

WCS in Yandex.Cloud

- [Server deployment from Yandex.Cloud Marketplace image](#)
 - [Create and launch VM instance](#)
 - [Testing WCS instance](#)
- [Server deployment from the scratch using one of the standard Linux images](#)
 - [Create and launch VM instance](#)
 - [Firewall rules setup](#)
 - [WCS installation and configuration](#)
 - [WCS starting and testing](#)
- [Default admin credentials](#)

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 VM instance

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

Yandex Cloud Marketplace

Tools from leading vendors for software development and data analysis, security, business application creation and more, with free use, monthly subscription, hourly rate, and BYOL payment models available.

Flashphoner Web Call Server

Software Data analysis

Products: 1

Flashphoner Web Call Server

Flashphoner

VM PAYG

New

Default

Category

Developer tools1

VoIP1

Analytics0

Business applications0

Connectors0

Content management systems0

Databases0

Frameworks0

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 [?](#)

test-wcs

Description [?](#)

Test WCS Marketplace image

Availability zone [?](#)

ru-central1-a

Image/boot disk selection

Operating systems

Container Solution

Cloud Marketplace

Custom

Recommended products



Flashphoner Web Call Server



PT Application Firewall 3.7.3



Hystax Acura Live Cloud Migration to Ya...



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	20 GB	300 / 300	30 / 30 MB/s
		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 ? default / default-ru-central1-a ▼

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 ? Not selected ▼ or Create new account

Login* ? fpsupport

SSH key* ? onOwsDCvYmVhNTZt4oOYU7t7OxRA5UjcWu
nPwzXFtikJDgIL0B25QuqObFP9NHc+ggJmNA
91dxUC9q1QY/GCJTIPdbYO9QOWGmrIXDDrZ
xroWDqxfXnMX5CEoVelsG556WUnYeXqqTG
G9mPnwohcsU41 support ×

Advanced ☐ Grant access to serial console ?

Create VM Cancel

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	9	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	fhung12bdt2ongrnuu6	...

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	9	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	9	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

Please sign in

admin

Login

5. Choose Two Way Streaming example

Demo

Security

License

v.2.0.226-5.2.1498-99cbd3e97ff22a45e503595b18cedd912e8823f3

Streaming

Streamer

Player

Two-Way Streaming

Firewall Streaming

WebRTC as RTMP

Stream Recording

Several Streams Recording

Stream Server Snapshot

Stream Local Snapshot

Stream Diagnostic

Screen Sharing

Two-way Streaming

Local

Player

4e68

Publish

4e68

Play

Available

{"count": 23}

Send payload as object

wss://158.160.42.178:8443

Connect

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

Flashphoner Web Call Server x +

Not secure https://158.160.42.178:8444/admin/demo.html#

Demo v.2.0.226-5.2.1498-99d

Streaming

Streamer

Player

Two-Way Streaming

Firewall Streaming

WebRTC as RTMP

Stream Recording

Several Streams Recording

Stream Server Snapshot

Stream Local Snapshot

Stream Diagnostic

Screen Sharing

158.160.42.178:8444 wants to

- Use your camera
- Use your microphone

Allow Block

Two-way Streaming

Local

Player

212b Publish

212b Play Available

{ "count": 23 }

Send payload as object

wss://158.160.42.178:8443 Disconnect

ESTABLISHED

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

Two-way Streaming

Local



212b Stop

PUBLISHING

```
{"count": 23}
```

Send payload as object

wss://158.160.42.178:8443 Disconnect

ESTABLISHED

Player



212b Stop Available

PLAYING

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 using one of the standard Linux images as separate media server or low latency streaming CDN node. This way is preferable for long term server instances (from year and more).

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

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

< Folder

Compute Cloud Service

Virtual machines

Disks

Snapshots

Images

Instance groups

Placement groups

Operations

Virtual machines



Create your first VM

Yandex Compute Cloud allows you to use virtual machines in the Yandex.Cloud infrastructure to meet your needs. You can use Yandex Compute Cloud to host your ready-to-use application or development infrastructure, and perform load or functional testing.

You can decide how many cores and disks you need, define your block storage and amount of RAM, and select the operating system and availability zone for your virtual machine.

To get started, just click **Create VM**. For more information about the service, see the documentation:

- [Getting started with VMs](#)
- [Yandex Compute Cloud documentation](#)

Create VM

2. Enter server name, description and choose datacenter region

Create a virtual machine

Basic parameters

Name ?

test-wcs



Description ?

Test WSC server|



Availability zone ?

ru-central1-a



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

Computing resources

Platform ?

Intel Cascade Lake



vCPU

2



2

80

Guaranteed vCPU performance ?

5% 20% 50% 100%

For any task, including high-load services.

RAM

2 GB



2 GB

32 GB

Additional

☐ Preemptible ?

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 	 CentOS 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 	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 ?

support

SSH key ?

DpQFxmFQFaFBVyyXihfFdHoFGFdXx84BUV
Kz35hYonOwsDCvYmVhNTZt4oOYU7t70xR
A5UjcWunPwzXFtikJDgIL0B25QuqObFP9NH
c+ggJmNA91dxUC9q1QY/GCJTIPdbYO9QO
WGmrlXDDrZxroWDqxfXnMX5CEoVelsgS56
WUnYeXqqTGG9mPnwohcsU41 support

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 pulic 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

```

#!/bin/bash
#
export IPT="iptables"

# External interface
export WAN=eth0

# Clean iptables
$IPT -F
$IPT -F -t nat
$IPT -F -t mangle
$IPT -X
$IPT -t nat -X
$IPT -t mangle -X

# Set default policies
$IPT -P INPUT ACCEPT
$IPT -P OUTPUT ACCEPT
$IPT -P FORWARD ACCEPT

# Allow local traffic
$IPT -A INPUT -i lo -s 127.0.0.0/8 -d 127.0.0.0/8 -j ACCEPT
$IPT -A OUTPUT -o lo -s 127.0.0.0/8 -d 127.0.0.0/8 -j ACCEPT

```

```

# Allow outgoing connections
$IPT -A OUTPUT -o $WAN -j ACCEPT

# Allow already established connections
$IPT -A INPUT -p all -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPT -A OUTPUT -p all -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPT -A FORWARD -p all -m state --state ESTABLISHED,RELATED -j ACCEPT

# Enable packet fragmentation
#$IPT -I FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS --clamp-mss-to-pmtu

# Drop invalid packets
$IPT -A INPUT -m state --state INVALID -j DROP
$IPT -A FORWARD -m state --state INVALID -j DROP
$IPT -A INPUT -p tcp ! --syn -m state --state NEW -j DROP
$IPT -A OUTPUT -p tcp ! --syn -m state --state NEW -j DROP

# Allow pings
$IPT -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type destination-unreachable -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type time-exceeded -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type echo-request -j ACCEPT

# Allow SSH
$IPT -A INPUT -p tcp --dport 22 -j ACCEPT
# Allow DNS
#$IPT -A INPUT -i $WAN -p udp --dport 53 -j ACCEPT
# Allow NTP
#$IPT -A INPUT -i $WAN -p udp --dport 123 -j ACCEPT

# Allow WCS ports
$IPT -A INPUT -p tcp --dport 80 -j ACCEPT
$IPT -A INPUT -p tcp --dport 443 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8888 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8443 -j ACCEPT
$IPT -A INPUT -p tcp --dport 1935 -j ACCEPT
$IPT -A INPUT -p udp --dport 1935 -j ACCEPT
$IPT -A INPUT -p tcp --dport 554 -j ACCEPT
$IPT -A INPUT -p tcp --dport 3478 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8080 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8081 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8084 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8082 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8085 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8445 -j ACCEPT
$IPT -A INPUT -p tcp --dport 8444 -j ACCEPT
$IPT -A INPUT -p tcp --dport 10000:50000 -j ACCEPT
$IPT -A INPUT -p udp --dport 10000:50000 -j ACCEPT
$IPT -A INPUT -p tcp --dport 50999 -j ACCEPT

$IPT -A INPUT -j DROP
$IPT -A FORWARD -j DROP

# Save the rules to file
/sbin/iptables-save > /etc/sysconfig/iptables

```

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.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

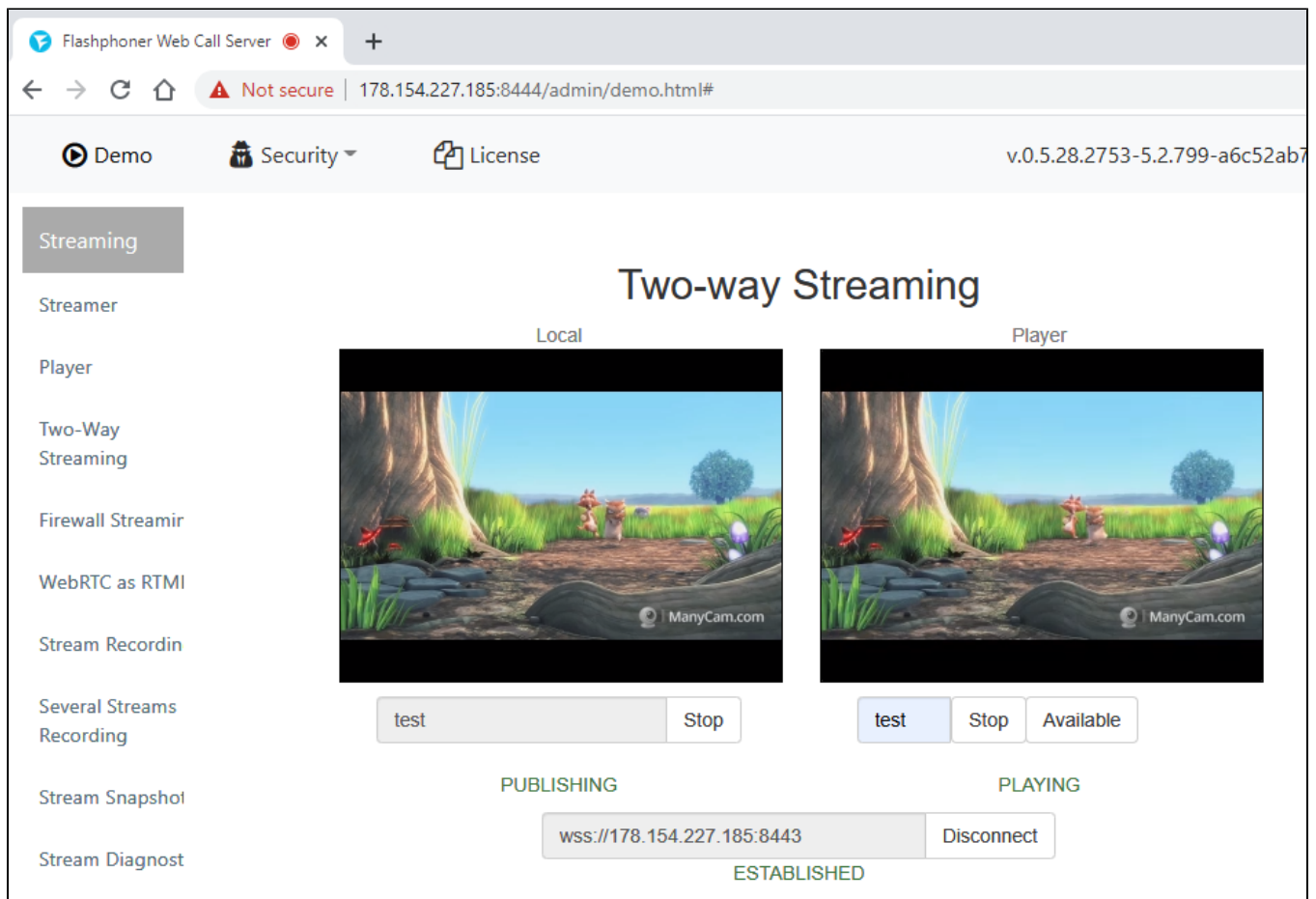
```
cd /usr/local/FlashphonerWebCallServer/bin
sudo ./activation.sh
```

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 instanceld available via API or in EC2 console.

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

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