Installation guide

Installation on AWS

Edition	Changes
1.0	Creation

Content

1 Introduction

1.1 Context

This document describes how to install WebRTC gateway on AWS environment.

- ☞ This document is a guideline that is not intended to replace a skilled AWS administrator.

1.2 Documents reference

Ref.	Title	Source
1		
2		

2 PREREQUIREMENTS

AWS trained, skilled experts and an AWS account (credit card mandatory).
User has appropriate admin rights on AWS.

EC2 instance will be created from official Amazon Machine Image (AMI) Debian 12 provided by Debian (from AWS & trusted third-party AMIs) to benefit from AWS management & interface.

- ☞ Only AMI from Debian is supported by ALE (or a fork of an official Debian AMI).
- ☞ Using Community AMIs (even verified provider) is at your own risks & responsibilities.

3 Launch EC2 instance

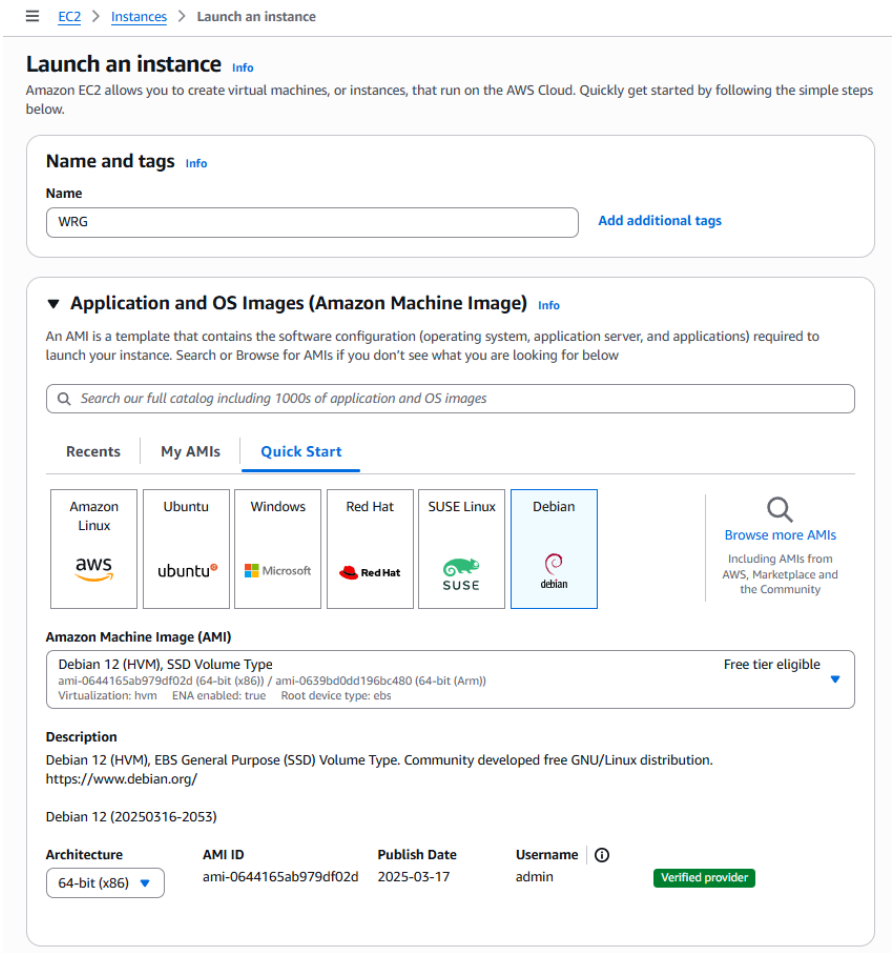
Launch EC2 instance based on Debian 12 AMI 64-bit (x86) provided by Debian (Verified provider) and compatible with your region.

- ☞ Architecture : 64-bit (x86)
- ☞ Select instance type following your needs
- ☞ Manage VPC
- ☞ Subnet
- ☞ Select or create SSH key pair
- ☞ Define private Primary IP according to your VPC range
- ☞ Storage size is 8 Gb minimum by default and cannot be decrease to 4 Gb
  - keep the default value
- ☞ Associate Public IP with Elastic Ips

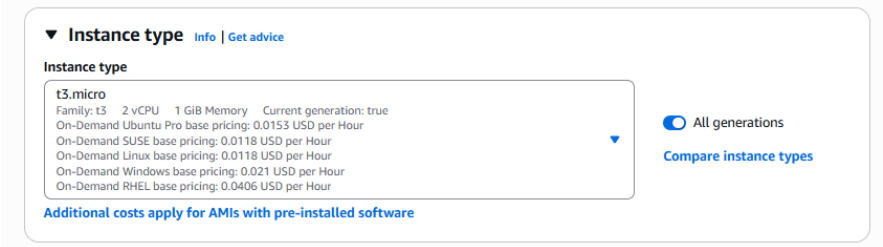
- ☞ Manage all necessary items (VPC, Security groups, Elastic IPs …) with your administrator according to company's policies & best practices.

Create a Snapshot of your instance before installing Rainbow packages.

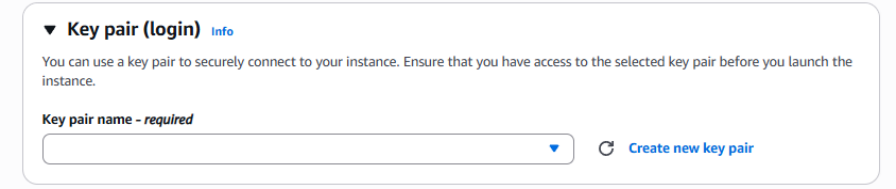
3.1 Instance creation



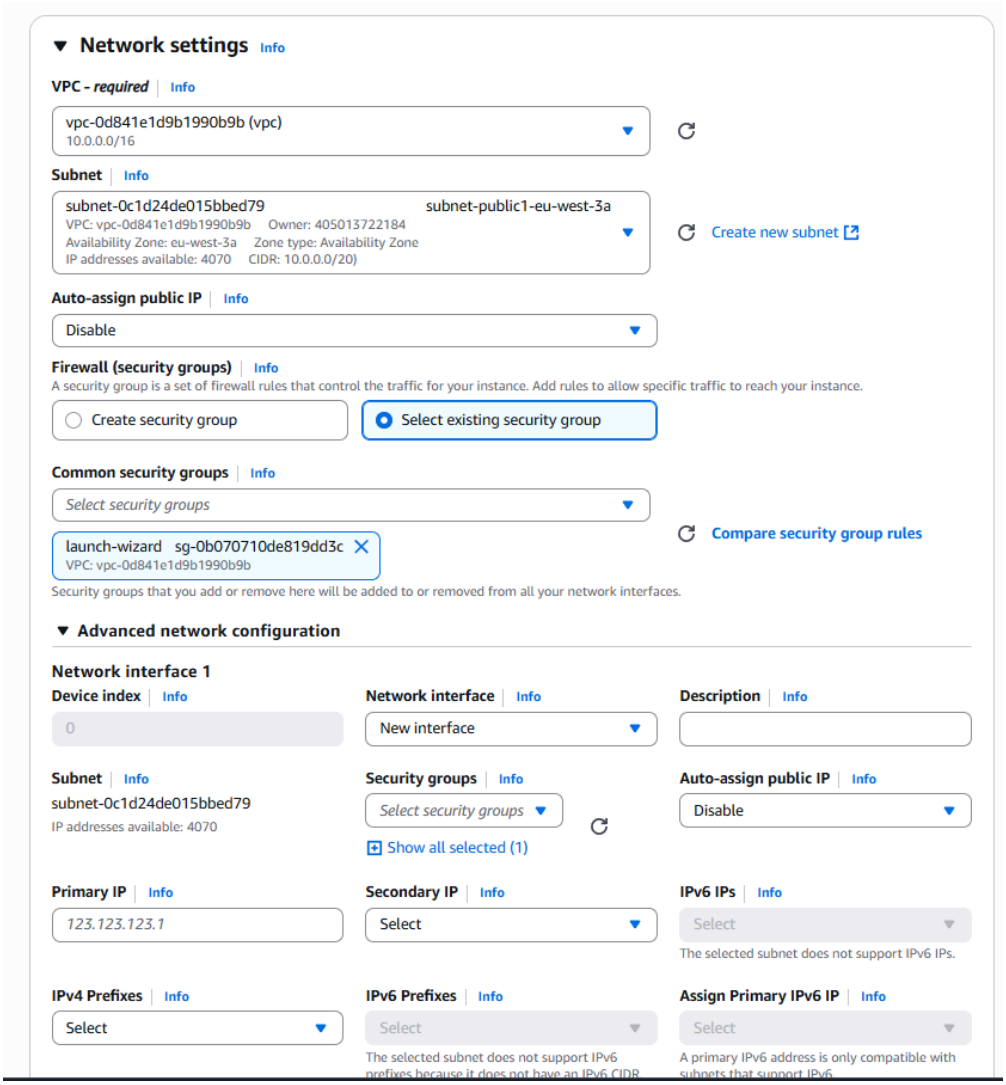
- AMI ID will change depending on region & version
- Note the default username



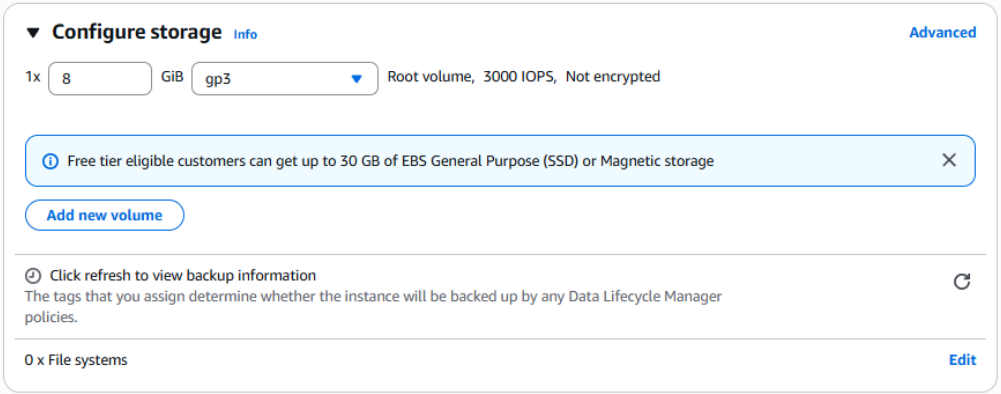
- t3.micro is for lab purpose only
  Minimum memory is 2Gb : adjust values following official ALE document regarding traffic & users



- Select/create ssh key pair



- ☞ Define your static private IP in “Primary IP”.
- ☞ Change any items as needed.



- ☞ Value cannot be less than 8Gb with selected Debian AMI: minimal recommended value is 4Gb of hard disk for WebRTC gateway. So 8Gb is correct.

Launch instance

- Wait until EC2 image is “Running”.

3.2 Create public IP address

Using “Elastic IPs” menu, create & associate a public IP address to the new EC2 instance.

- Your instance should now be accessible by ssh.
- Configure you ssh client to use the ssh key defined during instance creation.
- Configure your VPC to allow incoming ssh connection on public IP address.

4 Connect to EC2 instance

- These first steps are fully under AWS administrator control and are given as guideline only.

Using ssh connection, log to instance

- Default login is admin for official Debian AMI image

Before starting Rainbow WebRTC gateway installation, proceed with image update/upgrade:

- sudo apt update
- sudo apt list --upgradable
- sudo apt upgrade -y

Remove unnecessary packages

- sudo apt autoremove
- check & remove

Check number of linux versions:

- dpkg --list | grep linux-image
  ex:

```
rc  linux-image-6.1.0-32-cloud-amd64 6.1.129-1          amd64       Linux 6.1 for x86-64 cloud (signed)
ii  linux-image-6.1.0-34-cloud-amd64 6.1.135-1          amd64       Linux 6.1 for x86-64 cloud (signed)
ii  linux-image-6.1.0-37-cloud-amd64 6.1.140-1          amd64       Linux 6.1 for x86-64 cloud (signed)
ii  linux-image-cloud-amd64          6.1.140-1          amd64       Linux for x86-64 cloud (meta-package)
```

- Check running image
  uname -r
  6.1.0-34-cloud-amd64
  - Reboot your image to be able to remove unneeded images (32 & 34 here).

- After reboot
  uname -r
  6.1.0-37-cloud-amd64

- Remove unused images
  In our case:
  sudo apt remove --purge linux-image-6.1.0-34-cloud-amd64
  sudo apt remove --purge linux-image-6.1.0-32-cloud-amd64

- Update boot grub
  sudo update-grub
  sudo reboot

5 Snapshot

Create a snapshot of EC2 instance or volume.

6 Deploy rainbow packages

Download Rainbow package installer for AWS "Rainbow_AWS_Installer-xxxx.tar" in /tmp.

- Note that the /tmp directory is emptied after each reboot. You can use another working folder.

From /tmp, unzip packages:

tar -xvf Rainbow_AWS_Installer-xxxxxx.tar

From /tmp folder, install packages:

./Rainbow_AWS_Installer.sh

- You can log detailed operations by:
  bash -x Rainbow_AWS_Installer.sh (see Troubleshooting)

When installation is finished, reboot EC2 instance.

- sudo reboot

7 Rainbow configuration

7.1 login

After reboot, log in via ssh using the **rainbow** account.

- Default admin account can be used as backup or can be disabled (be sure to have another sudo account in this case).

Rainbow config file is allowing ssh (based on key) connection by default for AWS.

- You should have a warning message when login with the rainbow account:
  [WARNING] SSH server is active

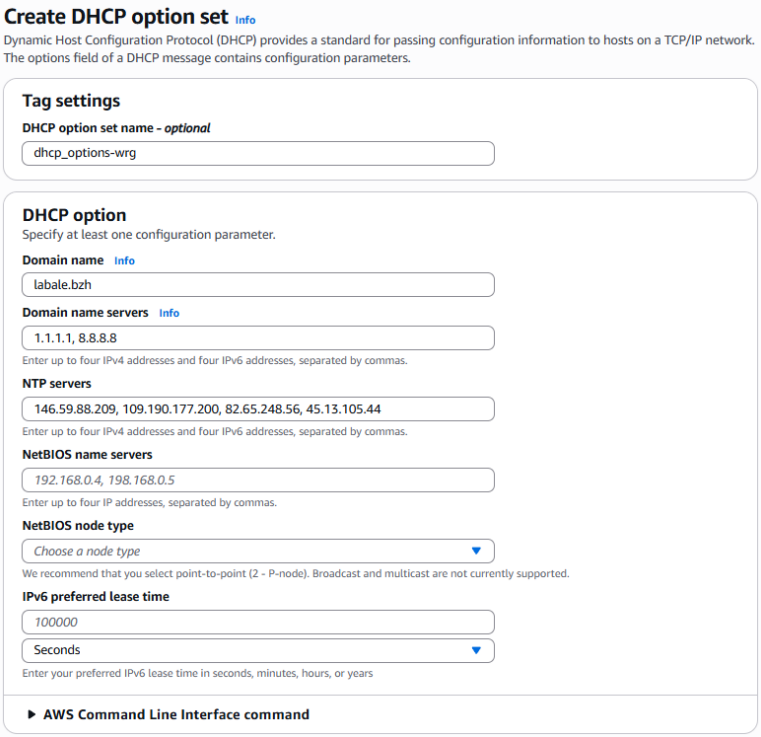
- You can disable ssh connection, type: mpssh off
- Note that ssh connection is using same key as admin account.
- You cannot connect through ssh with login/password.
- Login/password is used only for console connection.

7.2 Network parameters

Private IP address is affected during EC2 instance creation. This private IP is reserved until instance is terminated.
This private IP **cannot** be changed afterward.
For that reason, WebRTC gateway will be used in DHCP only at application level. Static private IP is managed at EC2 instance level.

- Only hostname & proxy settings can be changed with mpnetwork (all other options are ignored).
- Domain name, DNS servers and NTP servers values should be managed with "DHCP options sets" in VPC (Virtual Private Cloud) associated to your EC2 instance.

Ex:



☞ These values are reflected to EC2 instance with mpshow:

```
DHCP informations:

Current IP: 10.0.10.30
DHCP lease:
Link state: Link detected: yes

Latest informations found in lease file:

IP:         10.0.10.30
NETMASK:    255.255.240.0
GATEWAY:    10.0.0.1
DNS:        1.1.1.1 8.8.8.8
NTP:        146.59.88.209
HOSTNAME:   ip-10-0-10-30
HOSTDOMAIN: labale.bzh
```

7.3 Rainbow configuration

Configure Rainbow settings:

- mpconfig –PBXID=xxxx-xxxx-xxxx-xxxx -PBX_DOMAIN=<pabx FQDN or IP>

Force services restart to enable all necessary services.

- mprestart

Check settings:

- mpshow
  Check your configuration

- mpstatus

```
Rainbow WebRTC Gateway version: <version>
<date>
mp-monitor status ...
  [OK] enabled/active
kamailio status ...
  [OK] enabled/active
janus-gateway-mediapillar status ...
  [OK] enabled/active
otlitemediapillargateway status ...
  [OK] enabled/active
systemd-timesyncd status ...
  [OK] enabled/active
registration status ...
  traceroute -n -4 -T -p 5060 <PBX_DOMAIN>
  [OK] route <local private IP>
rainbow auth status ...
  [OK] [OK]
kamailio registered users ...
                "Address":  "sip:BBB@<local private IP:port>;transport=udp",
rainbow registered users ...
```

- mpcheck

```
Rainbow WebRTC Gateway version: <version>
<date>
Network settings ...
  [OK]
Rainbow settings ...
  [OK]
DNS test resolution of 'openrainbow.net' ...
  [OK] dnstest host (24 ms) 51.178.224.139 51.178.224.136
Rainbow connect ...
  [OK] Rainbow connection (openrainbow.net) can be established
  [OK] Rainbow connection (51.178.224.139) can be established
  [OK] Rainbow connection (51.178.224.136) can be established
Rainbow TLS check ...
  [OK] Rainbow TLS check to openrainbow.net
STUN/TURN test will be done using GEOIP config
DNS test resolution of 'turn-ipv6.openrainbow.net' ...
  [OK] dnstest turn (36 ms) 51.178.89.150
STUN/TURN test turn-ipv6.openrainbow.net ...
  51.178.89.150-u-3478 ... /opt/mediapillar/cv/mpstc  -u -s20000 -a51.178.89.150 -d3478 -w2
    [ok] 51.178.89.150:u:3478-51.44.100.60:20229
  [OK] stun/turn test (19 ms)
DNS test resolution of 'turn-rbx1.openrainbow.net' ...
  [OK] dnstest turn (38 ms) 51.178.224.132
STUN/TURN test turn-rbx1.openrainbow.net ...
  51.178.224.132-u-3478 ... /opt/mediapillar/cv/mpstc  -u -s20000 -a51.178.224.132 -d3478 -w2
    [ok] 51.178.224.132:u:3478-51.44.100.60:20302
  [OK] stun/turn test (29 ms)
PBX_DOMAIN access ...
  ping -4 -c 2 <PBX_DOMAIN>
  [OK] ping <PBX_DOMAIN>
PBX_DOMAIN SIP OPTIONS ...
  siptest ... /opt/mediapillar/cv/mpstc -S -a<PBX_DOMAIN> -d5060 -w2
  [OK] success
registration access ...
  traceroute -n -4 -T -p 5060 <local private IP>
```

[OK] route <local private IP>

- mpsystem
  - check that Amazon is recognized:
    machine: Amazon EC2

If all is correct, your system is ready to use.

8 Upgrade

8.1 Upgrade from

This is the same process as traditional WebRTC gateway:

- Download iso + iso.md5 files in /upgrade folder
- Upgrade manually with mpupgrade command
  mpupgrade [--now|--delay=XXX[s|m|h]] [--yes] [--abort] [--cleanup]

8.2 Remote upgrade

This is the same process as traditional WebRTC gateway using admin web interface:

- Remote download of available version
- Upgrade process

9 Limitation/restrictions

The following features are not supported with AWS:

- DO NOT manage multiple network interfaces (eni)
- Duolan mode is not supported

10 Firewall consideration

If your EC2 instance for WebRTC gateway has a public IP address, you can allow direct RTP between your Rainbow user (over internet) and WebRTC gateway's public IP.

- In that case, you will not pass through TURN servers.

11 Troubleshooting

If installation fails, please do following operations:

1) Come back to previous state:

☞ Use your snapshot (recommended solution to have clean disk)

☞ Uninstall all components:
./Rainbow_AWS_Installer.sh –undoall
sudo reboot

2) Install rainbow as follow:
bash -x Rainbow_AWS_Installer-xxxx.sh

☞ Save results of operations in a log file & send it to ALE support.

If remote upgrade fails, create an empty file before launching upgrade with admin web interface:

touch /tmp/tracelog

☞ This will create /temp/ folder with some log files.

Save /temp/*.log files & send it to ALE support.