WCS on Google Cloud Platform

- Server deployment
  - Create and launch VM instance
  - Firewall rules setup
  - WCS installation and configuration
  - WCS starting and testing
- CDN deployment
  - CDN testing

Since build5.2.679,WCS can be deployed on Google Cloud Platform as standalone media server, CDN with low latency and CDN with load balancing between nodes.

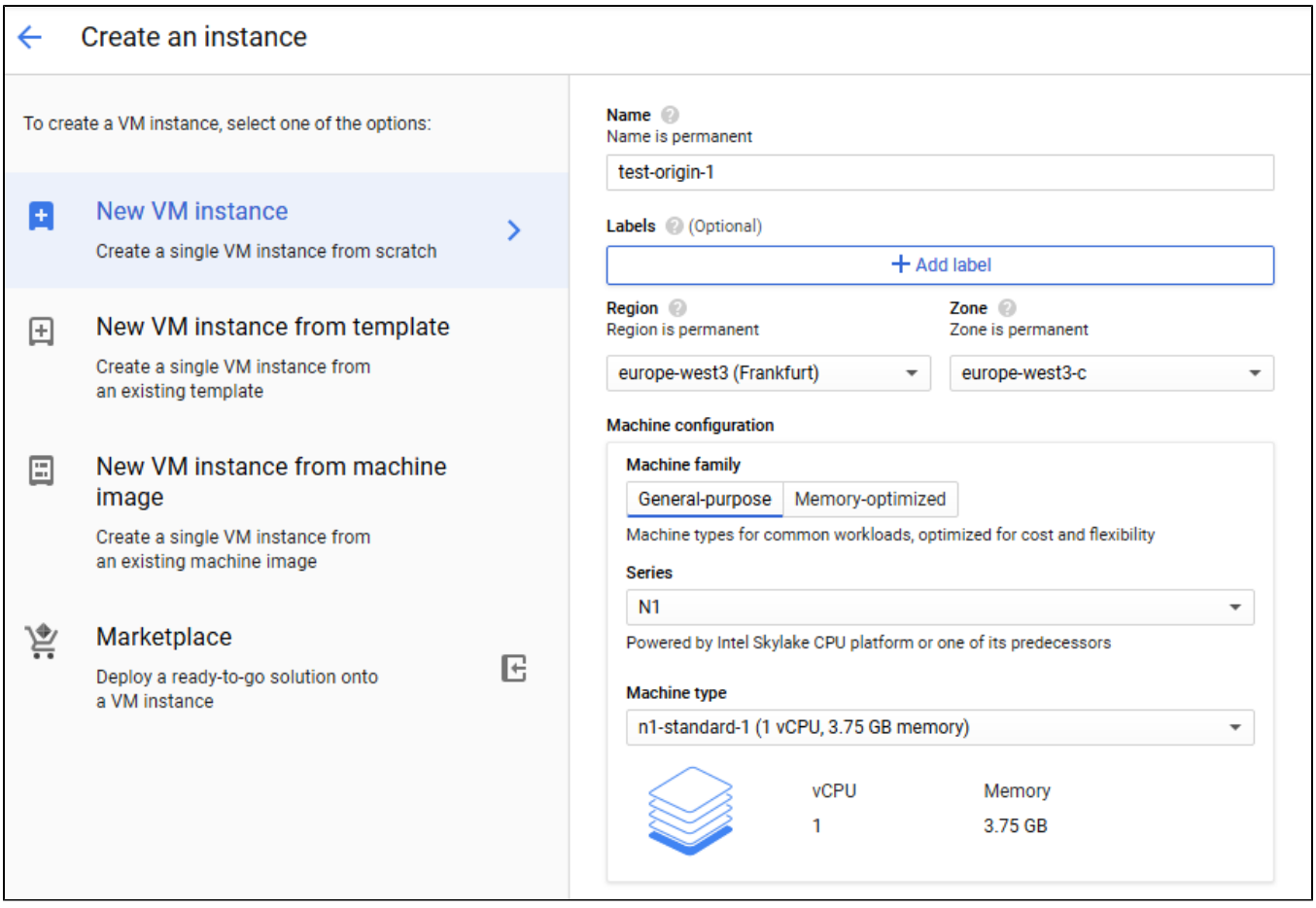
Anyway, before deploying, the following should be prepared:

- an active GCP account and a project in the account
- a WCSlicense to activate on server/servers
- optionally, domain names to bind to bind to servers static IPs and corresponding SSL certificates

Server deployment

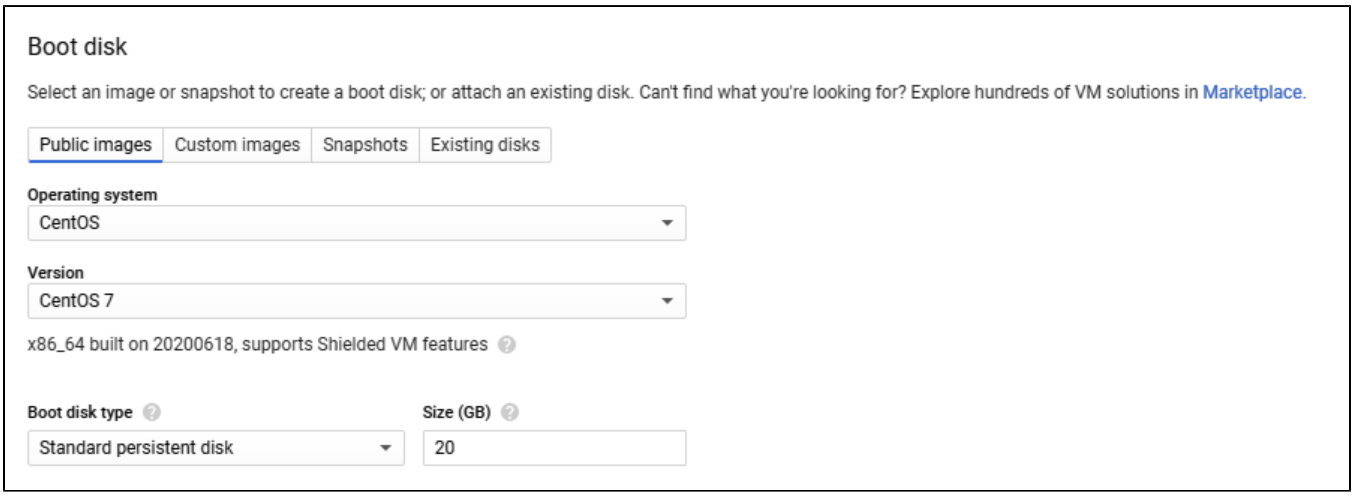
Create and launch VM instance

1. In Google Cloud console go to "Compute Engine - VM instances" section and click "Create VM instance" to start VM creation. Choose the server name, datacenter region and zone, VM configuration

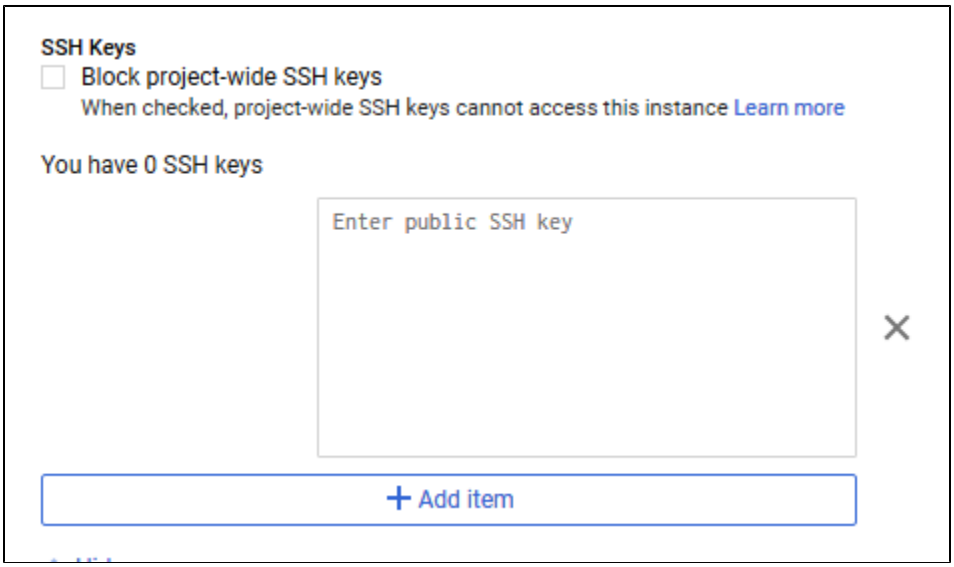


2. In "Boot disk" section click "Change". Choose CentOS 7.6 base image



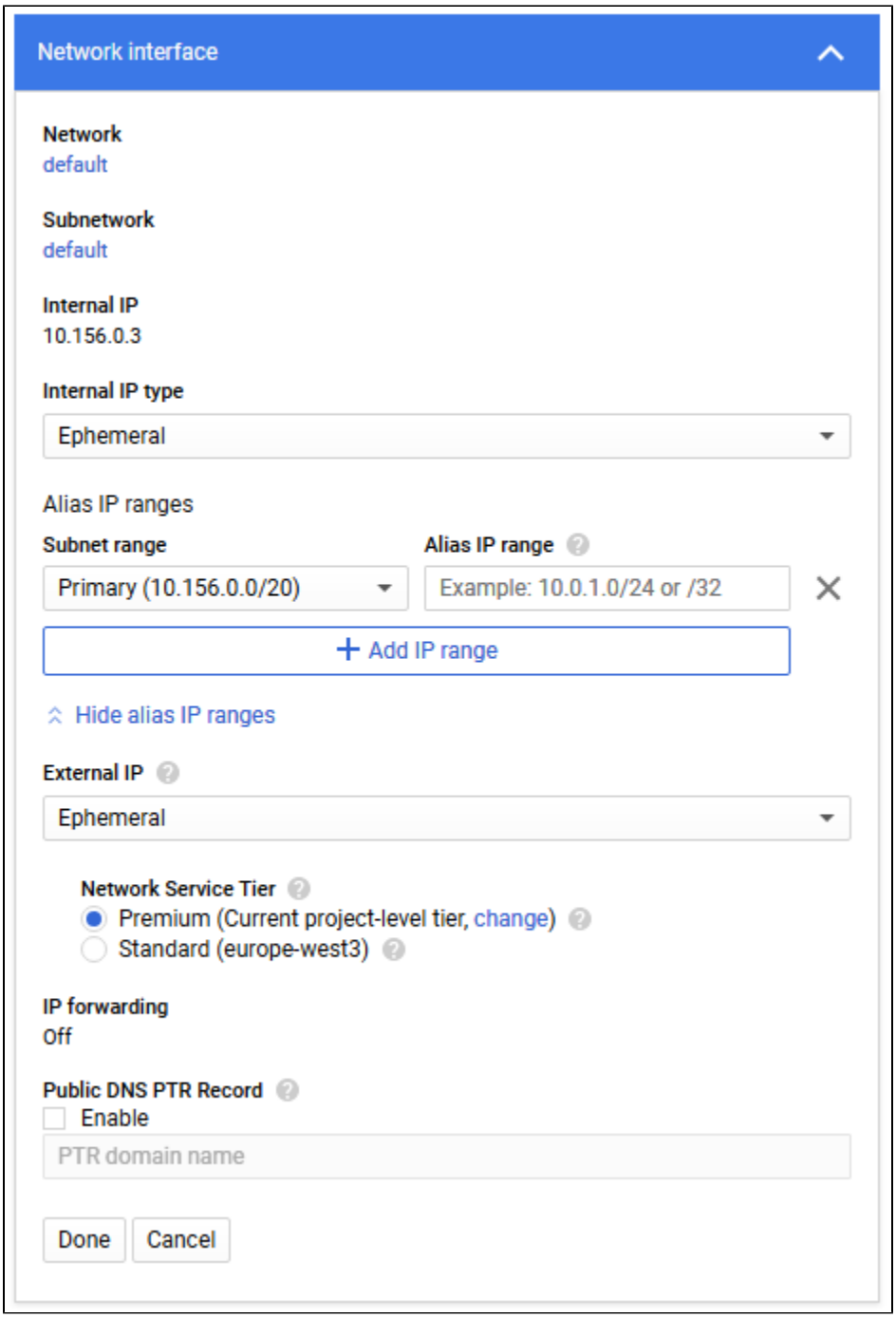


3. On "Security" tab add the public SSH key if you do not have project SSH keys

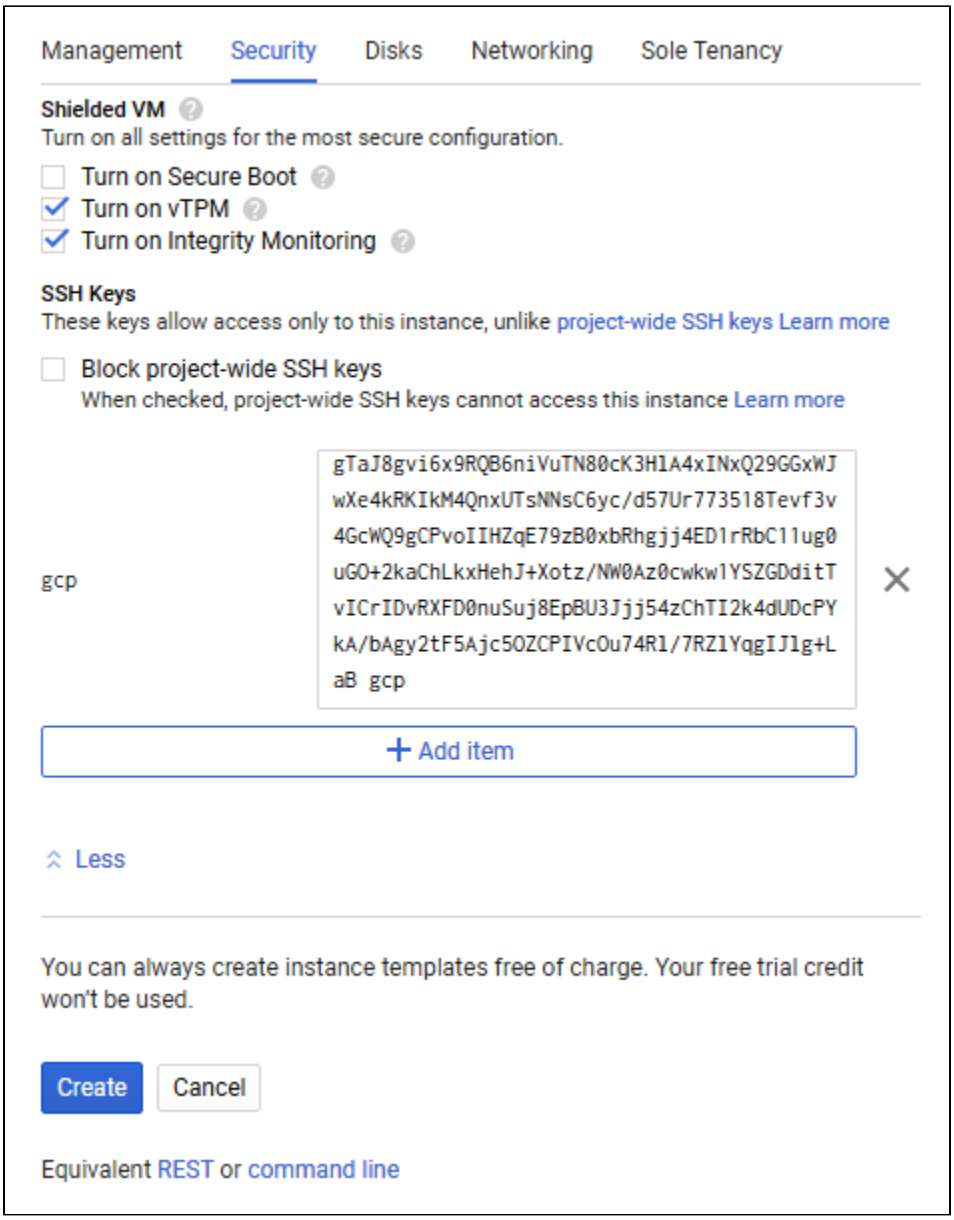


4. On "Network" tab choose external and internal IP addresses:

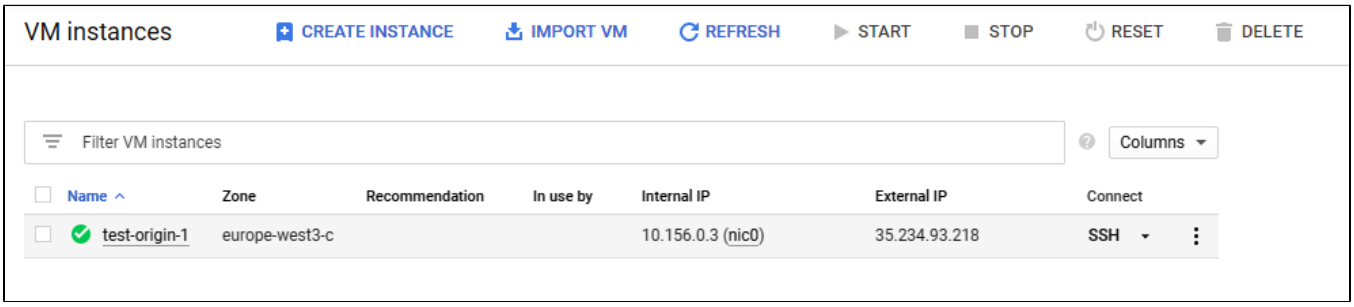
- if the server supposed to be Origin in CDN, it is recommended to reserve a static internal IP address;
- if there should be external entry points to the server (for example, to use for publishing/playing streams), it is recommended to reserve a static external IP address to bind domain name to



5. Click "Create"



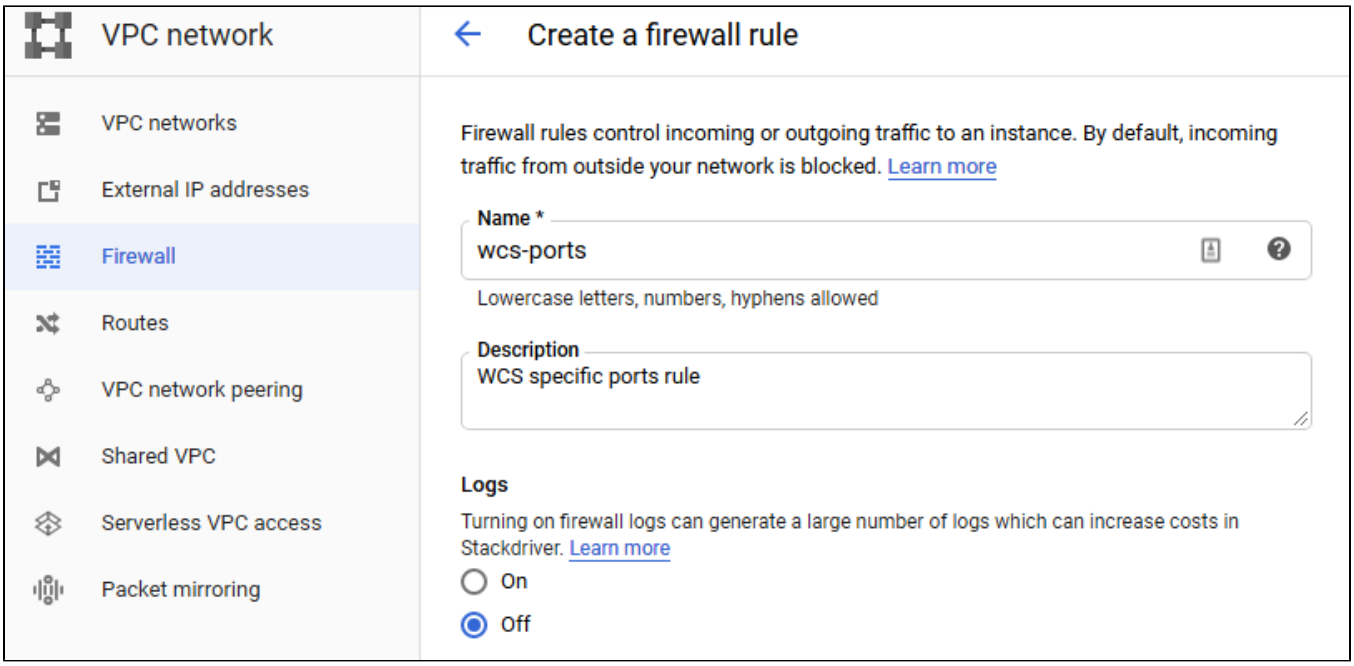
The server instab=nce wuill be created and launched



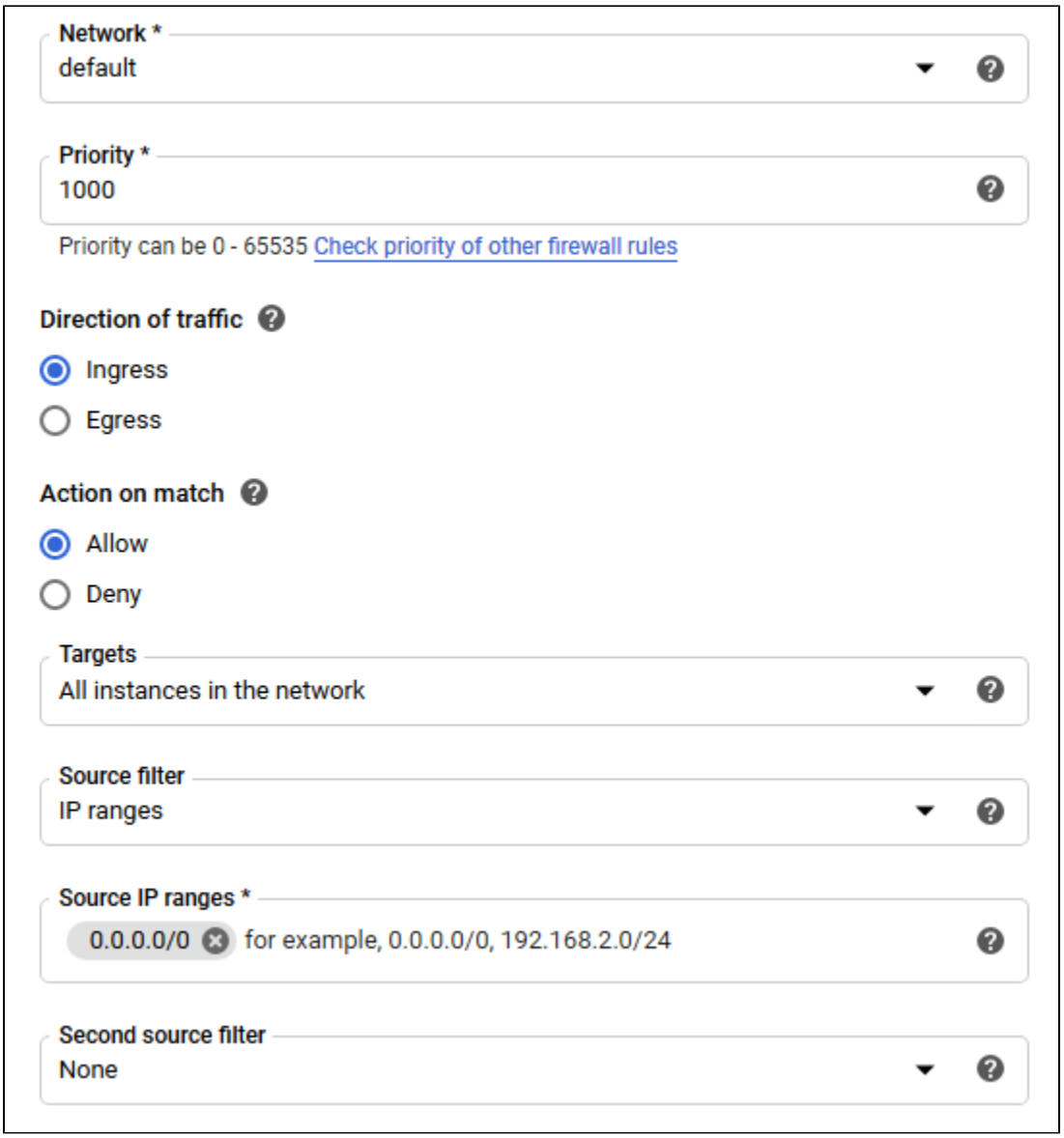
Firewall rules setup

Firewall rules affect all the instances in the project, so they should be set up once

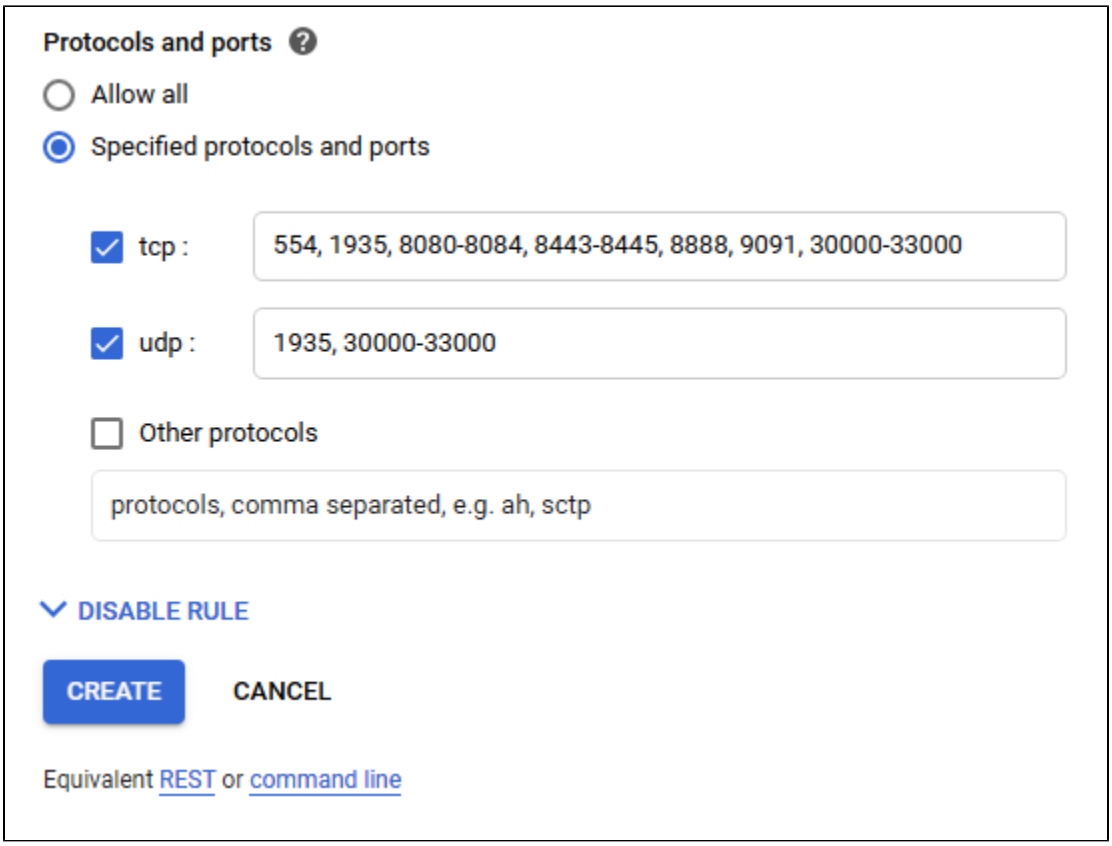
1. Go to "VPC network - Firewall"section and create "wcs-ports" rule



2. Allow incoming (ingress in terms of GCP) traffic from any node



3. Specify WCS ports and click "Create"



WCS installation and configuration

1. Install JDK. It is recommended to use JDK 12 or 14 if high load is planning

```
#!/bin/bash
sudo rm -rf jdk*
curl -s https://download.java.net/java/GA/jdk12.0.2/e482c34c86bd4bf8b56c0b35558996b9/10/GPL/openjdk-12.0.2
_linux-x64_bin.tar.gz | tar -zx
[ ! -d jdk-12.0.2/bin ] && exit 1
sudo mkdir -p /usr/java
[ -d /usr/java/jdk-12.0.2 ] && sudo rm -rf /usr/java/jdk-12.0.2
sudo mv -f jdk-12.0.2 /usr/java
[ ! -d /usr/java/jdk-12.0.2/bin ] && exit 1
sudo rm -f /usr/java/default
sudo ln -sf /usr/java/jdk-12.0.2 /usr/java/default
sudo update-alternatives --install "/usr/bin/java" "java" "/usr/java/jdk-12.0.2/bin/java" 1
sudo update-alternatives --install "/usr/bin/jstack" "jstack" "/usr/java/jdk-12.0.2/bin/jstack" 1
sudo update-alternatives --install "/usr/bin/jcmd" "jcmd" "/usr/java/jdk-12.0.2/bin/jcmd" 1
sudo update-alternatives --install "/usr/bin/jmap" "jmap" "/usr/java/jdk-12.0.2/bin/jmap" 1
sudo update-alternatives --set "java" "/usr/java/jdk-12.0.2/bin/java"
sudo update-alternatives --set "jstack" "/usr/java/jdk-12.0.2/bin/jstack"
sudo update-alternatives --set "jcmd" "/usr/java/jdk-12.0.2/bin/jcmd"
sudo update-alternatives --set "jmap" "/usr/java/jdk-12.0.2/bin/jmap"
```

2. Install accessory tools and libraries

```
sudo yum install -y tcpdump mc iperf3 fontconfig
```

3. Stop firewalld (it ti not necessary to block any ports on VM level because firewall rules were set up on project level)

```
sudo systemctl stop firewalld
sudo systemctl disable firewalld
```

4. Disable SELinux

```
sudo setenforce 0
```

5. Install WCS

```
curl -OL https://flashphoner.com/downloads/builds/WCS/5.2/FlashphonerWebCallServer-5.2.xxx.tar.gz
tar -xzf FlashphonerWebCallServer-5.2.xxx.tar.gz
cd FlashphonerWebCallServer-5.2.xxx
sudo ./install.sh
```

Where xxx is WCS actual build number

6. Activate your license

```
cd /usr/local/FlashphonerWebCallServer/bin
sudo ./activation.sh
```

7. Configure WCS (below the example of Origin server settings to publish WebRTC and RTMP streams)

```
flashphoner.properties   [----] 29 L:[  1+23  24/ 40] *(680 / 981b) 0010 0x00A
# Config flashphoner.properties
# To get more settings:
# ssh -p 2001 admin@localhost
# default password: admin
# show node-settings
# show node-settings | grep port

#server ip
ip                    =34.107.12.11
ip_local              =10.156.0.3

#webrtc ports range
media_port_from       =31001
media_port_to         =32000

#codecs
codecs                  =opus,alaw,ulaw,g729,speex16,g722,mpeg4-generic,telephone-event,h264,vp8,flv,mpv
codecs_exclude_sip      =mpeg4-generic,flv,mpv
codecs_exclude_streaming =flv,telephone-event
codecs_exclude_sip_rtmp  =opus,g729,g722,mpeg4-generic,vp8,mpv

#websocket ports
ws.port                =8080
wss.port               =8443

cdn_enabled=true
cdn_ip=10.156.0.3
cdn_role=origin
cdn_nodes_resolve_ip=false

# Request keyframes from WebRTC publishers every 5 seconds
periodic_fir_request=true

# Disable RTMP keepalives to publish from OBS
keep_alive.enabled=websocket,rtmfp

client_mode=false

rtc_ice_add_local_component=true
```

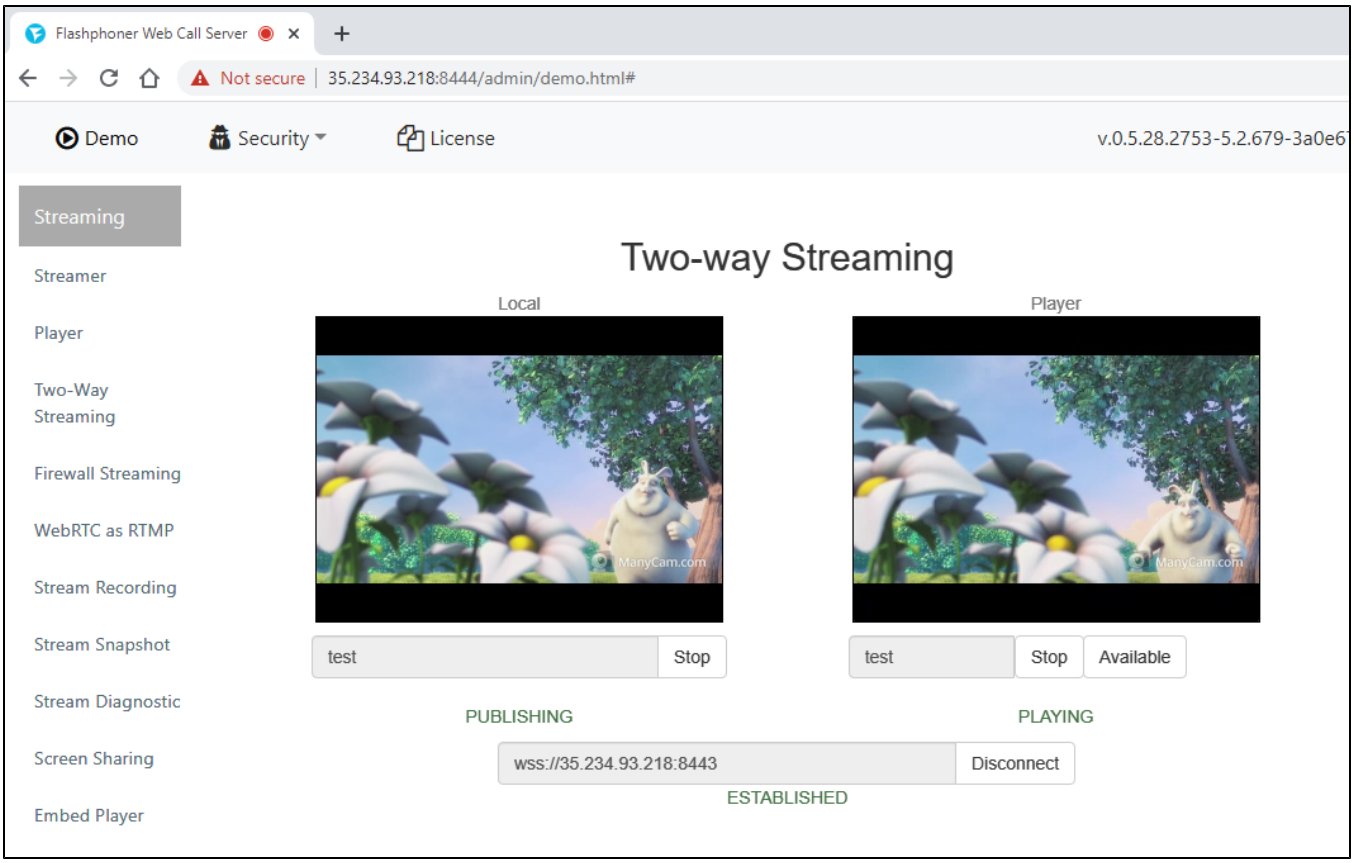
Do not change IP addresses in`ip`, `ip_local`и `cdn_ip`settings, they will be configured automatically on WCS startup.

WCS starting and testing

1. Start WCS

```
sudo systemctl start webcallserver
```

2. Enter to WCS web interface, open Two Way Streaming example, publish and play test stream



CDN deployment

1. Createand configureOrigin server as described above

2.Create and configure Edge server (below the example of Edge server settingt to play WebRTC streams)

```
flashphoner.properties   [-M--]  0 L:[  1+36  37/ 37] *(874 / 874b) <EOF>
# Config flashphoner.properties
# To get more settings:
# ssh -p 2001 admin@localhost
# default password: admin
# show node-settings
# show node-settings | grep port

#server ip
ip                    =34.107.12.11
ip_local              =10.156.0.5

#webrtc ports range
media_port_from       =31001
media_port_to         =32000

#codecs
codecs                  =opus,alaw,ulaw,g729,speex16,g722,mpeg4-generic,telephone-event,h264,vp8,flv,mpv
codecs_exclude_sip      =mpeg4-generic,flv,mpv
codecs_exclude_streaming =flv,telephone-event
codecs_exclude_sip_rtmp  =opus,g729,g722,mpeg4-generic,vp8,mpv

#websocket ports
ws.port               =8080
wss.port              =8443

cdn_enabled=true
cdn_ip=10.156.0.5
cdn_role=edge
cdn_point_of_entry=10.156.0.3
cdn_nodes_resolve_ip=false

client_mode=false

rtc_ice_add_local_component=true

http_enable_root_redirect=false
```

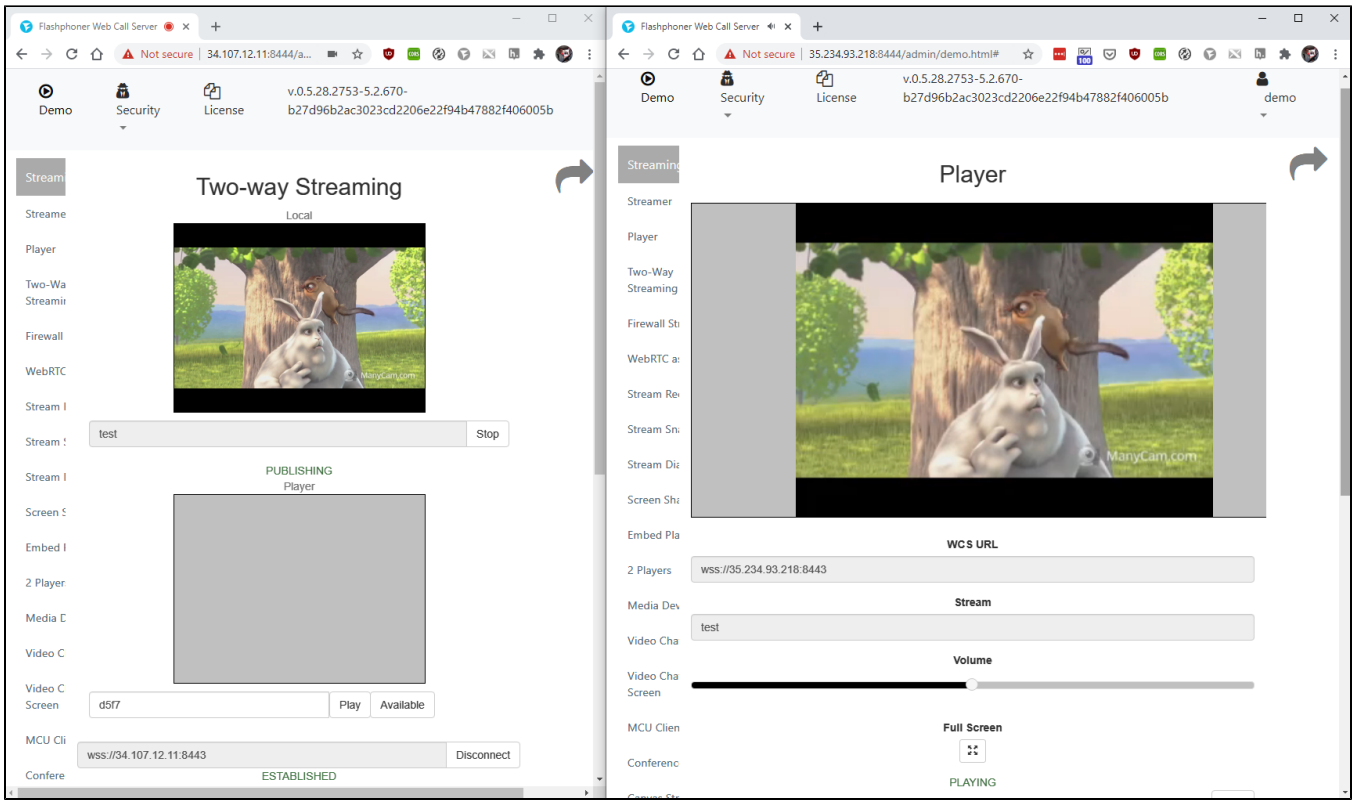
Do not change IP addresses in`ip`, `ip_local`и `cdn_ip`settings, they will be configured automatically on WCS startup.Set the`cdn_point_of_entry`para meter to Origin server static internal IP address

CDN testing

1. Start WCS on Origin and Edge VM instances.

2. Go to Origin web interface and publish test stream in Two Way Streaming example

3.Go to Edge web interface and play test stream in Player example


Flashphoner Web Call Server
Not secure
34.107.12.11:8444/a...
Demo
Security
License
v.0.5.28.2753-5.2.670-
b27d96b2ac3023cd2206e22f94b47882f406005b
Two-way Streaming
Local
test
Stop
PUBLISHING
Player
d5f7
Play
Available
wss://34.107.12.11:8443
Disconnect
ESTABLISHED
Flashphoner Web Call Server
Not secure
35.234.93.218:8444/admin/demo.html#
Demo
Security
License
v.0.5.28.2753-5.2.670-
b27d96b2ac3023cd2206e22f94b47882f406005b
demo
Player
WCS URL
wss://35.234.93.218:8443
Stream
test
Volume
Full Screen
PLAYING
ManyCam.com