的流量将通过聚合链路进行传输。如果在设备端 WEB 管理页面进行流服务推送，则仅通过单一网络进行传输，且网络链路不可选。
- 推流码率最大不超过编码器端配置的编码码率，当聚合链路带宽不够时，流服务将自适应降低输出码率。

在预览页面，点击“添加流服务”，在弹出的流服务配置页面选择需要推送的“服务类型”。以 SRT 推送为例：

SRT 推送握手模式：支持 Listener 和 Caller 两种模式，聚合服务器部署在具有公网 IP 的云服务器端，握手模式可以直接选择 Listener，聚合服务器部署时端口默认预设为 30003-30050，例如选择端口 30004。

传输延迟根据当前发送和接收端的网络 RTT 延迟来配置，一般建议为 RTT 的 2.5-4 倍。

The image shows a configuration dialog box titled "添加流服务" (Add Stream Service). It contains the following fields and options:

* 名称	SRT
* 源选择	主码流
服务类型	SRT
开启推送服务	开启
握手模式	Listener
端口号	30004
传输延时(ms)	125 (0~5000)
显示高级设置	关闭

At the bottom of the dialog, there are two buttons: "取消" (Cancel) and "确定" (Confirm).

配置参数后点击“确定”保存，在流服务列表下可以看到添加的服务。接收端可以通过复制 SRT 完整地址拉流，或者通过 Caller 模式下，填写聚合服务器的 IP 和端口号（30004）进行拉流，其他参数发送和接收端保持一致。

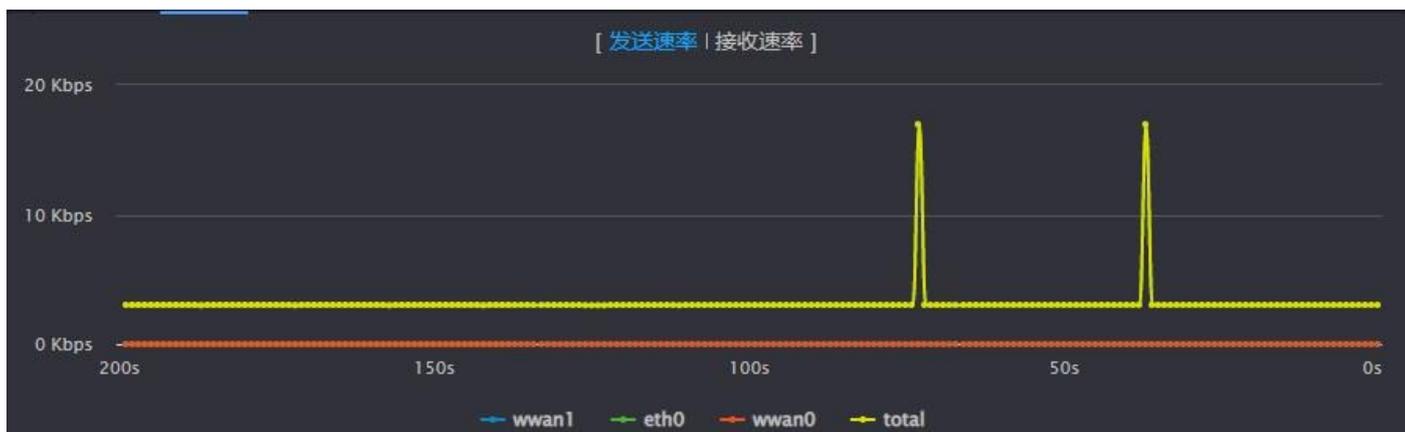


获取更多关于 P 系列编码器编码参数配置使用帮助，请访问 Kiloview 官方网站地址

<https://www.kiloview.com/cn/support/docs/p2/user/config/encode-and-stream/>

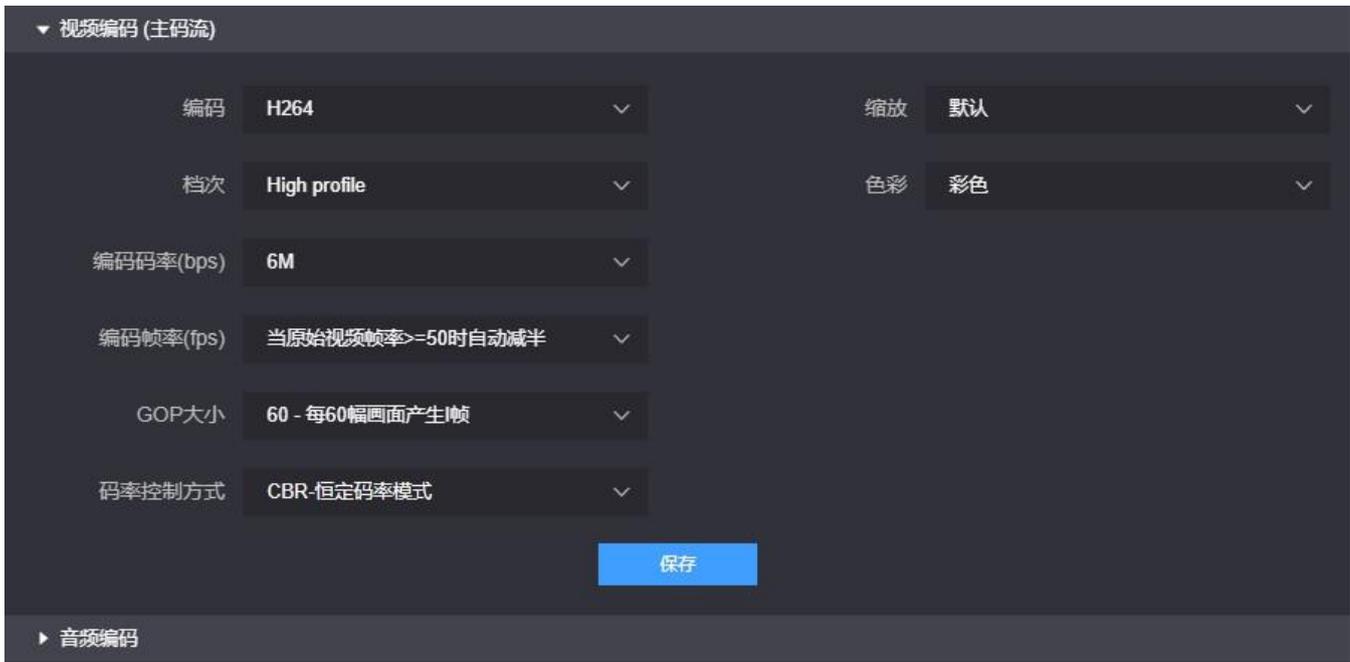
③ 链路状态

在链路状态中可实时查看每个网络通道的实时发送和接受的数据。



④ 终端配置

可对设备的主码流编码参数和音频参数进行配置，调整后的参数在设备的将实时更新为最新修改的参数信息，等同于在设备端进行编码参数的配置。



⑤ 端口转发

点击  图标进入端口映射页面，通过端口映射将设备的内网端口映射互联网，允许互联网其他用户来访问设备或者流服务。

点击“添加端口映射”，在弹出的对话框中进行参数配置。这里以设备的 RTSP 流为例进行配置，设备的 RTSP 流默认端口为 554，服务器端口 10240，选择端口的传输类型，确定保存后生效。

互联网其他用户可以通过 `rtsp://服务器 IP:10240/ch01`，拉取设备聚合后传输的 RTSP 流，例如：
`rtsp://43.128.30.176:10240/ch01`。



⑥ 聚合状态

点击  图标进入聚合状态页面，可以查看聚合实时状态和数据统计状态。

实时状态可以实时查看当前设备每个网络链路的实时发送/接收和丢包率等，统计状态统计当前设备的发送/接收和丢包重传数据的总量。

⑦ 设备页面管理

点击  图标，跳转到当前设备的 WEB 管理页面，可对设备参数进行配置。



获取更多关于 P 系列编码器参数配置的使用帮助，请访问 Kiloview 官方网站地址

<https://www.kiloview.com/cn/support/docs/p2/user/employ/>

3.3.3 用户列表

用户管理列表，可添加新的用户和对已有用户经常修改和删除，默认有一个 admin 的管理账号。

添加用户

* 用户名

* 昵称

* 新密码

* 确认密码

取消 确定



说明

- **管理账号：**管理账号为 admin，使用管理账号登录，设备列表展示所有添加的设备，并可以对列表中的设备进行编辑管理。
管理账号可添加新用户，并对所有用户管理和删除。对管理账号可以进行修改密码和昵称，不能被删除。
- **普通用户：**管理账号以外，所有添加账号为普通用户。使用普通用户登录，设备列表仅展示和管理分配给该用户的设备信息。
普通用户仅对当前的用户进行修改密码和昵称，没有删除权限。

3.4 其他

当不使用设备进行聚合链路传输时候，请及时关闭聚合服务，使设备处于离线状态，否则，聚合服务器将持续使用流量，造成流量的浪费。



P 系列设备的官方在线文档，请访问

<https://www.kiloview.com/cn/support/docs/p2>



Kiloview 官方技术支持网站，请访问

<https://www.kiloview.com/cn/support/>



长沙千视电子科技有限公司

<https://www.kiloview.com/>

湖南省长沙市雨花区汇金路 877 号嘉华智谷产业园（长沙屿）B4-106 栋/109 栋

邮箱：support@kiloview.com 联系电话：18573192787