# Kiloview Intercom Server Deployment (KIS)

# 1. Installation Environment Requirements

### 1.1. Hardware environment

Processor: Intel Core i3 CPU or higher

Hard disk: 32G Hard disk or higher

Memory: 2GB RAM or higher

### 1.2. Software environment

Ubuntu 18.04+ / Debian 9+

### 1.3. Network environment

1.3.1. IP and bandwidth requirements

IP address: if all the calling device is under local network, the voice intercom server doesn’ t need a public network IP, otherwise it needs a public network IP address.

Bandwidth: According to the numbers of simultaneous calling users, the bandwidth requirements of different user quantity as below:

| Numbers of simultaneous user | Bandwidth requirements（Mbps) |
|---|---|
| 10 | 7 |
| 20 | 30 |
| 30 | 70 |

Ways of calculation：64kbps×(The square of the user quantity – user quantity)/0.8

Note: Considering the hardware and maintenance costs of the server, as well as subsequent updating the version of cloud platform, it is recommended to lease the server from the cloud server provider, such as Alibaba Cloud, Tencent Cloud, Huawei Cloud, etc.

### 1.3.2. Port requirement

The voice intercom device need establish calling by voice intercom server port, so it is necessary to open all the corresponding ports.

Below port needs to be open:

TCP port：443 and 81

UDP port：50000-55000

# 2. Deployment process

## 2.1. Login to server

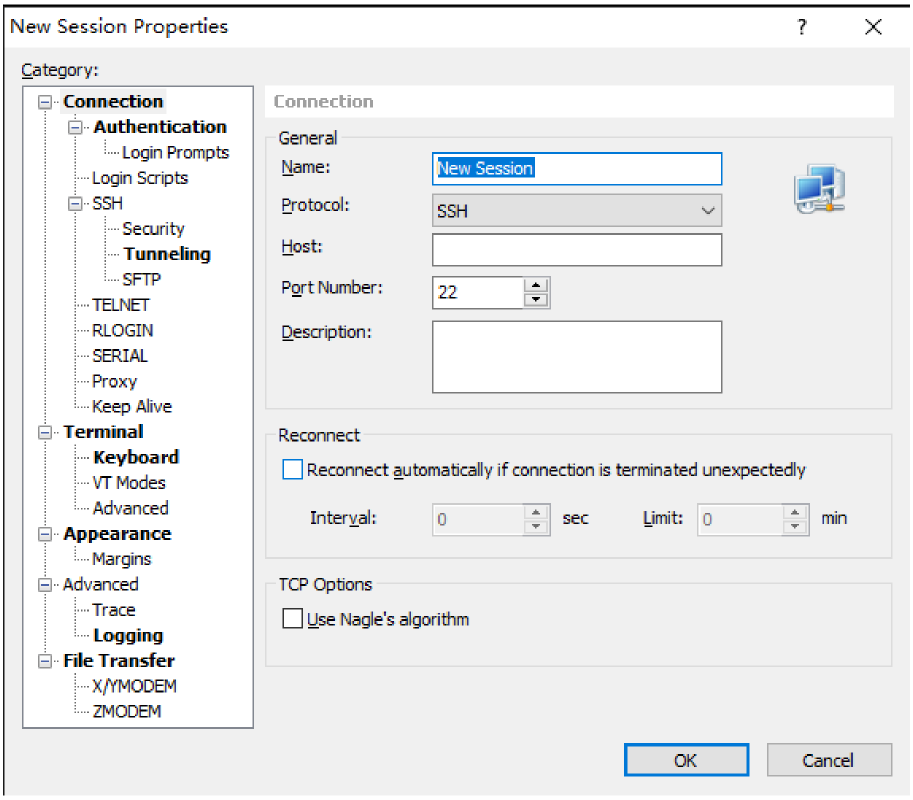
You can use remote terminal software to login to the server, Xshell tool or Putty is recommended.

Xshell download website: https://www.netsarang.com/zh/xshell-download/

Putty download website: http://putty.cs.utah.edu/download.html

### 2.1.1. Xshell tool

(1) After downloaded and installed Xshell, enter IP address of the server in the new session property and use SSH protocol to communicate. The default port is 22, click "Ok" after input completed.



New Session Properties

Connection

General

Name: New Session

Protocol: SSH

Host:

Port Number: 22

Description:

Reconnect

□ Reconnect automatically if connection is terminated unexpectedly

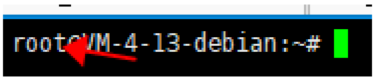
Interval: 0 sec Limit: 0 min

TCP Options

□ Use Nagle's algorithm

(2) Enter the user name and password in the pop up dialog box, if you are not a root user, it is recommended to switch to the root environment before operation.

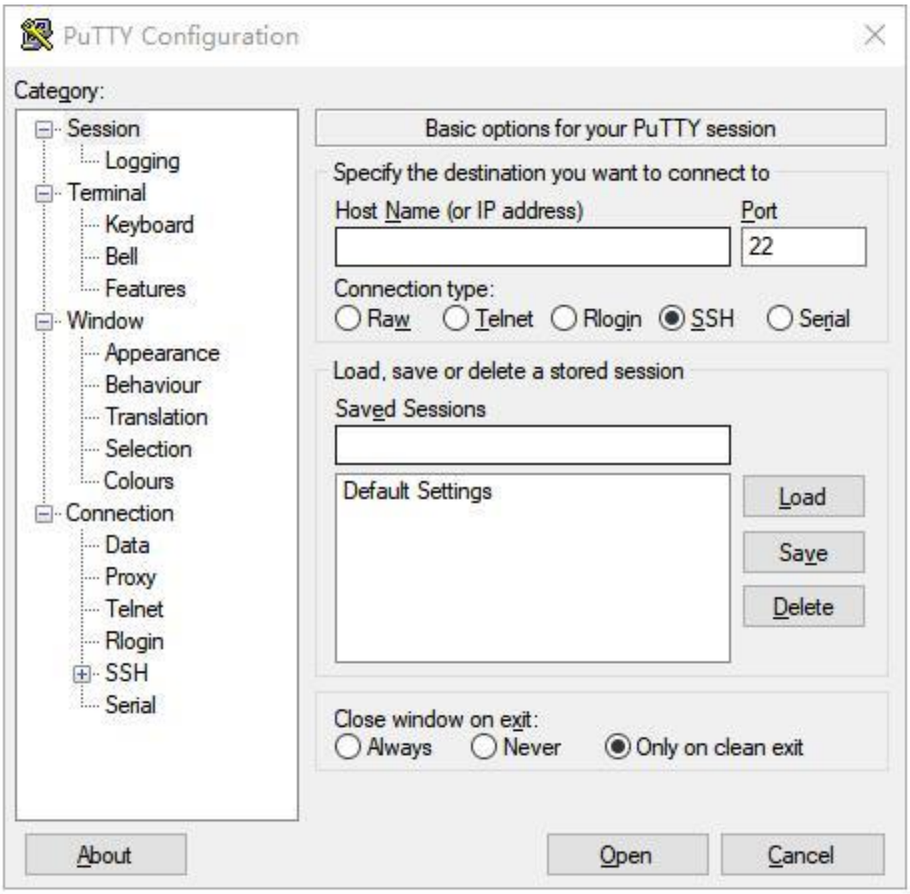
```
sudo su -
```



```
root@VM-4-13-debian:~#
```

2.1.2. Putty tool

(1) After downloaded and installed Putty, enter IP address of the server in the Putty configuration and select connection type as "SSH" . The default port is 22, click "open" after input completed



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address)

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

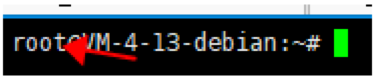
Saved Sessions

Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

(2) Enter the user name and password in the pop-up dialog box, if you are not a root user, it is recommended to switch to the root environment before operation.

```
sudo su -
```



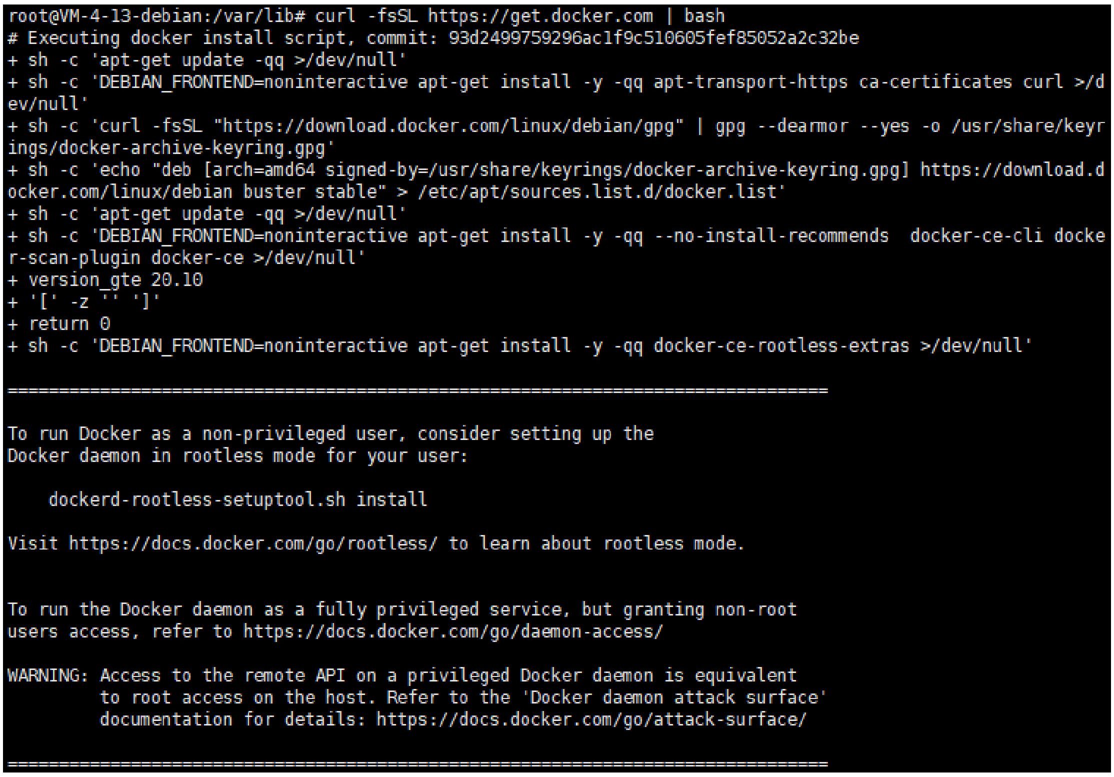
```
root@VM-4-13-debian:~#
```

## 2.2. The Container Installation Environment

### 2.2.1. Online installation

If your Docker version is 17.06 or above, you can skip this. You can check the current Docker version No. by inputting “docker version” in the terminal.

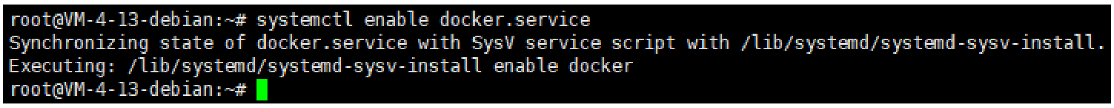
```
curl -fsSL https://get.docker.com | bash
```



```
root@VM-4-13-debian:/var/lib# curl -fsSL https://get.docker.com | bash
# 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 >/d
ev/null'
+ sh -c 'curl -fsSL "https://download.docker.com/linux/debian/gpg" | gpg --dearmor --yes -o /usr/share/keyr
ings/docker-archive-keyring.gpg'
+ sh -c 'echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.d
ocker.com/linux/debian buster 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 docke
r-scan-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'

================================================================================

To run Docker as a non-privileged user, consider setting up the
Docker daemon in rootless mode for your user:

    dockerd-rootless-setuptool.sh install

Visit https://docs.docker.com/go/rootless/ to learn about rootless mode.


To run the Docker daemon as a fully privileged service, but granting non-root
users access, refer to https://docs.docker.com/go/daemon-access/

WARNING: Access to the remote API on a privileged Docker daemon is equivalent
         to root access on the host. Refer to the 'Docker daemon attack surface'
         documentation for details: https://docs.docker.com/go/attack-surface/

================================================================================
```

### 2.2.2. Set Docker server to start automatically.

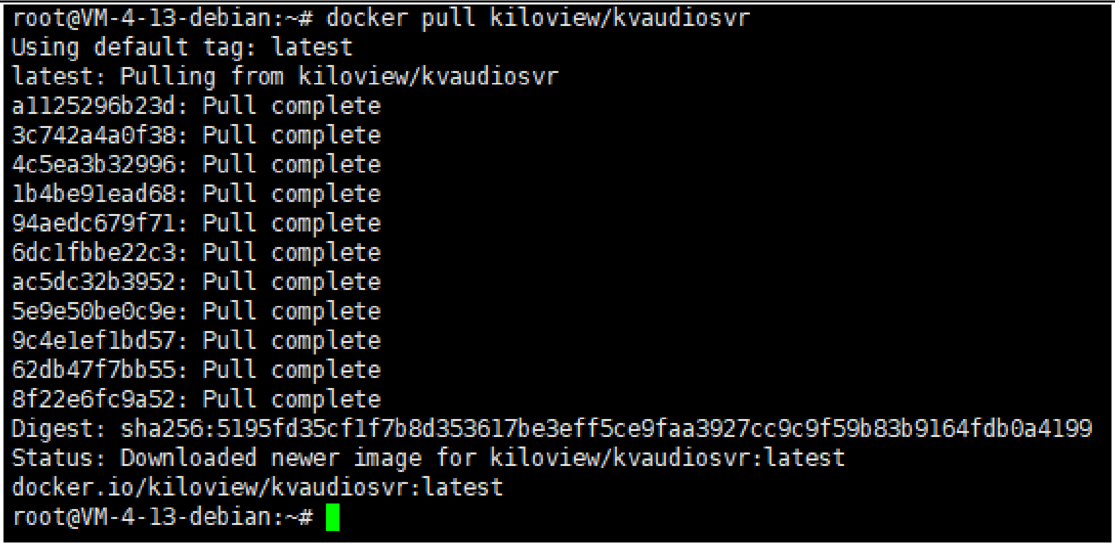
```
systemctl enable docker.service
```



```
root@VM-4-13-debian:~# systemctl enable docker.service
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
root@VM-4-13-debian:~# 
```

## 2.3. Pull the intercom server image file

```
docker pull kiloview/kvaudiosvr
```



```
root@VM-4-13-debian:~# docker pull kiloview/kvaudiosvr
Using default tag: latest
latest: Pulling from kiloview/kvaudiosvr
a1125296b23d: Pull complete
3c742a4a0f38: Pull complete
4c5ea3b32996: Pull complete
1b4be91ead68: Pull complete
94aedc679f71: Pull complete
6dc1fbbe22c3: Pull complete
ac5dc32b3952: Pull complete
5e9e50be0c9e: Pull complete
9c4e1ef1bd57: Pull complete
62db47f7bb55: Pull complete
8f22e6fc9a52: Pull complete
Digest: sha256:5195fd35cf1f7b8d353617be3eff5ce9faa3927cc9c9f59b83b9164fdb0a4199
Status: Downloaded newer image for kiloview/kvaudiosvr:latest
docker.io/kiloview/kvaudiosvr:latest
root@VM-4-13-debian:~#
```

If accessing to hub.docker.com is slower, you can follow below instead.

1. Download image file

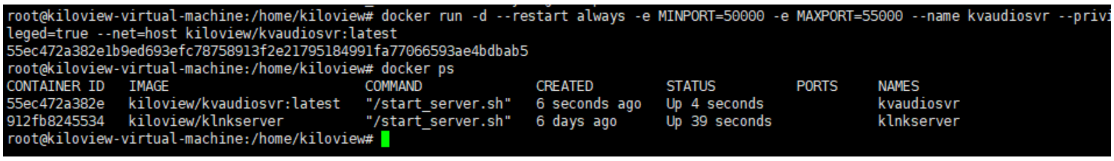
```
wget https://www.kiloview.com/downloads/Tools/.server/kvaudiosvr.tar
```

2. Decompress and load voice intercom image to Docker.

```
docker load < kvaudiosvr.tar
```

## 2.4 . Create and run the container

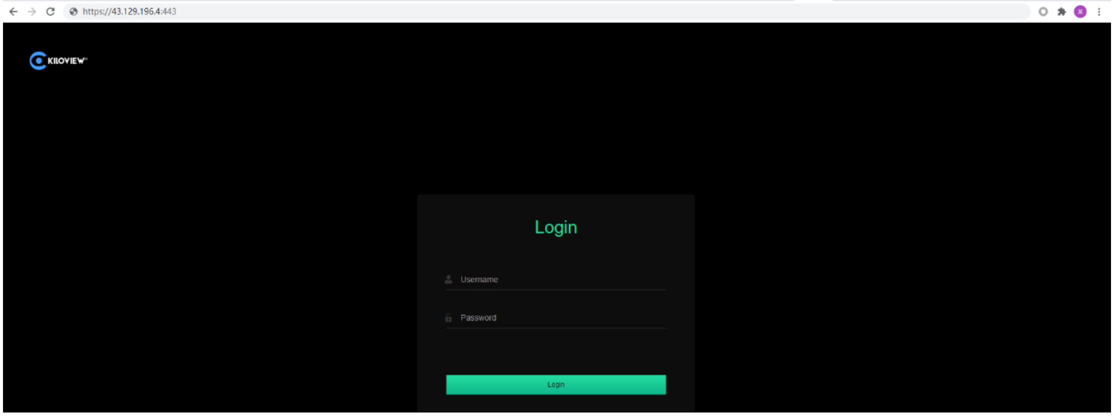
```
docker run -d --restart always -e MINPORT=50000 -e MAXPORT=55000 --name
kvaudiosvr --privileged=true --net=host kiloview/kvaudiosvr:latest
```



```
root@kiloview-virtual-machine:/home/kiloview# docker run -d --restart always -e MINPORT=50000 -e MAXPORT=55000 --name kvaudiosvr --privi
leged=true --net=host kiloview/kvaudiosvr:latest
55ec472a382e1b9ed693efc78758913f2e21795184991fa77066593ae4bdbab5
root@kiloview-virtual-machine:/home/kiloview# docker ps
CONTAINER ID   IMAGE                       COMMAND              CREATED          STATUS          PORTS     NAMES
55ec472a382e   kiloview/kvaudiosvr:latest  "/start_server.sh"   6 seconds ago    Up 4 seconds              kvaudiosvr
912fb8245534   kiloview/klnkserver         "/start_server.sh"   6 days ago       Up 39 seconds             klnkserver
root@kiloview-virtual-machine:/home/kiloview#
```

## 2.5. Login to the server

```
https://server IP:443
```



Initial user name and password are admin, please modify it in time.

# 3. Common questions about KIS deployment

### 3.1. After deployed KIS, login to the server by IP and port, after entering password, it prompts that unable to connect system service.

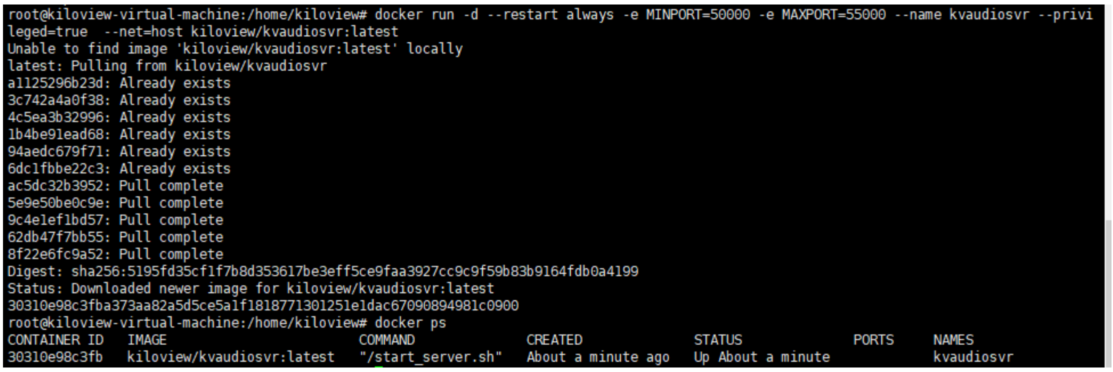
Solutions: “unable to connect system service” error. Firstly, you need delete previously KIS docker deployed, and use below command to re-image “docker pull kiloview/kvaudiosvr”, and then run the following command.

Step a. Stop KIS container: docker stop kvaudiosvr

Step b. Delete KIS container: docker rm kvaudiosvr

Step c. Delete KIS image: docker rmi kvaudiosvr

Step d. Recreate and run container: docker run -d --restart always -e MINPORT=50000 -e MAXPORT=55000 --name kvaudiosvr --privileged=true --net=host kiloview/kvaudiosvr:latest



```
root@kiloview-virtual-machine:/home/kiloview# docker run -d --restart always -e MINPORT=50000 -e MAXPORT=55000 --name kvaudiosvr --privi
leged=true  --net=host kiloview/kvaudiosvr:latest
Unable to find image 'kiloview/kvaudiosvr:latest' locally
latest: Pulling from kiloview/kvaudiosvr
a1125296b23d: Already exists
3c742a4a0f38: Already exists
4c5ea3b32996: Already exists
1b4be91ead68: Already exists
94aedc679f71: Already exists
6dc1fbbe22c3: Already exists
ac5dc32b3952: Pull complete
5e9e50be0c9e: Pull complete
9c4e1ef1bd57: Pull complete
62db47f7bb55: Pull complete
8f22e6fc9a52: Pull complete
Digest: sha256:5195fd35cf1f7b8d353617be3eff5ce9faa3927cc9c9f59b83b9164fdb0a4199
Status: Downloaded newer image for kiloview/kvaudiosvr:latest
30310e98c3fba373aa82a5d5ce5a1f1818771301251e1dac67090894981c0900
root@kiloview-virtual-machine:/home/kiloview# docker ps
CONTAINER ID   IMAGE                        COMMAND              CREATED              STATUS              PORTS      NAMES
30310e98c3fb   kiloview/kvaudiosvr:latest   "/start_server.sh"   About a minute ago   Up About a minute              kvaudiosvr
```

### 3.2. After deployed KIS, unable to make a normal voice call.

Solution: You need use below port for the server, if there is firewall in the network environment of the server, the corresponding ports must be open.

TCP port: 443 and 81

UDP port: 50000-55000

### 3.3. KIS and Kilolink are simultaneously deployed in the same server, and one of which cannot work normally.

As both KIS server and Kilolink server need work with 81 port, which will cause KIS or Kilolink to fail to login.

The default login method of Kilolink: IP + 81(port)

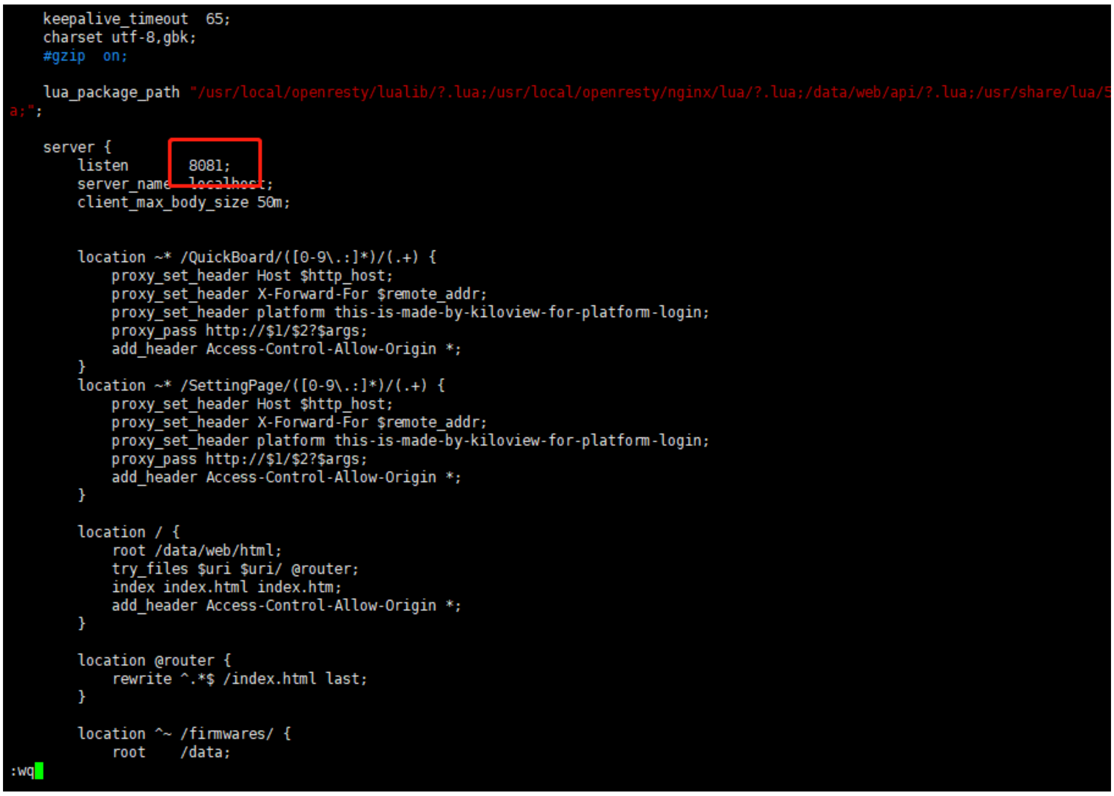
Solutions: Modify the default login port 81 of Kilolink to other port.

Step a: enter container: docker exec -it klnkserver bash

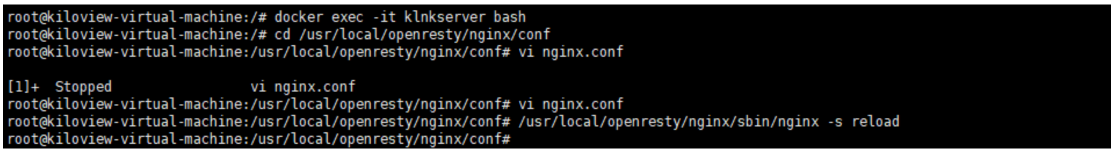
Step b: open the file: vi /usr/local/openresty/nginx/conf/nginx.conf

Modify “server-listen” in the file to 8081, save and exit.

Step c: restart 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#
```

Note: After modified, the method of KIS login is “https：IP+443 port”.

The method of Kilolink login is “IP+8081 port”.

For more questions, please contact us via：

https://www.kiloview.com/en/support

Please scan with browser.

**KILOVIEW Electronics CO., LTD.**

Tel：86-18573192787 Email：support@kiloview.com Web：www.kiloview.com/en

Address: B4-106/109, Jiahua Intelligence Valley Industrial Park, 877 Huijin Road, Yuhua District,
Changsha City, Hunan Province, China.