

# WCS in Yandex.Cloud

## Server deployment from Yandex.Cloud Marketplace image

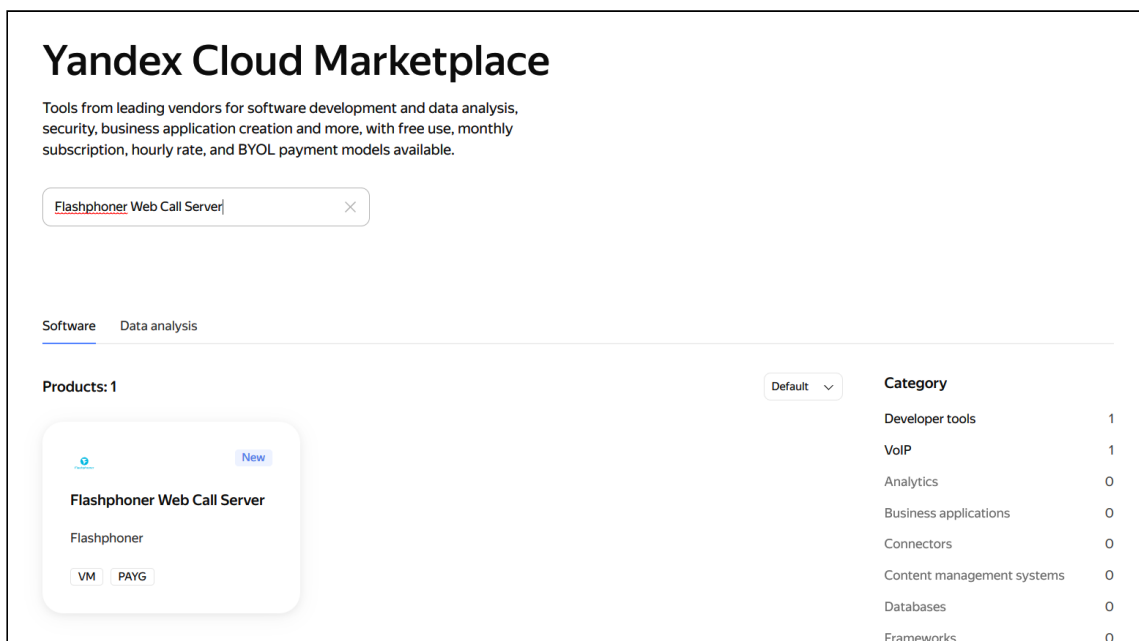
Flashphoner Web Call Server image with hourly billing is available in [Yandex.Cloud Marketplace](#). This way is preferable for short term instances (weeks and months).

The following is necessary to be prepared before deploy:

- active Yandex.Cloud account, a cloud and virtual private network in this account
- optionally, domain names to bind to servers static IPs and corresponding SSL certificates

## Create and launch Marketplace VM instance

1. Find the product Flashphoner Web Call Server in Yandex.Cloud Marketplace or open the page in [Yandex.Cloud Marketplace](#) directly



2. Click "Create VM" on the product page

## Flashphoner Web Call Server

Updated December 13, 2022

Web Call Server is a platform for real-time audio and video applications. It is designed primarily for developers who spin up streaming projects such as video chat, webinar, mass broadcasting, web calls, low-latency web and mobile apps.

The platform supports all popular today streaming video web-technologies such as WebRTC, Flash, RTMP, RTMFP, RTSP, HLS, MSE, SIP, and Websocket streaming, which allows delivering a video stream to a wide range of browsers and mobile devices.

Development tools and APIs:

- Web SDK
- iOS SDK
- Android SDK
- REST API

### Deployment instructions

1. Choose Flashphoner Web Call Server image from Cloud Marketplace when creating a virtual machine. A minimal VM configuration is 2 CPU, 2 Gb RAM, 100% CPU
2. Wait at least 30 seconds after the VM is created (all the first launch scripts should finish in this time)
3. Copy public VM IP address from Yandex Cloud console, open the page in a browser using this address `https://instance-public-ip:8444/admin/` and confirm the security exception when using a self-signed (embedded) SSL certificate

**from RUB 2,981 / per month**

The minimum VM cost with a basic configuration [?](#)

[Create VM](#)

[Calculate costs](#)

Billing type [?](#)

Hourly (Pay as you go)

Type

Virtual Machine

Category

VoIP Developer tools

Publisher

Flashphoner

### 3. Enter server name, description and choose datacenter region

## Create a virtual machine

### Basic parameters

Name [?](#)

Description [?](#)

Availability zone [?](#)

### Image/boot disk selection

Operating systems   Container Solution   Cloud Marketplace   Custom

### Recommended products

Flashphoner Web Call Server
i

PT Application Firewall 3.7.3
i

Hystax Acura Live Cloud Migration to Ya...
i

[Show more](#)

### 4. Choose storage type and size in "Disks" section

**Disks and file storages**

Disks 1 File storages

Disk name	Type	Size	Max. IOPS ?	Max. bandwidth ?
Flashphoner Web Call Server	Boot	HDD	300 / 300	30 / 30 MB/s
		20 GB		
		10 GB		
		8192 GB		

Add disk

5. Choose CPU type and count, adjust RAM size in "Computing resources" section. A minimal required parameters are set by default. Note that Guaranteed vCPU performance parameter must be 100%

**Computing resources**

Platform ? Intel Ice Lake

vCPU 2 2 96

Guaranteed vCPU performance ? 20% 50% 100%  
For any task, including high-load services.

RAM 2 GB 2 GB 32 GB

Additional  Preemptible ?

6. Choose available subnet, set manual IP addresses if necessary in "Network settings" section

### Network settings

Subnet ?  ▼

Public IP  Auto  List  No address

Advanced  DDoS protection ?

Internal IPv4 address  Auto  Manual

DNS settings for internal addresses ▼

7. Set user name and public SSH access key in "Access section", then click "Create VM"

### Access

Service account ?  ▼ or

Login\* ?

SSH key\* ?

Advanced  Grant access to serial console ?

8. Wait for VM changes its state to "Running" (page refresh may be required)

Virtual machines

Filter by name  All statuses ▼ All availability zones ▼

Name	Status	OS	Platform	vCPU	vCPU performance	RAM	Preemptible	Disk size	Availability zone	Internal IPv4	Public IPv4	Created on	ID	
test-wcs	Running		Intel Ice Lake	2	100%	2 GB	no	20 GB	ru-central1-a	10.128.0.32	158.160.42.178	13.12.2022, at 05:48	fhung12bdtzongrnuo6	...

Testing WCS instance

1. Wait at least for 30 seconds after VM changes its state to "Running" for all the first launch scripts to work. Then copy a public IP address

Virtual machines													
Filter by name	All statuses		All availability zones										
Name	Status	OS	Platform	vCPU	vCPU performance	RAM	Preemptible	Disk size	Availability zone	Internal IPv4	Public IPv4	Created on	ID
test-wcs	Running	Ubuntu	Intel Ice Lake	2	100%	2 GB	no	20 GB	ru-central1-a	10.128.0.32	158.160.42.178	13.12.2022, at 05:48	fhnung12bdt2ongrnuu6

2. In a new browser tab, open the URL `https://public-ip:8444/admin/`, where `public-ip` - public IP address copied above. Accept the security exception (WCS uses self signed SSL certificates by default)

**Your connection is not private**

Attackers might be trying to steal your information from **158.160.42.178** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR\_CERT\_AUTHORITY\_INVALID

To get Chrome's highest level of security, [turn on enhanced protection](#)

[Hide advanced](#) [Back to safety](#)

This server could not prove that it is **158.160.42.178**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

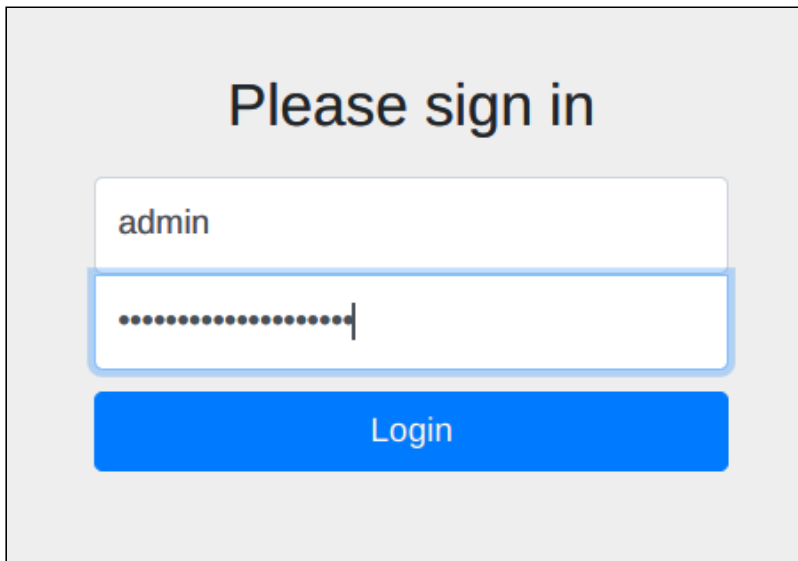
[Proceed to 158.160.42.178 \(unsafe\)](#)

WCS web interface login page will open.

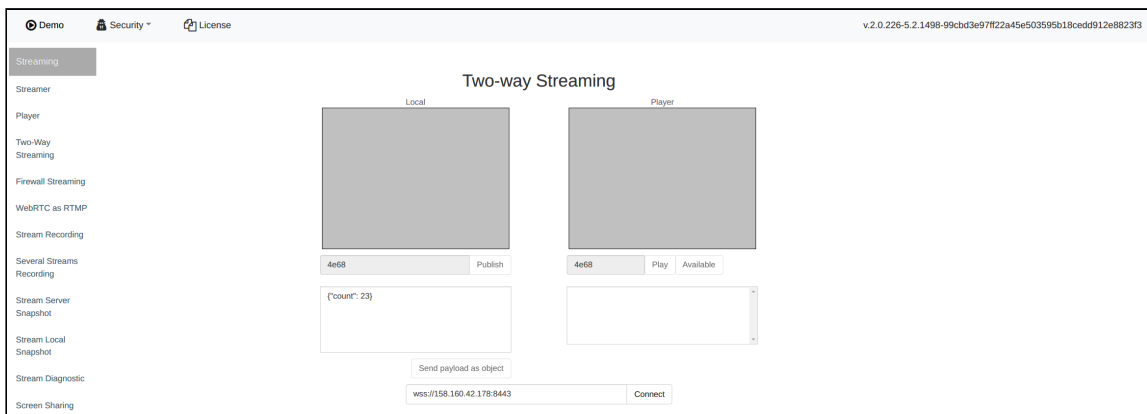
3. Copy ID from VM parameters

Virtual machines													
Filter by name	All statuses		All availability zones										
Name	Status	OS	Platform	vCPU	vCPU performance	RAM	Preemptible	Disk size	Availability zone	Internal IPv4	Public IPv4	Created on	ID
test-wcs	Running	Ubuntu	Intel Ice Lake	2	100%	2 GB	no	20 GB	ru-central1-a	10.128.0.32	158.160.42.178	13.12.2022, at 05:48	fhnung12bdt2ongrnuu6

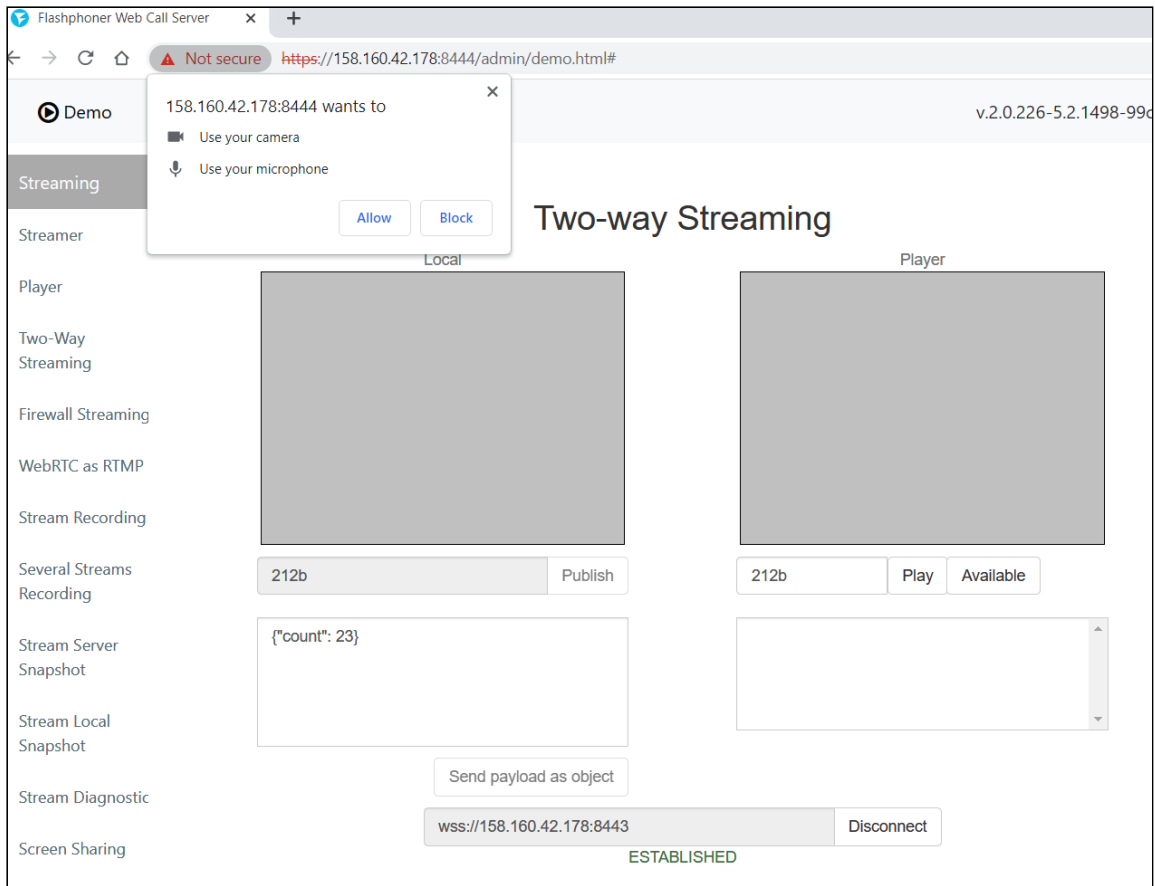
4. On WCS web interface login page enter **admin** user name, use ID copied above as password



5. Choose Two Way Streaming example




6. Click "Connect", then "Publish". Allow camera and microphone access



7. Click "Play" when **PUBLISHING** is displayed under the "Local" window

## Two-way Streaming

Local




212b Stop

PUBLISHING

{"count": 23}

Send payload as object

Player



212b Stop Available

PLAYING

wss://158.160.42.178:8443 Disconnect

ESTABLISHED

The browser sends media stream to the server and plays it from the server. The WCS instance is working.

## Server deployment from the scratch using one of the standard Linux images

Since build [5.2.759](#), WCS can be deployed in Yandex.Cloud as separate media server or low latency streaming CDN node.

The following is necessary to be prepared before deploy:

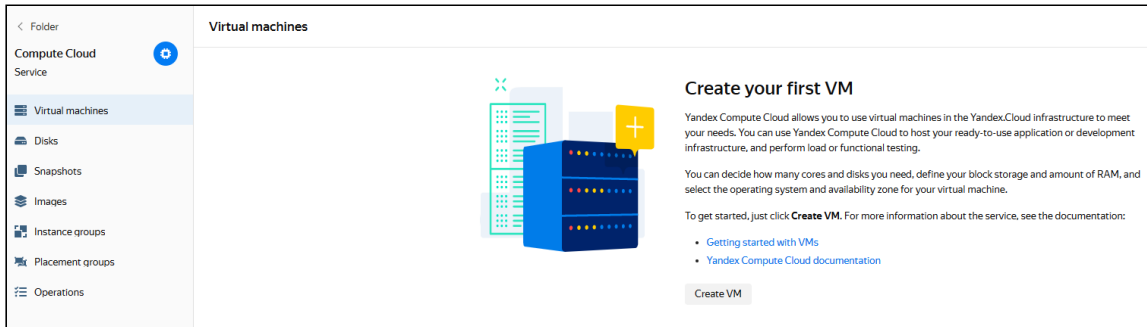
- active Yandex.Cloud account, a cloud and virtual private network in this account
- a WCS [license](#) to activate on server/servers
- optionally, domain names to bind to servers static IPs and corresponding SSL certificates

### Server deployment

#### Create and launch VM instance



1. In Yandex.Cloud console go to "Compute Cloud - Virtual machines" section and click "Create VM" to begin VM instance creation



2. Enter server name, description and choose datacenter region

### Create a virtual machine

**Basic parameters**

Name <sup>?</sup>

Description <sup>?</sup>

Availability zone <sup>?</sup>

3. In "Computing resources" section choose processor type and count, memory size. Set the parameter "Guaranteed vCPU performance" to "100%"

### Computing resources

Platform <sup>?</sup>

vCPU  2 80

Guaranteed vCPU performance <sup>?</sup>      
For any task, including high-load services.

RAM  2 GB 32 GB

Additional  Preemptible <sup>?</sup>

4. In "Image/boot disk selection" section choose Centos, version 7 (other operating systems listed here are allowed too)

Image/boot disk selection

Operating systems    Container Solution    Cloud Marketplace    Custom

Filter by operating system

Ubuntu	20.04	Windows Server	2019 Datacenter
Debian	10	<b>CentOS</b>	7
CoreOS	2303.4.0	openSUSE	42.3

Show all products

5. In "Disks" section choose disk type and size

Disks

Disk name	Type	Size	Max. IOPS	Max. bandwidth
CentOS 7 <b>Boot</b>	HDD	20 GB	—	—
		10 GB		4096 GB

Add disk

6. In "Network settings" section choose available subnet, set manual IP addresses if necessary

Network settings

Subnet ?    default / default-ru-central1-a

Public IP    Auto    List    No address

Advanced     DDoS protection ?

Internal address    Auto    Manual

7. In "Access" set user name and public SSH access key

### Access

Service account ? Create account

Login ?

SSH key ?

Advanced  Grant access to serial console ?

Create VM

then click "Create VM"

8. VM instance created will appear in VMs list

Virtual machines

Filter by name  All statuses ⌵ All availability zones ⌵ ⚙ Table settings 13/14

Name	Status	OS	Platform	vCPU	vCPU performance	RAM	Preemptible	Disk size	Availability zone	Internal IPv4	Public IPv4
test-wcs	Running		Intel Cascade Lake	2	100%	2 GB	no	20 GB	ru-central1-a	10.130.0.20	178.154.227.185

9. Click VM instance string, copy public IP address from "Network" section to access the server

### Network

Network interface ⋮

Private IPv4..... 10.130.0.20

Public IPv4..... 178.154.227.185

Subnet..... default-ru-central1-a

10. Connect to the instance by SSH

```

$ ssh -i /g/.ssh/id_rsa_yandex support@178.154.227.185
The authenticity of host '178.154.227.185 (178.154.227.185)' can't be established.
ECDSA key fingerprint is SHA256:69SQ1JWPNe3+F7fHhX1K70gmN/hIohHce9NNsrWbVA0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '178.154.227.185' (ECDSA) to the list of known hosts.
[support@test-wcs ~]$ uname
Linux
[support@test-wcs ~]$ uname -a
Linux test-wcs.ru-central1.internal 3.10.0-1127.el7.x86_64 #1 SMP Tue Mar 31 23:36:51 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
[support@test-wcs ~]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:             Little Endian
CPU(s):                 2
On-line CPU(s) list:   0,1
Thread(s) per core:    2
Core(s) per socket:    1
Socket(s):              1
NUMA node(s):          1
Vendor ID:              GenuineIntel
CPU family:             6
Model:                  85
Model name:             Intel Xeon Processor (Cascadelake)
Stepping:               6
CPU MHz:                2095.068
BogoMIPS:               4190.13
Hypervisor vendor:     KVM
Virtualization type:   full
L1d cache:              32K
L1i cache:              32K
L2 cache:               4096K
L3 cache:               16384K
NUMA node0 CPU(s):     0,1
Flags:                  fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht sys
x16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand hypervisor lahf_lm abm 3dnowprefetch in
dx smap clflushopt clwb avx512cd avx512bw avx512vl xsaveopt xsavec xgetbv1 arat pku ospke avx512_vnni md_clear spec_ctrl intel_
[support@test-wcs ~]$ free
              total        used         free      shared  buff/cache   available
Mem:           1881860         99072        1662108           556        120680        1647320
Swap:              0              0              0
[support@test-wcs ~]$ |

```

## Firewall rules setup

Yandex.Cloud does not support security groups now (the feature is in Preview state), therefore it is necessary to set up firewall on the instance itself:

 [iptables\\_setup.sh](#) 

## WCS installation and configuration

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

```

#!/bin/bash
sudo rm -rf jdk*
curl -s
https://download.java.net/java/GA/jdk12.0.2/e482c34c86bd4bf8b56c0b35558996b9/10/
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. Disable SELinux

```
sudo setenforce 0
```

## 4. 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

## 5. Activate your license

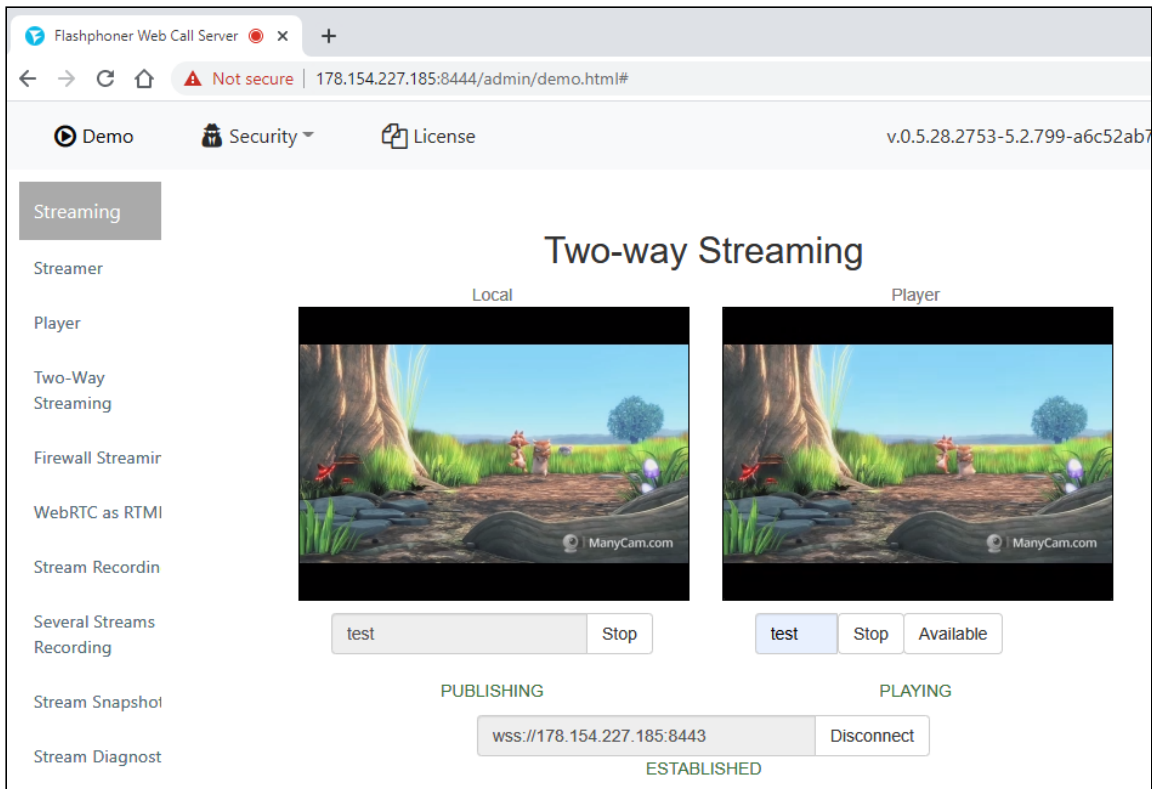
```
cd /usr/local/FlashphonerWebCallServer/bin
sudo ./activation.sh
```

## 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



## Default admin credentials

The running instance data can be received in Yandex.Cloud by two ways: using Google Cloud API endpoints or AWS EC2 API endpoints. Therefore, WCS detects cloud environment as Amazon-like since build [5.2.921](#).

In its turn, Amazon requires to use an unique admin password for every instance, and WCS sets admin password in Amazon-like cloud environment by unique instancelid available via API or in EC2 console.

Therefore, since build [5.2.921](#) WCS sets admin password to instancelid on first launch in Yandex.Cloud. However, this parameter may not be displayed in Yandex.Cloud console. To get instancelid, connect to the instance via SSH and use the following command

```
curl http://169.254.169.254/latest/meta-data/instance-id
```