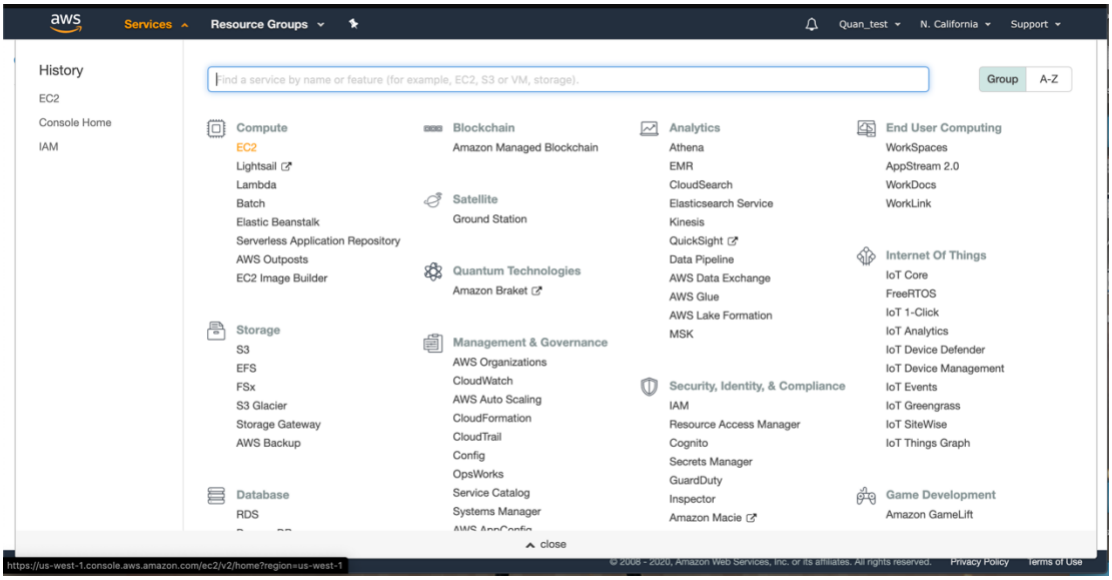
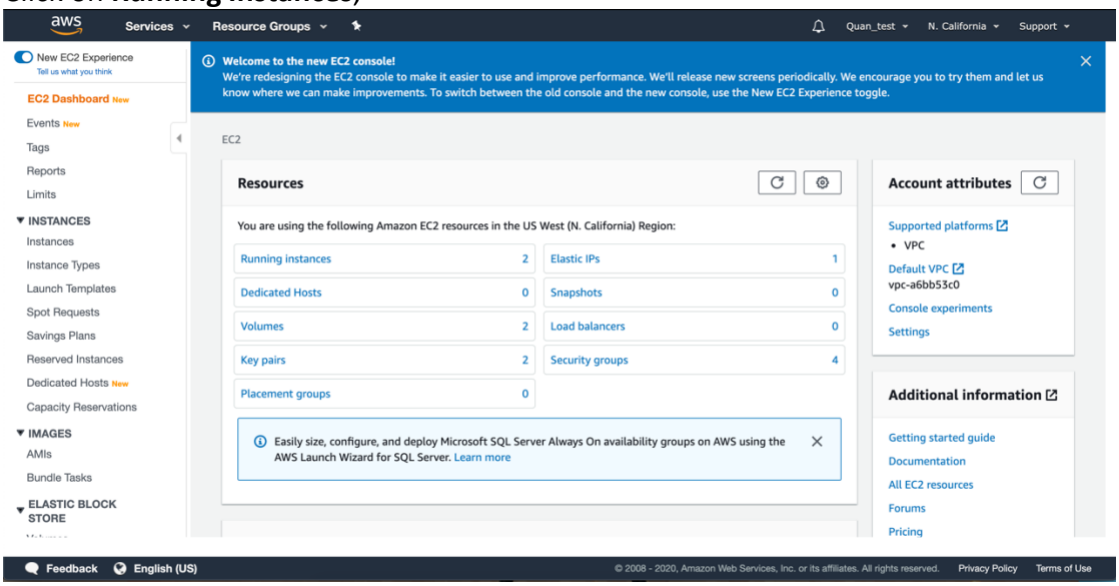
This document is about how-to deploy and run EasyRTC on Amazon Web Service (AWS)

1. Log in to your Root AWS account

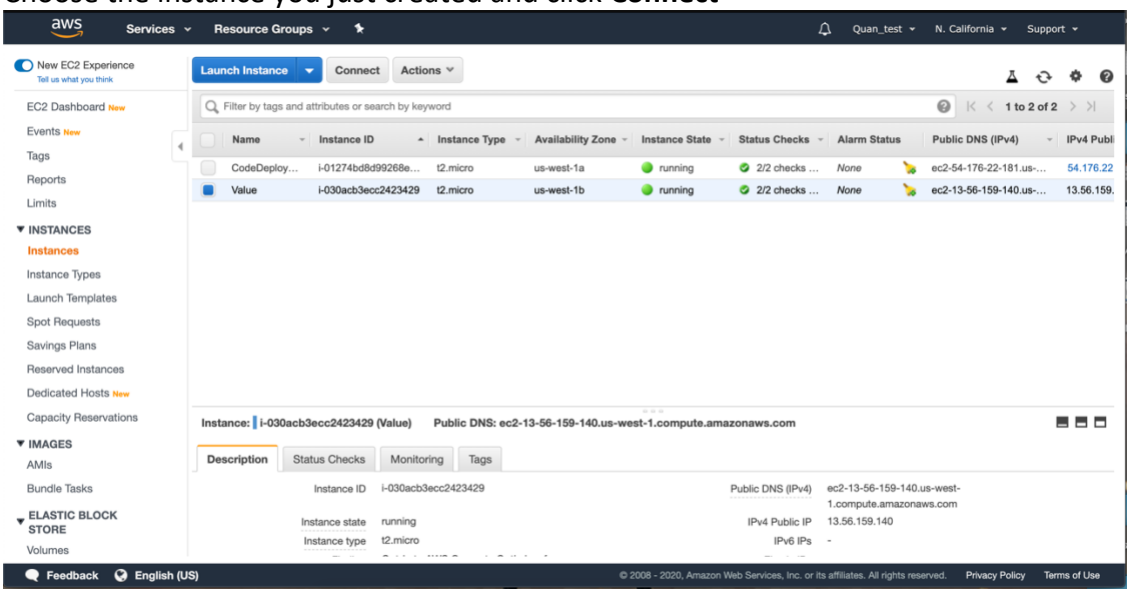
2. Click on **Services** and choose **EC2**,



3. Click on **Running Instances**,

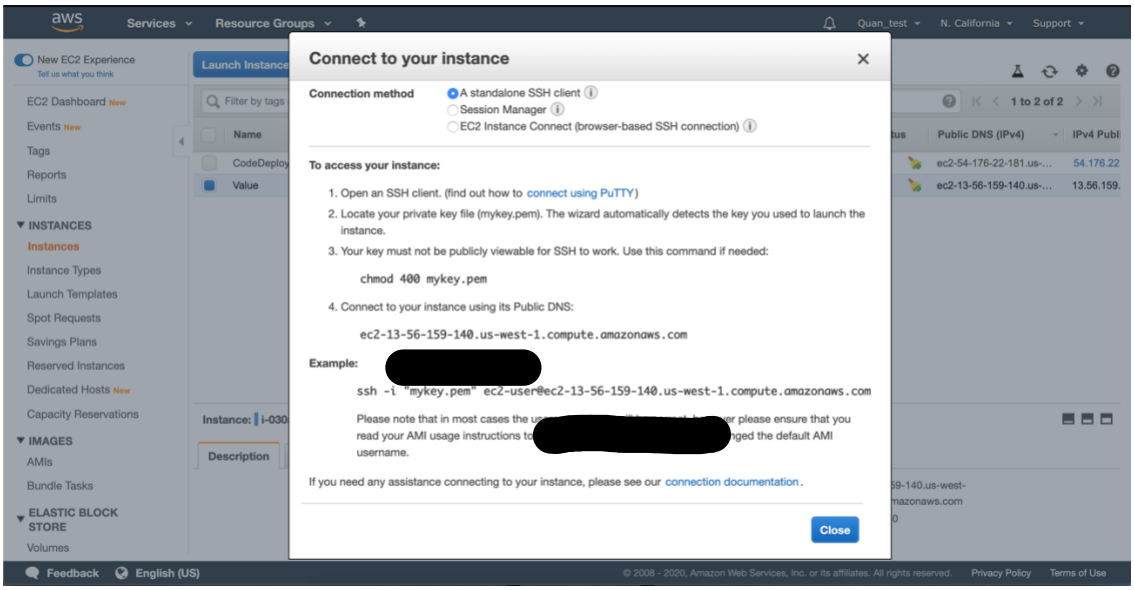


4. Choose the instance you just created and click **Connect**

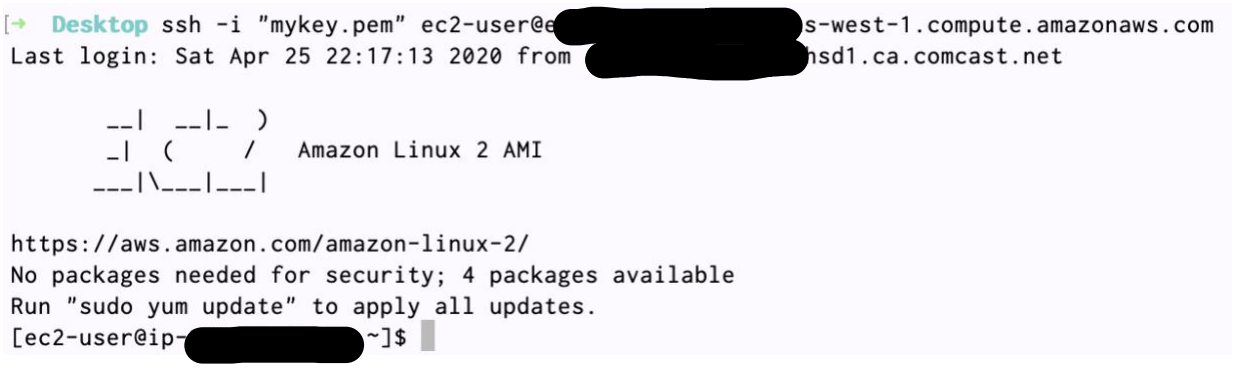


5. Copy the code below **Example:**
The code should be similar to this

```
ssh -i "path to your certificate" ec2-user@ec2-xx-xx-xxx-xxx.us-west1.compute.amazonaws.com
```



6. Go to your terminal and execute this line of code you just copied, and you should be able to login to your AWS EC2 instance as shown below



7. Run the code below to install essential tools for the application

   Update:

   ```
   sudo yum update
   ```

   Install Git:

   ```
   sudo yum install git
   ```

   Install Node:

   ```
   curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.34.0/install.sh
   | bash

   . ~/.nvm/nvm.sh

   nvm install node
   ```

8. Download the source code from open-easyrtc on github using the code below:

   ```
   git clone https://github.com/open-easyrtc/open-easyrtc
   ```

9. Change directory to open-easyrtc,

   ```
   cd open-easyrtc
   ```

   - Install Node modules for the application:

     ```
     npm install
     ```

   - change directory to server_example:

     ```
     cd server_example
     ```

   - install node modules again:

     ```
     npm install
     ```

10. modify server.js so it establishes connection using HTTPS protocol, (note: EasyRTC only ran full functionalities on localhost or https).

    - Open server.js

      ```
      vi server.js
      ```

    - Modify it so it look the same as this one

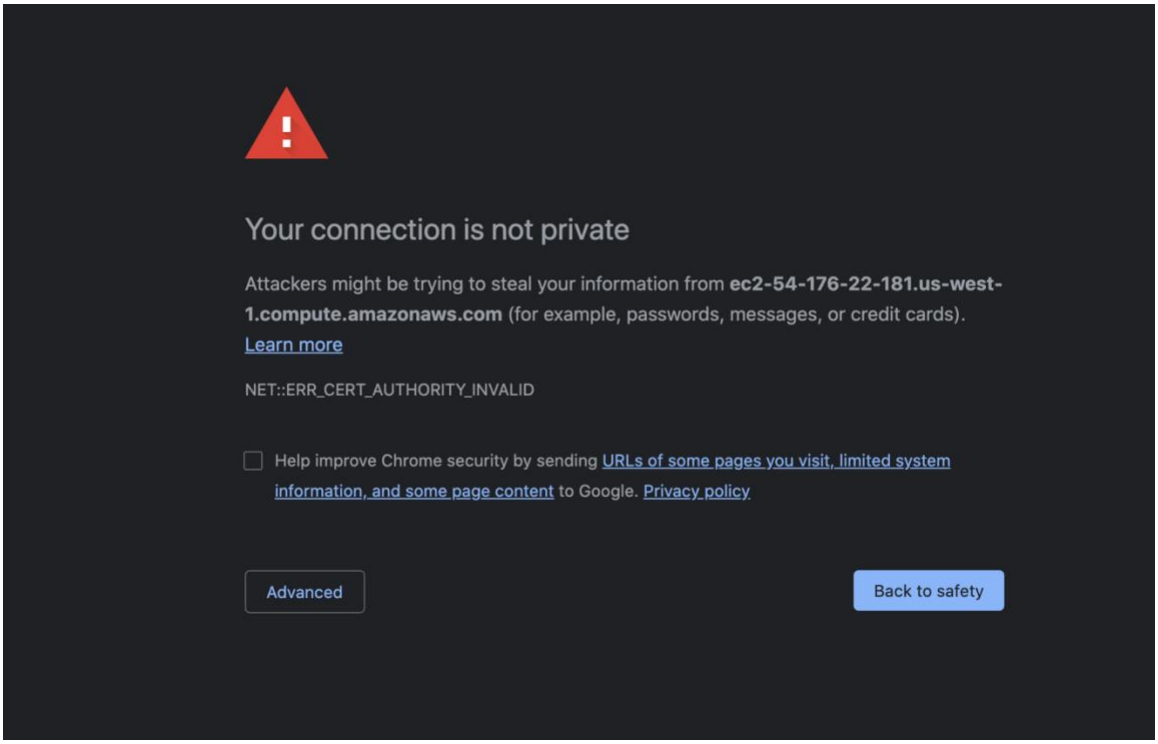
      ```
      // Load required modules
      ```

```
var https   = require("https");     // https server core module
var fs      = require("fs");        // file system core module
var express = require("express");           // web framework external module
var serveStatic = require('serve-static');  // serve static files
var socketIo = require("socket.io");        // web socket external module

// This sample is using the easyrtc from parent folder.
// To use this server_example folder only without parent folder:
// 1. you need to replace this "require("../");" by "require("open-
easyrtc");"
// 2. install easyrtc (npm i open-easyrtc --save) in
server_example/package.json

var easyrtc = require("../"); // EasyRTC internal module

// Set process name
process.title = "node-easyrtc";

// Setup and configure Express http server. Expect a subfolder called
"static" to be the web root.
var app = express();
app.use(serveStatic('static', {'index': ['index.html']}));

// Start Express http server on port 8080
var webServer = https.createServer(
  {
    key:  fs.readFileSync("certs/localhost.key"),
    cert: fs.readFileSync("certs/localhost.crt")
  },
app);

// Start Socket.io so it attaches itself to Express server
var socketServer = socketIo.listen(webServer, {"log level":1});

easyrtc.setOption("logLevel", "debug");

// Overriding the default easyrtcAuth listener, only so we can directly
access its callback
easyrtc.events.on("easyrtcAuth", function(socket, easyrtcid, msg,
socketCallback, callback) {
    easyrtc.events.defaultListeners.easyrtcAuth(socket, easyrtcid, msg,
socketCallback, function(err, connectionObj){
        if (err || !msg.msgData || !msg.msgData.credential
|| !connectionObj) {
            callback(err, connectionObj);
            return;
        }

        connectionObj.setField("credential", msg.msgData.credential,
{"isShared":false});

        console.log("["+easyrtcid+"] Credential saved!",
connectionObj.getFieldValueSync("credential"));

        callback(err, connectionObj);
    });
});

// To test, lets print the credential to the console for every room join!
easyrtc.events.on("roomJoin", function(connectionObj, roomName,
roomParameter, callback) {
    console.log("["+connectionObj.getEasyrtcid()+"] Credential retrieved!",
connectionObj.getFieldValueSync("credential"));
```

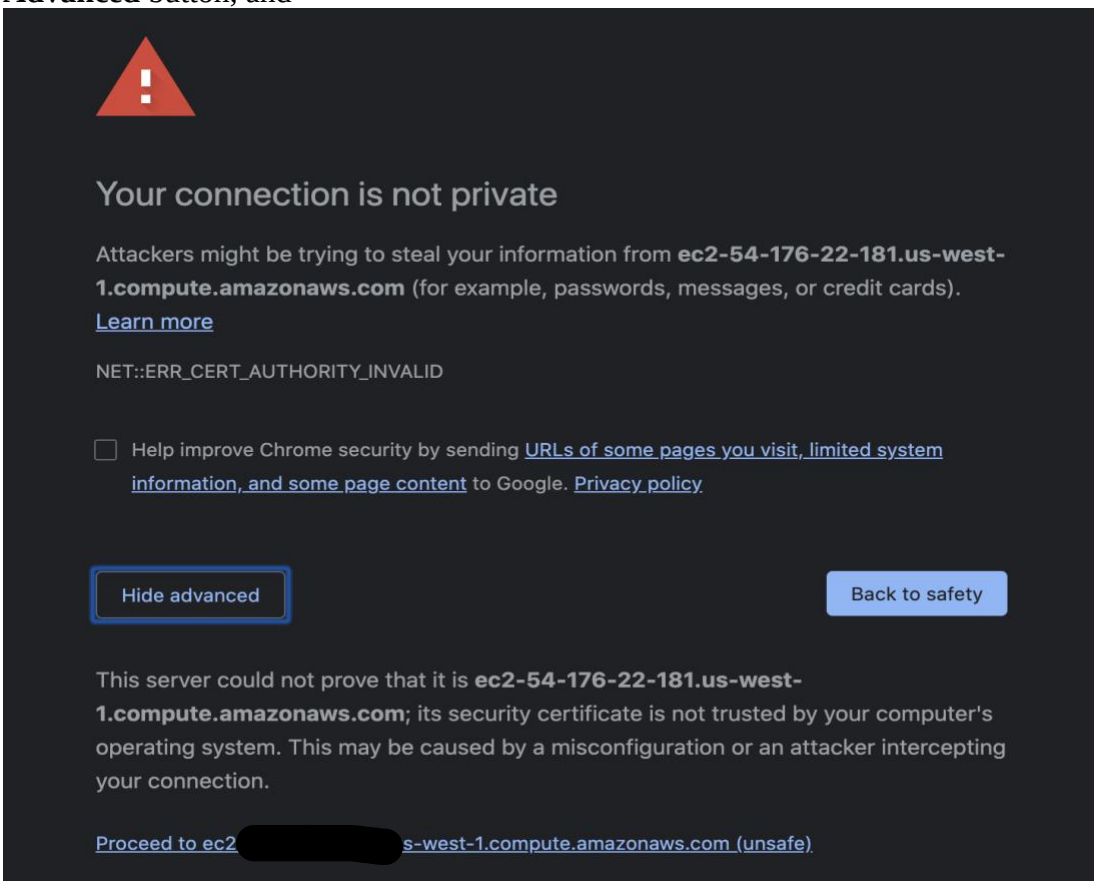
```
    easyrtc.events.defaultListeners.roomJoin(connectionObj, roomName,
roomParameter, callback);
});

// Start EasyRTC server
var rtc = easyrtc.listen(app, socketServer, null, function(err, rtcRef) {
    console.log("Initiated");

    rtcRef.events.on("roomCreate", function(appObj, creatorConnectionObj,
roomName, roomOptions, callback) {
        console.log("roomCreate fired! Trying to create: " + roomName);

        appObj.events.defaultListeners.roomCreate(appObj,
creatorConnectionObj, roomName, roomOptions, callback);
    });
});

// Listen on port 8080
webServer.listen(8443, function () {
    console.log('listening on http://localhost:8443');
});
```

11. Run `node server.js,` and
    go to your own ec2 website,
    This website is similar to this one:
    https://ec2-xx-xx-xx-xxx.us-west-1.compute.amazonaws.com:8443/demos/

    if you see something like this, don't panic



It’s because I used self-signed certificate to establish https connections, just click **Advanced** button, and



And click **Proceed** to your website

Congratulation! you have installed easyRTC on AWS!

You have installed EasyRTC!

This is your server's Web Root

This is the easiest location to put your own static html files for developing WebRTC applications.

In the meantime we'll forward you to the EasyRTC Demo page...