Setting up Windows Server 2016

Documentation process of setting up the server

Author: Roberto Sanchez (014587792)
Date: November 06, 2017
Revision: 1

Table of Contents

Amazon Web Services - Creating Server

After creating your Amazon Web Services (AWS) account.

---

1. Getting Started
    a. To get started, log into Amazon Web Services (AWS)
    b. Then on AWS Services click on **EC2** under Compute table. (Fig 1)

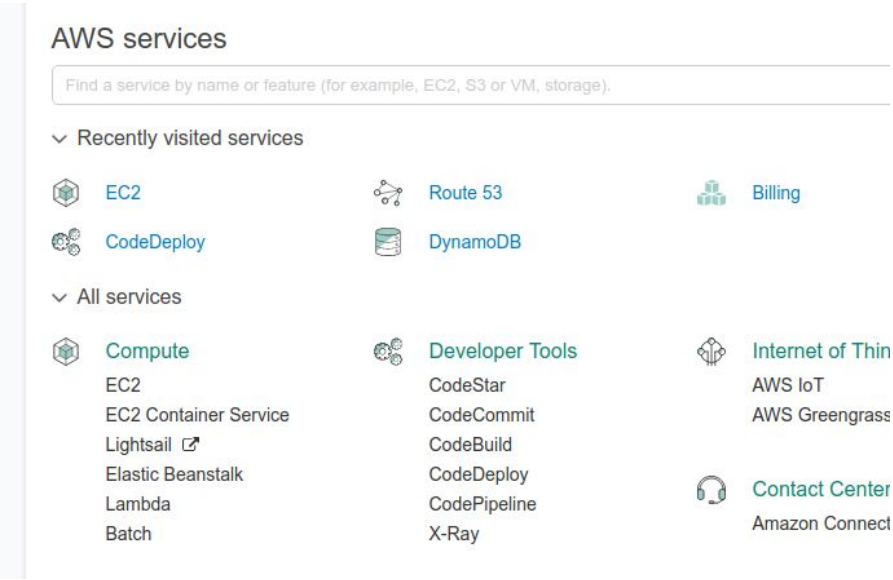


Fig 1: Select EC2 under the Compute

    c. This should bring you to the EC2 Dashboard

    d. On the EC2 Dashboard, click on **Launch Instance** (Fig 2)

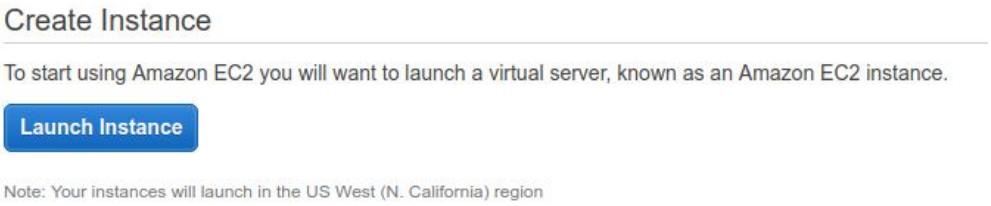
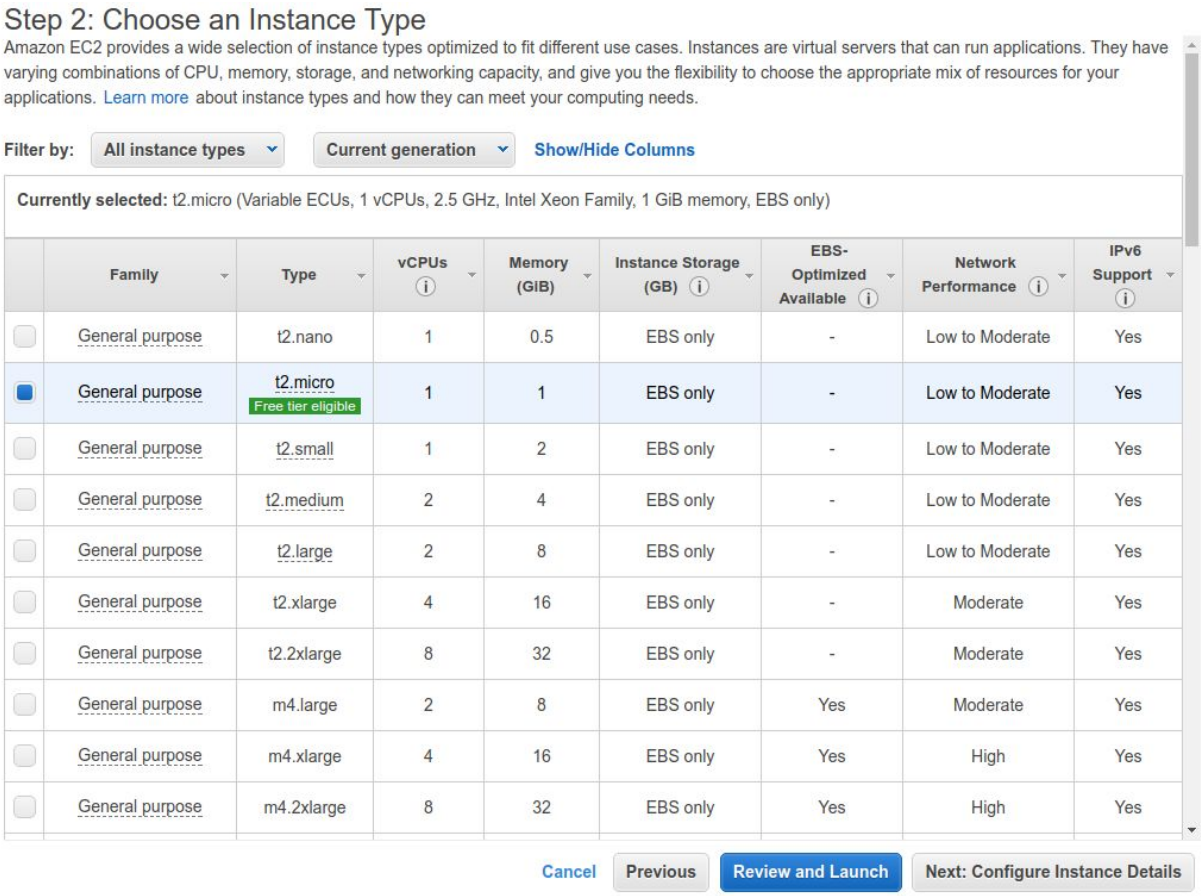


Fig 2: This should be on the EC2 Dashboard in the middle

2. Creating an EC2 Instance of a Windows Server
    a. Select Microsoft Windows Server 2016 Base (64-Bit)

Fig 3: Pick this one

b. Choosing an Instance Type (**Should be the free tier**):
    i. Click on **Review and Launch** after you selected this



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	General purpose	t2.micro (Free tier eligible)	1	1	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
☐	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
☐	General purpose	m4.large	2	8	EBS only	Yes	Moderate	Yes
☐	General purpose	m4.xlarge	4	16	EBS only	Yes	High	Yes
☐	General purpose	m4.2xlarge	8	32	EBS only	Yes	High	Yes

Cancel | Previous | Review and Launch | Next: Configure Instance Details

Fig 4 : Listing the types of instances (We want free)

c. You Should see a page like this after clicking on **Review and Launch**

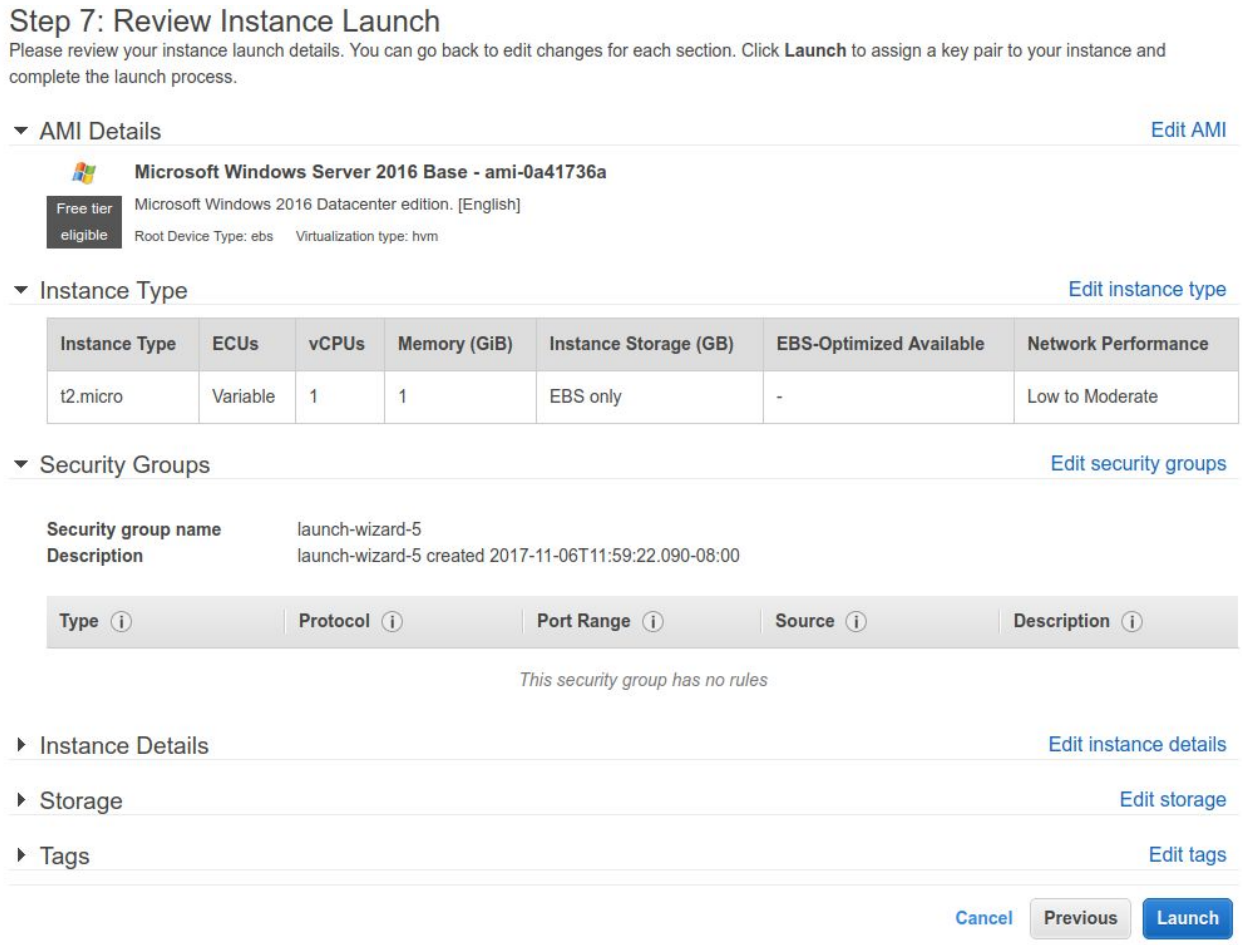


Fig 5: Ignore the Step 7 label

d. Enabling Security Groups (Enables your website to be visible on browser)

i. On the **Step 7 page** click on **Edit Security Groups**

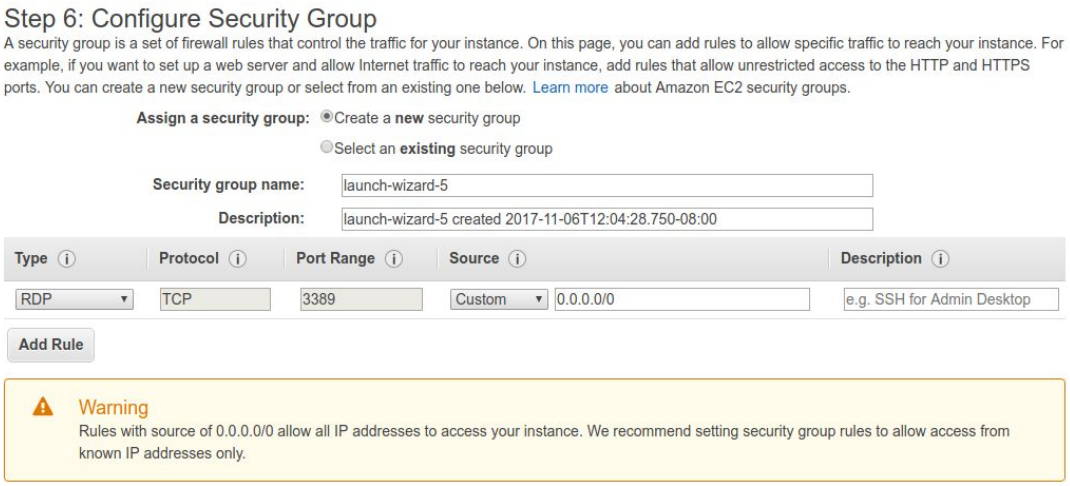


Fig 6 : You should see this

ii. Click on **Add Rule**
iii. It should add another row below the already existing row containing RDP as a type
iv. For Type select HTTP and another rule for HTTPS

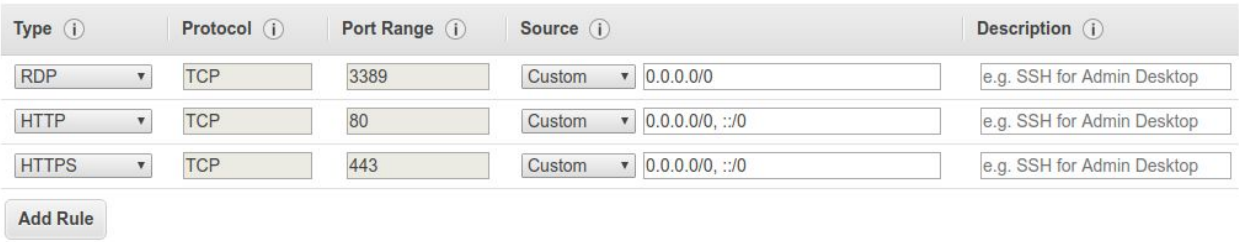


Fig 7: Your rules should look like this

v. Click on **Review and Launch**

e. Creating a PEM key
   i. Once you hit Review and Launch it will prompt your for creating a PEM key
   ii. Name it and Download Key Pair and save it in a safe location

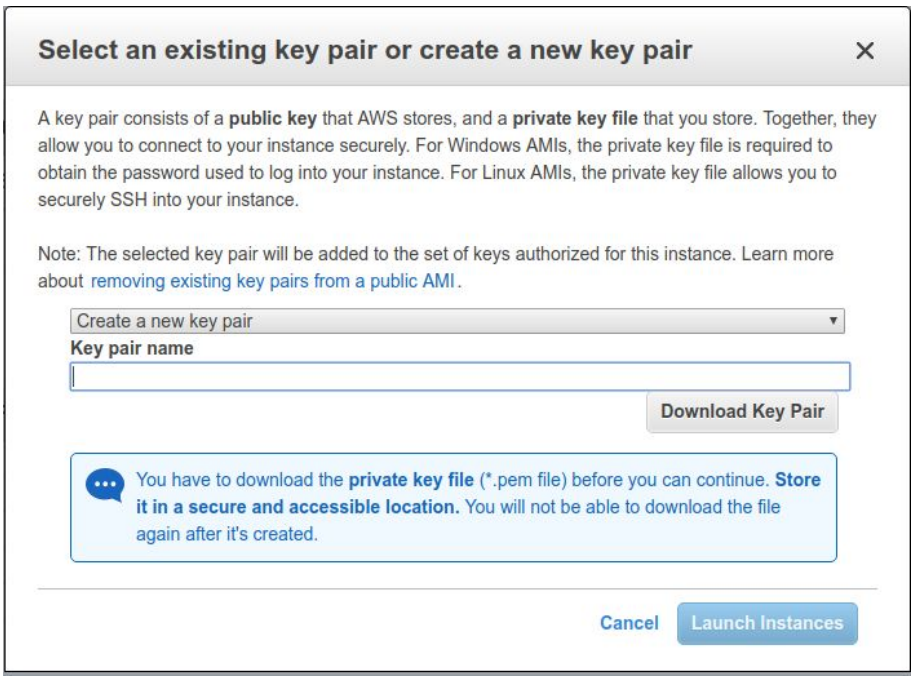


Fig 8: Give it a name -> Download Key Pair -> **Launch Instances**

f. Click on **Launch Instance**
g. **Done**: Your server is being set up right now. Click on **View Instances** to see a list of your servers. (Click on the empty space under **Name** and give it a name)

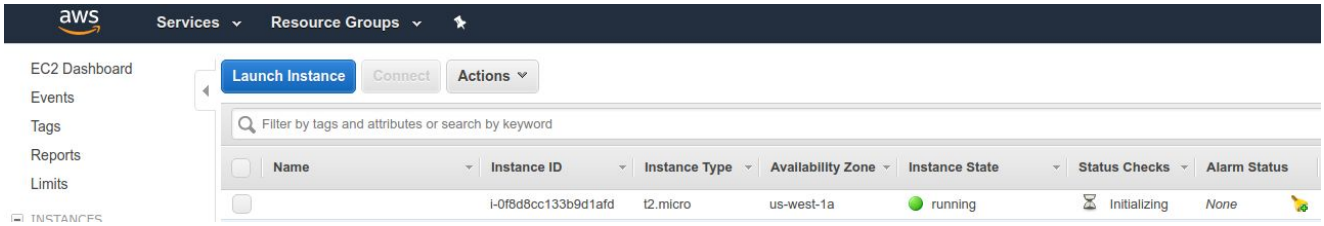


Fig 9: What you should see if all went right

Amazon Web Services - Connecting to Server

This part gets you connected to your windows server.

---

1. Logging into server
    a. Use RDP Client to connect to server
        i. For WINDOWS 10
        Use remote desktop (already preinstalled)
        ii. For MAC OS X
        Install a RDP client from app store such as Microsoft Remote Desktop 8.0, which you can get here
        iii. For LINUX (Specifically Ubuntu 16.04)
        Install Remmina, which you can get it here

2. Credentials
    a. Username: **Administrator**
    b. IP Address:
        i. On AWS -> EC2 Manager-> instances, select your server from the list (The one we created)
        ii. On the bottom there should be a bunch of information
        iii. Its **IPv4 Public IP**: XX.XXX.XXX.XXX

Fig 10: This is located on the bottom right

    c. Password
        i. On the same page (EC2 Manager -> Instances), Click connect (With your server selected)

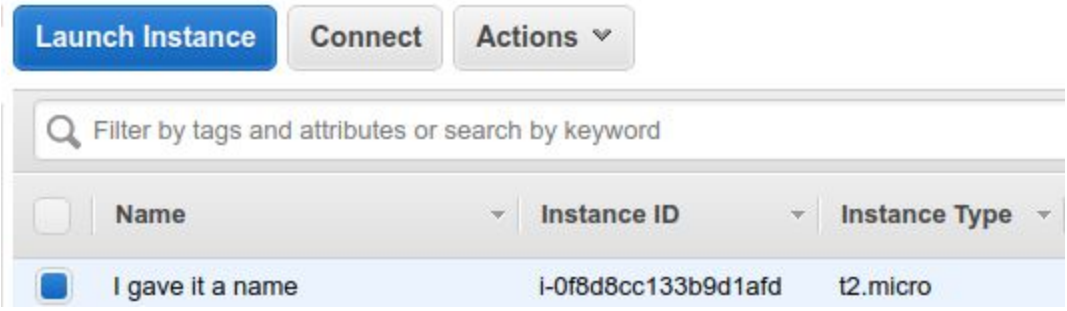


Fig 11: I gave it a name

ii. This will pop up

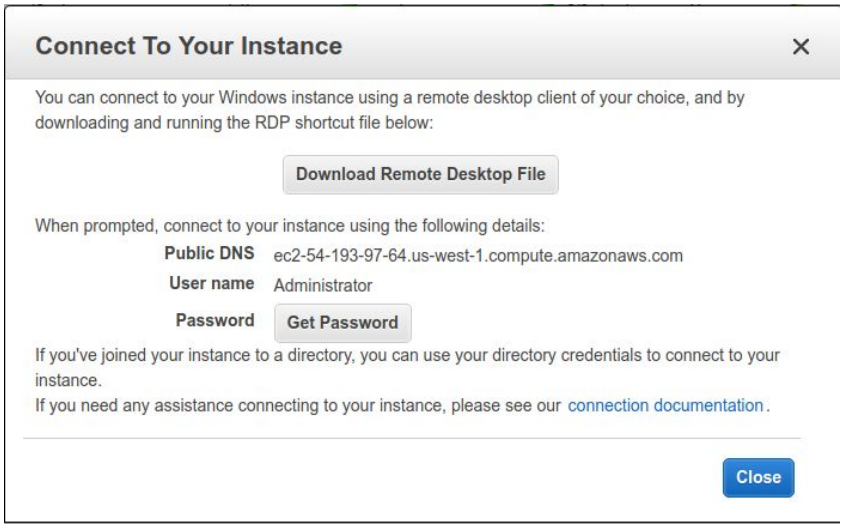


Fig 12: Getting your password to login

iii. Click on **Get Password** and it will prompt you to upload your PEM Key

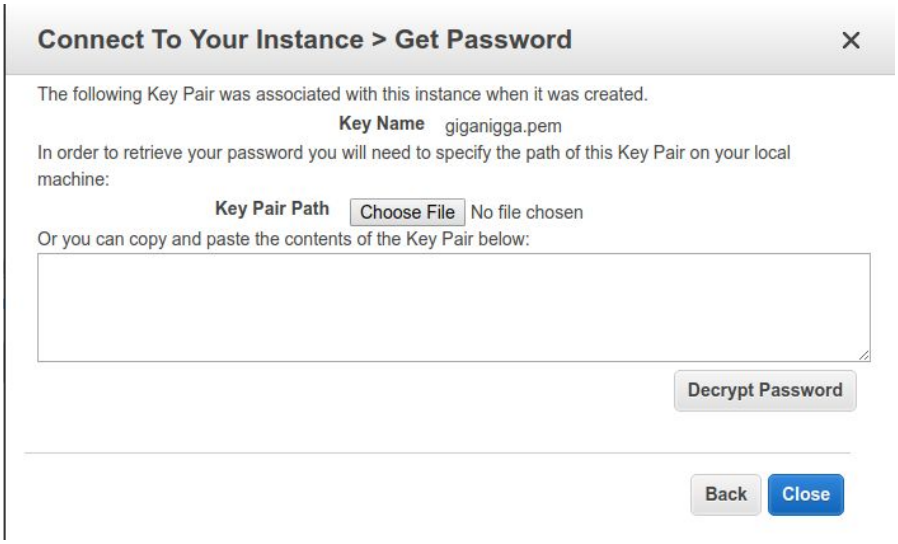


Fig 13: Actually getting your password

iv. Click on **Choose File** and select your PEM Key that you downloaded earlier

v. Once that click on **Decrypt Password**

vi. What you see there is the password for you to connect to the server
vii. Save it somewhere safe and click **Close**

d. At this point you can open your RDP client and enter the IP address of the computer you want to connect to along with the Username and password you have.
e. Done you should see a windows loading up and connecting. It should bring you to a desktop.

Fig 14: Your web server on windows

Windows Server 2016 - Installing/Setting Up Internet Information Services 10 (IIS 10)

Insert something useful

---

1. Installing IIS
    a. Login into the Windows Server
    b. Go to **Start-> Server Manager**

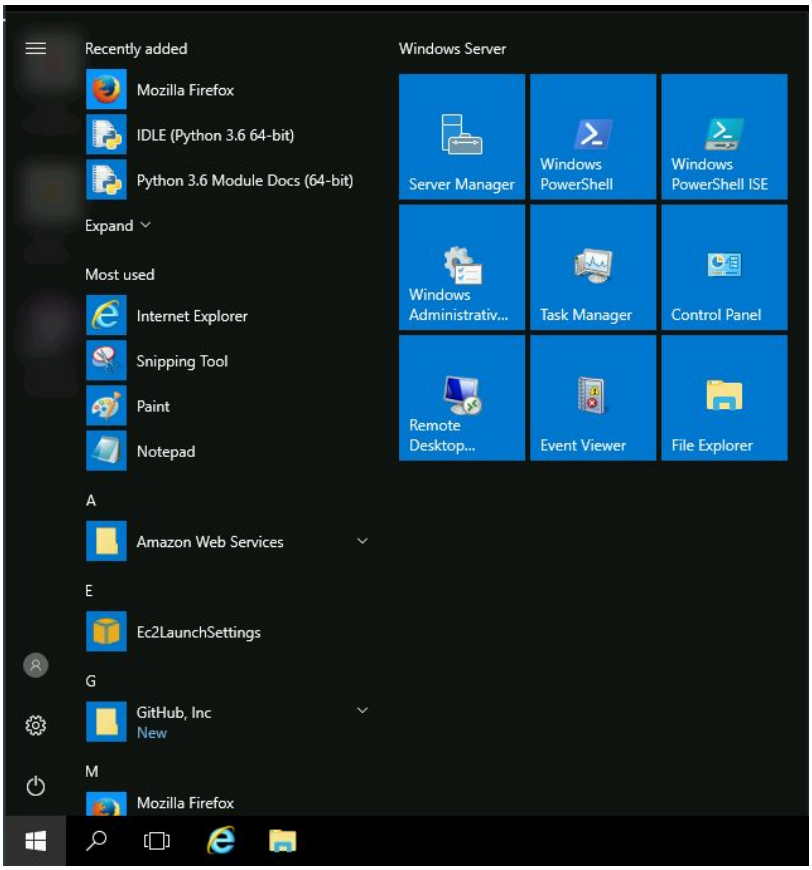


Fig 15: It's the Top left Tile

c. There click on **add roles and features**

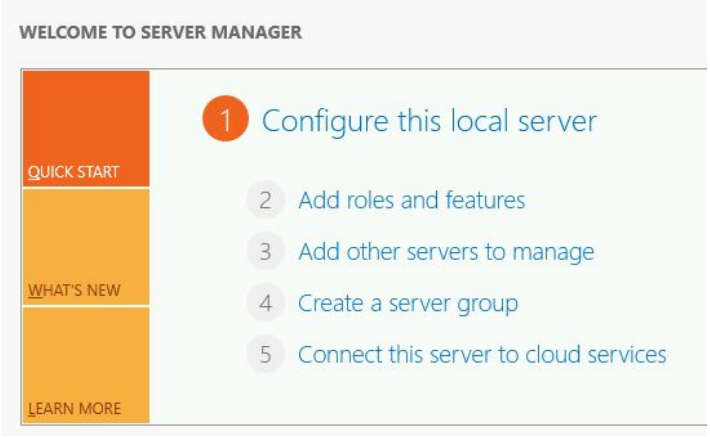


Fig 16: Inside Server Manager, focusing on the welcome screen

d. A window will pop up. Click **Next**.

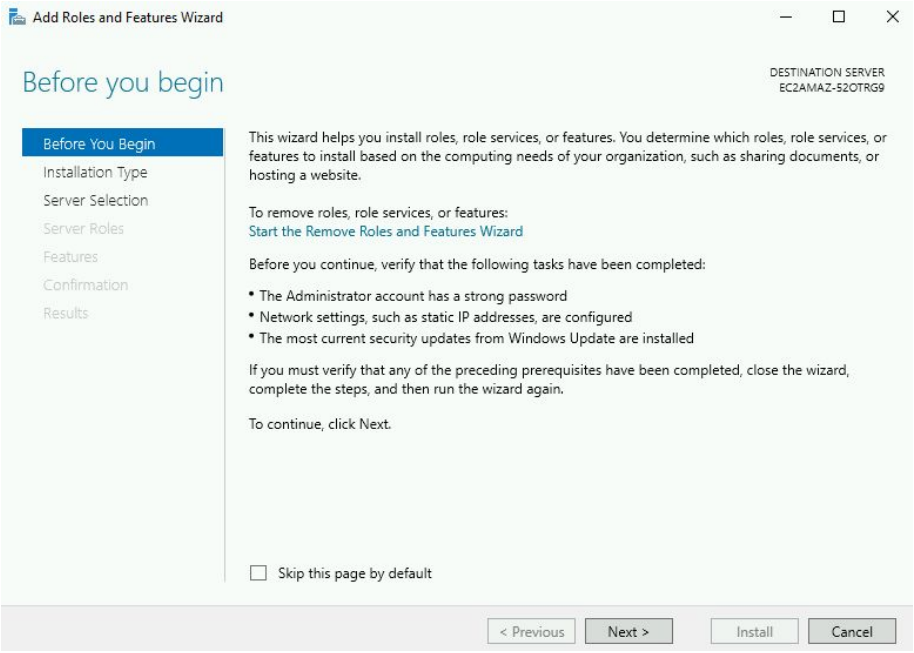


Fig 17: First three sections click next

e. Installation Type: **Role-based or feature-based installation** will be pre-selected, Click **Next**

f. Server Selection: **Select a server from the server pool** will be pre-selected. In the server pool area there should only be one server highlighted. Click **Next**

g. Server Roles: In the section you need to scroll all the way down and check off **Web Server (IIS)** and **File and Storage Services->File and iSCSI Services-> Work Folders** Click **Next.**

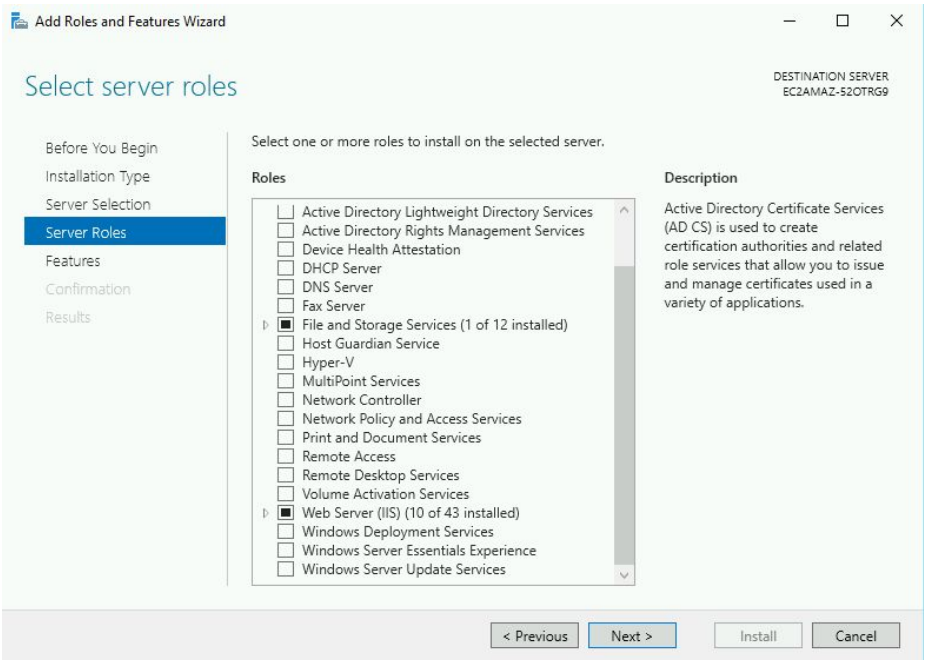


Fig 18: Select Web server (IIS), since i already installed it it shows 10 or 43 installed

h. Features: Select **.NET Framework 4.6 Features** and Click **Install**

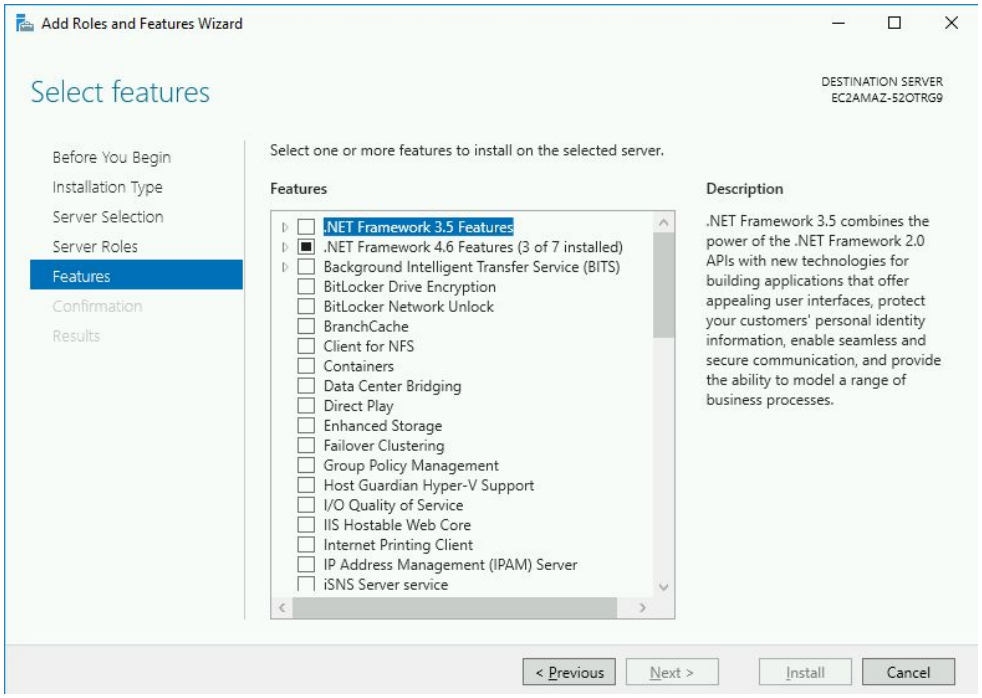


Fig 19: Select .NET Framework

i. At this point the server is going to install all what you have selected. If you have installed IIS correctly, copy and paste the Public IP of your server into a browser will show you the IIS welcome page.

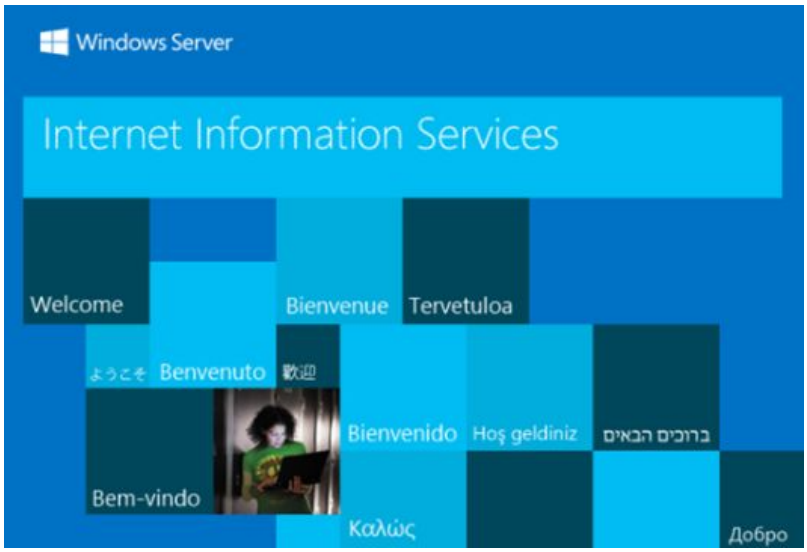


Fig 20: If all went right this should appear on your web browser

Creating and Displaying Hello World

---

1. Creating the file
    j. Go to **C:\inetpub\wwwroot** in your windows server
    k. Create a folder for your project (In this case “Hello_World”) in this directory
    l. Go into **C:\inetpub\wwwroot\Hello_World**
    m. Open a text editor such as Notepad or Atom (if installed)
    n. Create a basic hello world html file

```
<!DOCTYPE html>
<html>
   <head>
     <title>A Small hello</title>
   </head>

   <body>
     <H1>HI</H1>
     <p>This is a minimal "hello world" HTML document.</p>
   </body>
</html>
```

Fig 21: Sample hello world html file

    o. Save it (name it: **index.html**)
    p. So you should have **C:\inetpub\wwwroot\Hello_World\index.html**
2. Getting IIS to run your Hello World file
    a. Open **Server Manager** from the start menu
    b. On the top right, there is an option called **Tools,** click on it
    c. A drop down menu will appear, Click on **Internet Information Services (IIS) Manager**
    d. In IIS, on the left sidebar there should be a **Start Page** and a **Server**. You should be able to expand the server so you see its children (**Application Pools** and **Sites**)
    e. Right click on **Sites**
    f. Click **Add Website**

g. You should get this pop-up window

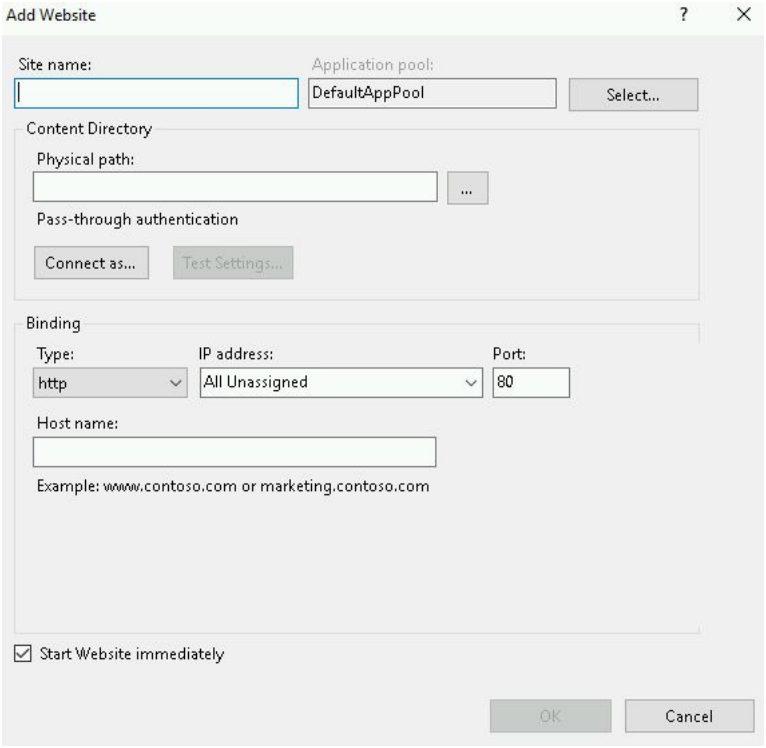


Fig 22: Input the following

h. For Site name: Give it whatever name you want (Hello_World is a good name)
i. In Content Directory: In physical path click on the … box and locate your hello_world folder in **C:\inetpub\wwwroot\Hello_World**
j. Click OK (Note: it will complain that a website is already using port 80, ignore this for now we need to disable the default IIS page)
k. Back in IIS, click on **Websites**. There should be a list that is displayed. One is the **Default website** (the IIS welcome page) and the other one is what you just created.
l. Right Click on the **Default Website -> Manage Website -> Stop**
m. Do the same to the Hello_World website, right click **Hello_World -> Manage Website -> Start**
n. Going back to the Public IP address of the website you should see the hello world on the web browser.

Fig 23: Assuming everything went alright you should see this

Route 53 - Connecting your Domain name to your EC2 Instance

Assumptions: You already bought a domain name from somewhere else.

---

1. When you login into AWS go to **Route 53** under **Network & Content Delivery** table
2. On your left you should see **Hosted Zones**, click on it
3. Assuming you bought a domain name from another provider such as Namecheap, Godaddy, or Domain.com click on **Create Hosted Zone**
4. On the right a panel should pop up, enter the **Domain Name** you bought (example.com) and leave **Type** as **Public Hosted Zone**, click **Create**

Fig 24: Create it

5. It should bring you into the Hosted Zone page

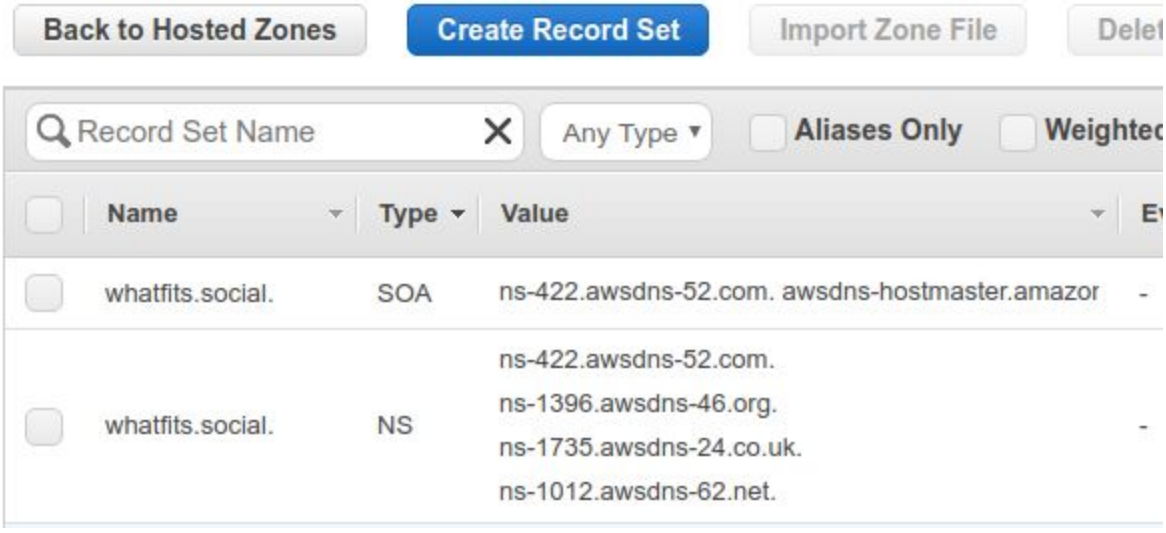


Fig 25: These are your Record Sets for the domain by default

6. Keep in mind the Values in the Name Server (In NS type), you are going to need to enter these 4 values in your own domain DNS manager on what ever service you bought it from

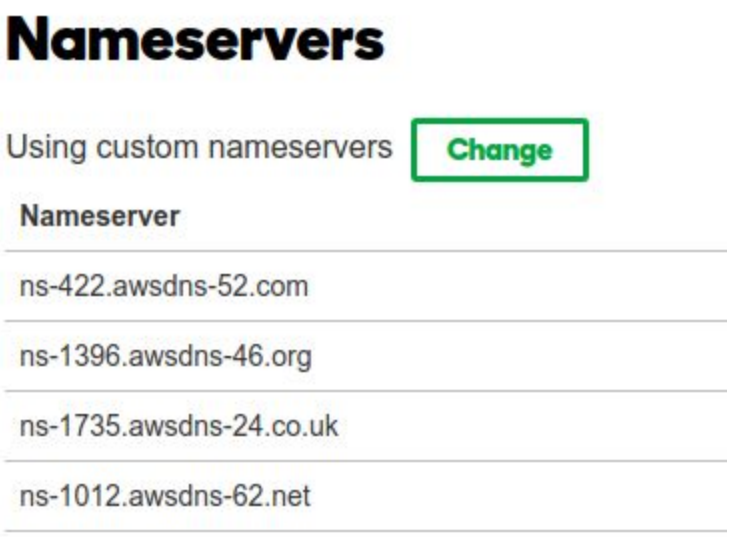


Fig 26: Change your Nameservers on your domain site (where you bought it) with the nameservers from AWS Route 53. This is from godaddy.

7. Back in AWS -> Route 53 -> Hosted Zones, Click on **Create Record Set**
8. Create another Record set, and enter the following
    a. Name: **www**
    b. Type: **A - IPv4 address**
    c. Alias: **No**
    d. TTL (Seconds): **[Leave default]**
    e. Value: **[Your server's public IP Address] XX.XXX.XXX.X**

f. Routing Policy: **Simple**

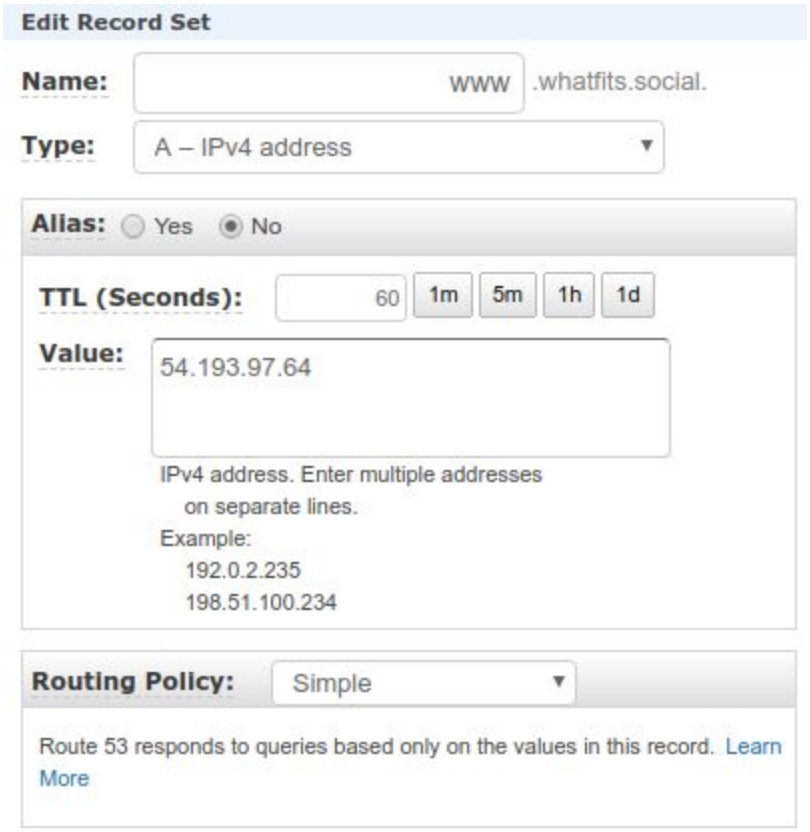


Fig 27: How the other one should look like

9. On the panel on your right, enter the following
    a. Name: **[Leave empty]**
    b. Type: **A - IPv4 address**
    c. Alias: **Yes**
    d. Alias Target: **www.[YourDomain].com** or whatever extension you have
    e. Routing Policy: **Simple**
    f. Evaluate Target Health: **No**

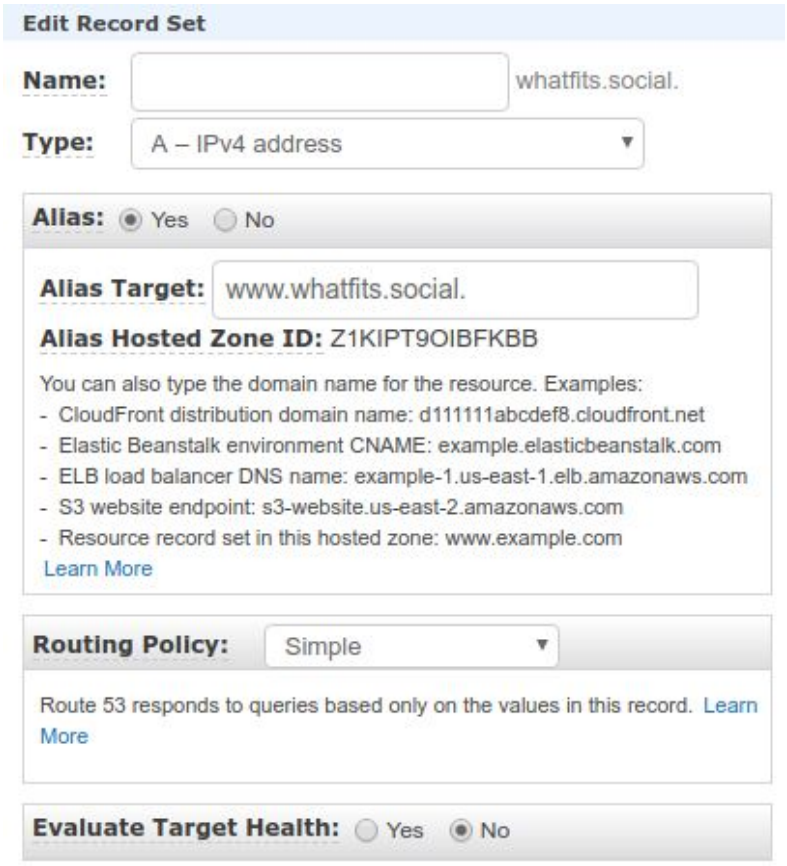


Fig 27: What is should look like

10. At this point your domain name should now point to your IP address of your server. The weird part is might take a while (around an hour or so) for it to work.
11. Entering your domain name on your web browser should now take you to your server's webpage.

Installing SSL Certificate on Windows Server

The following steps requires the use of letsencrypt-win-simple from Lone-Code to create the certificates. Which can be found here. The steps used can be found here as well. Also this phase **assumes you already have a domain name that routes to your site**.

---

1. Before you create the certificate, we need to add a binding to our website in IIS.
    a. Go to IIS
    b. Double click on your website on the left bar and on the right there should be a right bar with options
    c. Under **Actions** select **Bindings**

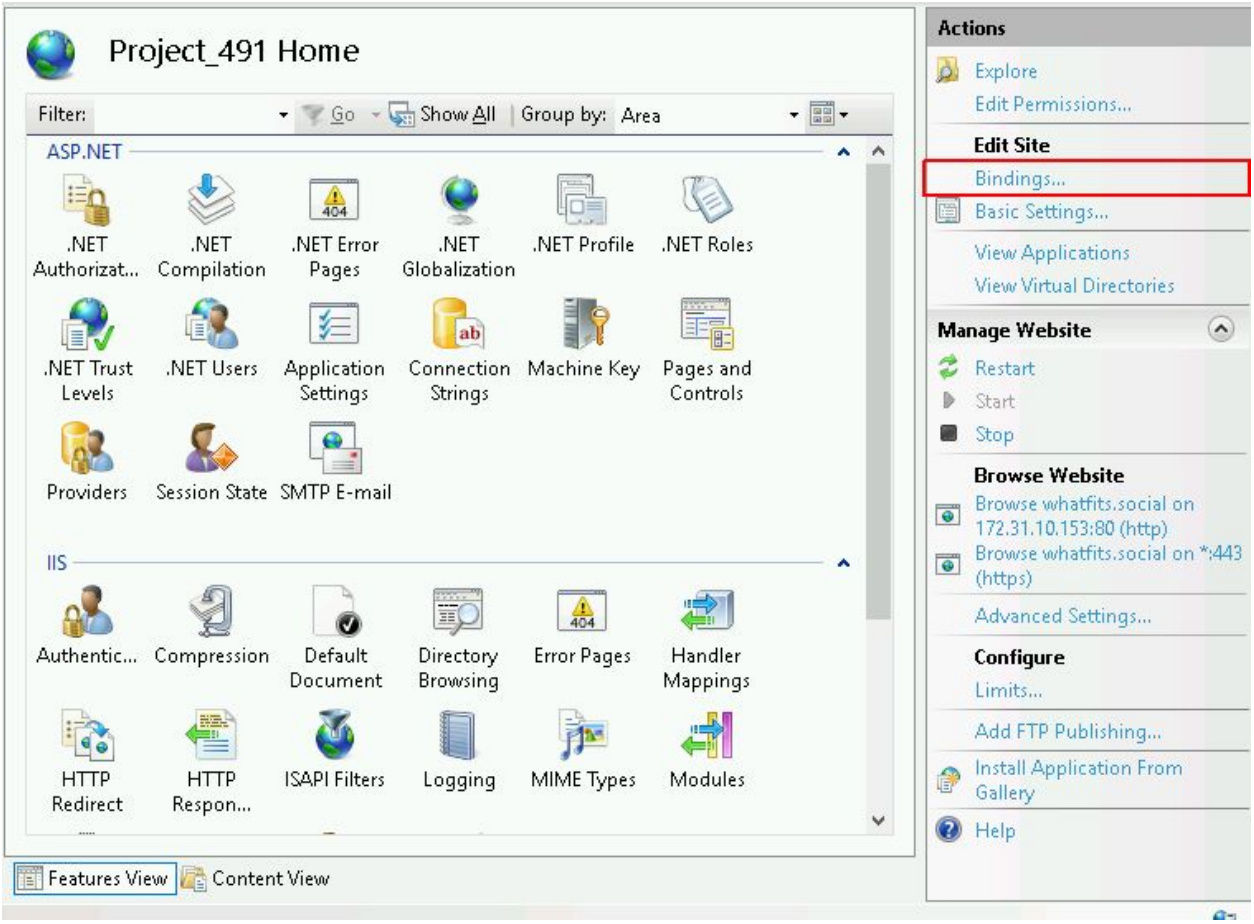


Figure 28: Select Binding on the right

    d. On Site Bindings click on **Add**
    e. Leave Type as **HTTP**
    f. Set **IP address** as your down(use drop down box)
    g. Leave **port 80** as is
    h. Enter your **Hostname** (ex. whatfits.social)
    i. OK

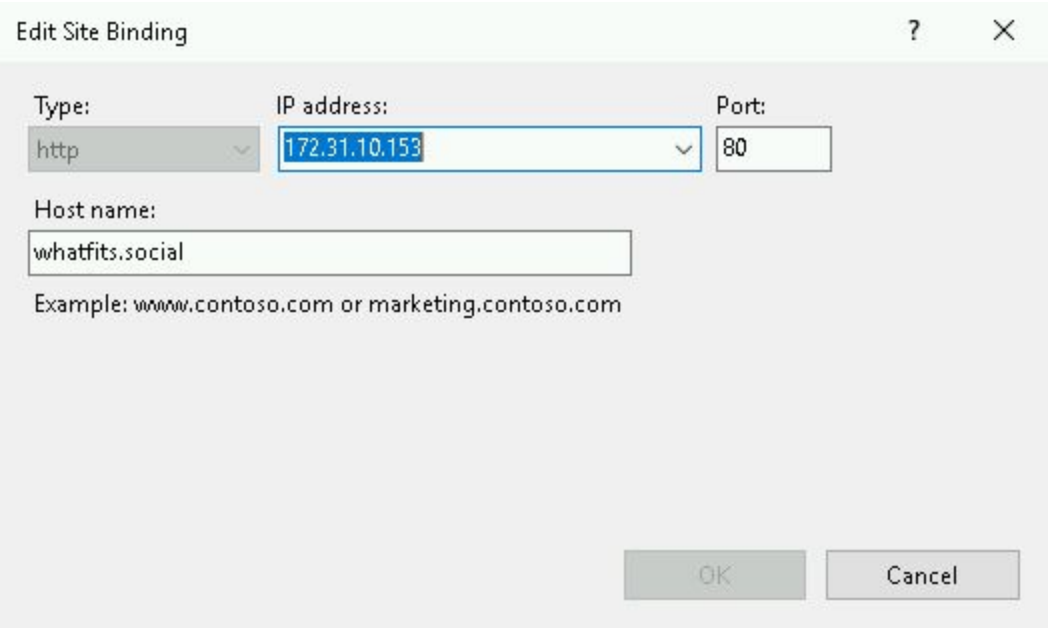


Figure 29: What you should have

2. Installing SSL Certificate on the website
    a. Go to https://github.com/Lone-Coder/letsencrypt-win-simple/releases
    b. Download the latest package: letsencrypt-win-simple.v1.9.7.2.zip
    c. Extract anywhere in your directory and navigate into the folder
    d. **Run** letsencrypt.exe
    e. It's going to ask **your email** and whether you agree with their **terms and conditions**
    f. Enter your email and agree to terms of agreement
    g. Then a menu should appear

```
N: Create new certificate
L: List scheduled renewals
R: Renew scheduled
S: Renew specific
A: Renew *all*
C: Cancel scheduled renewal
X: Cancel *all* scheduled renewals
Q: Quit

Please choose from the menu: n
```

    h. Select **Create new certificate**
    i. Select **1: Single binding of an IIS site**

```
1: Single binding of an IIS site
2: SAN certificate for all bindings of an IIS site
3: SAN certificate for all bindings of multiple IIS sites
```

```
 4: Manually input host names
 C: Cancel

 Which kind of certificate would you like to create?: 1
```

j. At this point you should see your website as an option, select it

```
 1: whatfits.social (SiteId 3) [@C:\inetpub\wwwroot\cuppajoe]
 C: Cancel

 Choose site: 1
```

k. Then it's going to ask you how you want to validate this certificate, select **Create temporary application in IIS (recommended)**

```
 1: [tls-sni-01] Use IIS as endpoint
 2: [dns-01] Azure DNS
 3: [dns-01] Run external program/script to create and update records
 4: [http-01] Create temporary application in IIS (recommended)
 5: [http-01] Save file on local (network) path
 6: [http-01] Self-host verification files (port 80 will be unavailable
during validation)

 How would you like to validate this certificate?: 4
```

l. It's going to prompt you about stuff, type y

```
 [INFO] Authorizing whatfits.social using http-01 validation (IIS)
 [INFO] Answer should now be browsable at
http://whatfits.social/.well-known/acme-challenge/yI9gQozPzAaMw-8dZwcpqWjoQ
c5LLfzQs4rUpBWtZCE
 [INFO] Authorization result: valid
 [INFO] Requesting certificate whatfits.social 2017/11/17 12:50:20 PM
 [INFO] Saving certificate to
C:\ProgramData\letsencrypt-win-simple\httpsacme-v01.api.letsencrypt.org
 [INFO] Installing certificate in the certificate store
 [INFO] Adding certificate whatfits.social 2017/11/17 12:50:20 PM to store
WebHosting
 [INFO] Installing SSL certificate in server software
 [INFO] Adding new https binding whatfits.social:443
 [INFO] Committing binding changes to IIS
 [INFO] IIS will serve the new certificates after the Application Pool
IdleTimeout has been reached.

 Do you want to replace the existing task? (y/n): - yes
```

m. When it prompts you if you want to specify the user the task will run as, select **no**

```
 [INFO] Deleting existing task letsencrypt-win-simple
httpsacme-v01.api.letsencrypt.org from Windows Task Scheduler.
 [INFO] Creating task letsencrypt-win-simple
httpsacme-v01.api.letsencrypt.org with Windows Task scheduler at 9am every
day.

 Do you want to specify the user the task will run as? (y/n): - no

 [INFO] Adding renewal for [IIS] whatfits.social (SiteId 3) [ @
C:\inetpub\wwwroot\cuppajoe]
 [INFO] Next renewal scheduled at 2018/1/16 8:51:19 PM
```

n. At this point if you manually enter https://whatfits.social you should see a green lock next to your domain name.

Figure 30: You have to manually enter https everytime else it will take you to the normal unsecure site

o. On IIS, double click on your website, on the right hand side click **Bindings…**

Figure 31: Double check Site Binding for https

p. A new binding should've been created that has type **https** and **port 443**

3. Redirecting HTTP to HTTPS
    a. Download url-rewrite module for IIS from the iis.net website here
    b. Install on Server
    c. Go to IIS and **double click on your website** you want to enable redirection (In this case Project_491)
    d. Click on **URL Rewrite**

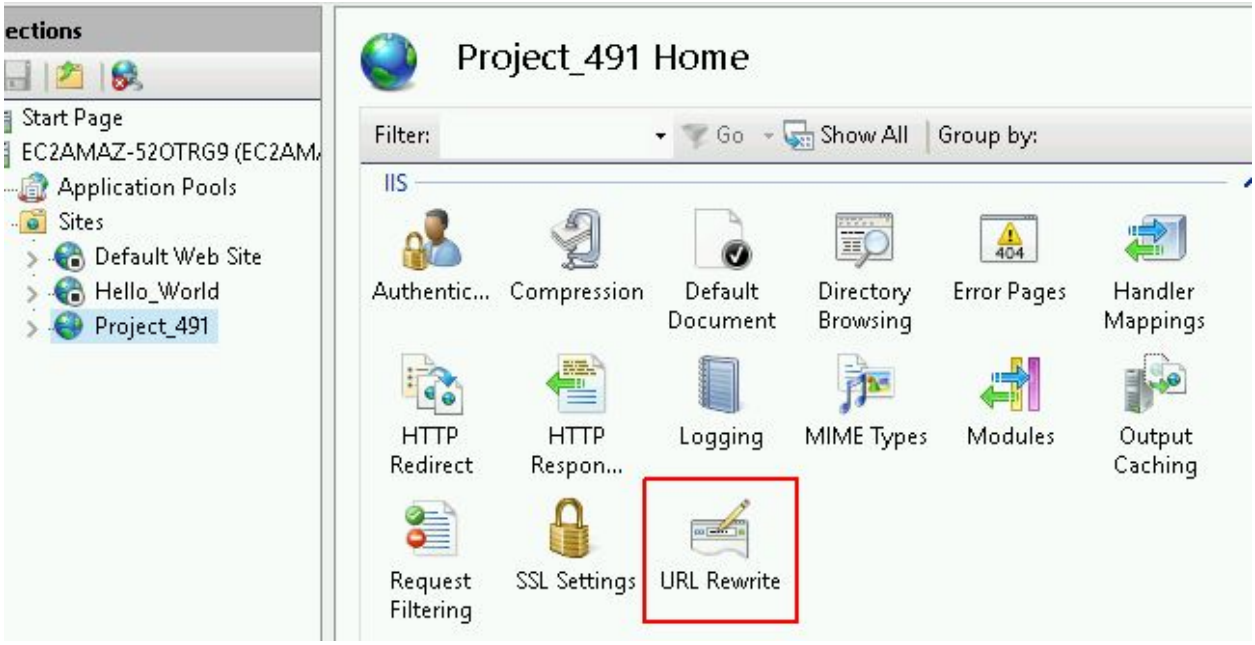


Figure 32: Options when you double click on Project_491

e. On the right select **Add Rule**

f. Select **Blank Rule** then **OK**

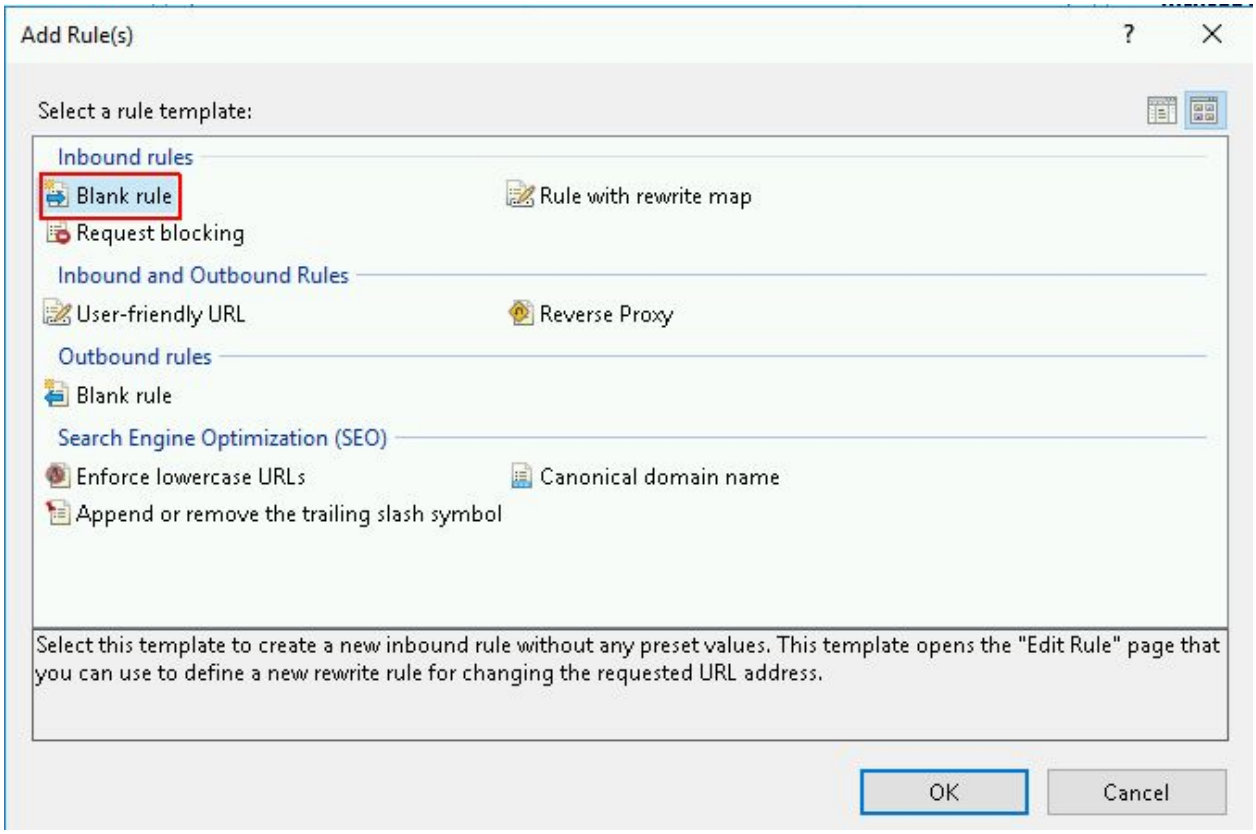


Figure 33: Add Rules Popup

g. For Edit Inbound Rules, for Name enter **https**

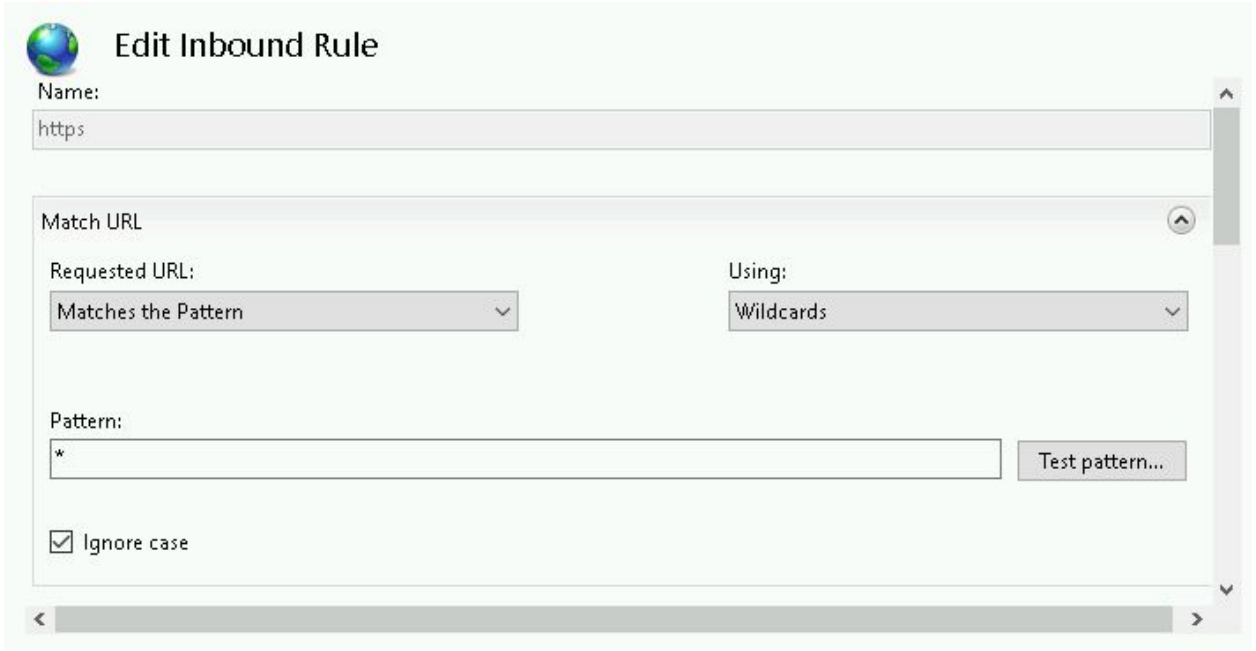


Figure 34: Inbound Rules for your webpage

h. Leave **Requested URL** as is, Change Using to **Wildcards**
i. Enter For **Pattern**: *
j. Scroll down and expand Conditions
k. Add Condition

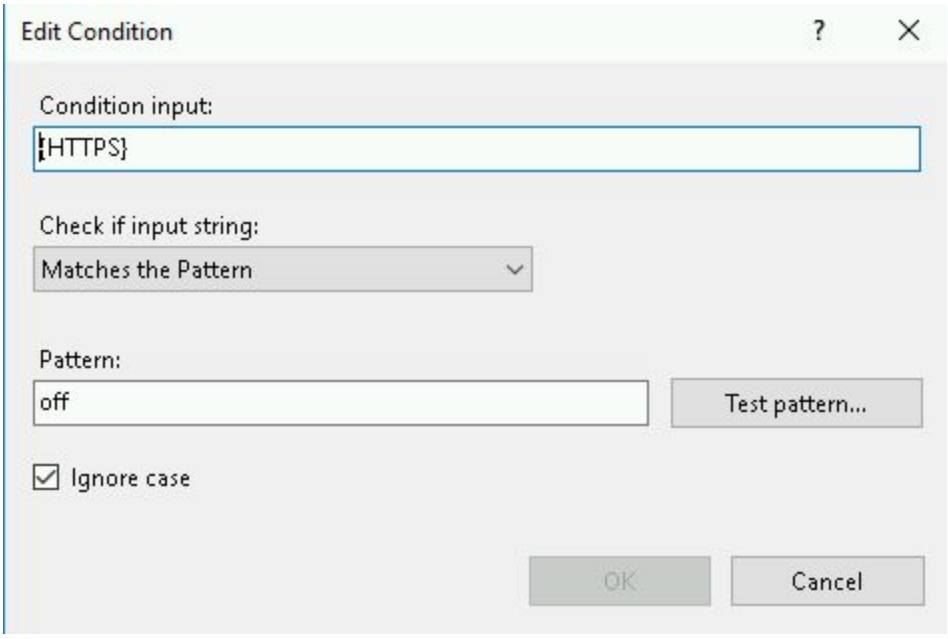


Figure 35: I already added a condition hence it says Edit Condition on the top

l. Enter for **Condition Input**: {HTTPS}
m. Enter for **Pattern**: off
n. Leave **Ignore Case** checked
o. Scroll down to **Action**

Figure 36: This is the last step you need

p. **Action Type**: Redirect
q. Enter for **Redirected URL**: https:{HTTP_POST}{REQUEST_URI}
r. Change **Redirect type** to: Found (302)
s. Click **Apply** on the right sidebar
t. Click **Back to Rules**

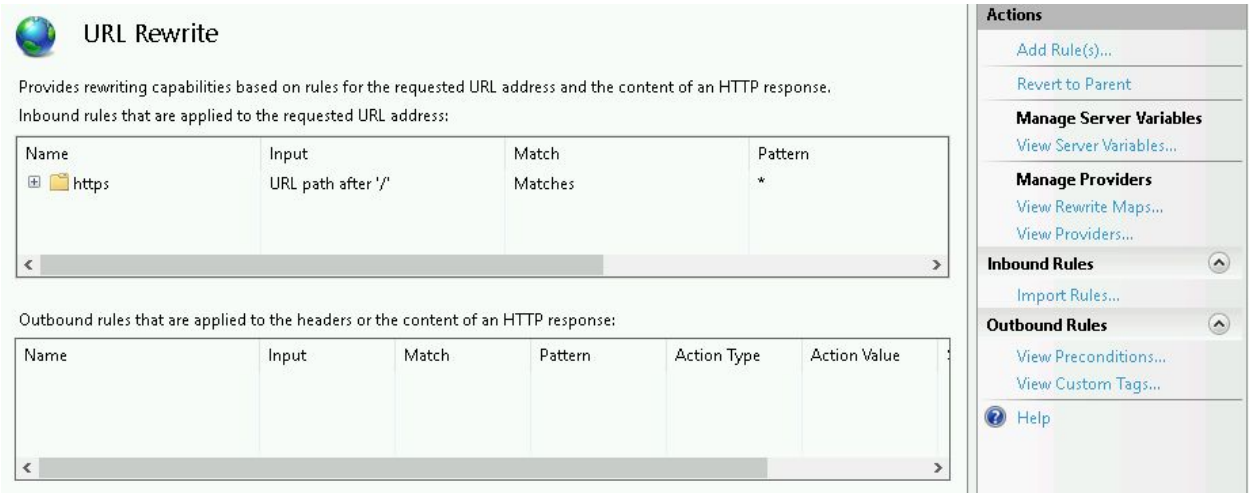


Figure 37: You should see your new rule to redirect any traffic to port 443

u. Select the rule you just created, on the right sidebar there is a disable rule
v. This step should create a file in your project folder called web.config, this file enables you to redirect to https (DON'T DELETE IT)

4. Improving SSL Grade to A+
   a. Go to Qualys SSL Labs
      i. https://www.ssllabs.com/index.html
   b. On your right click on **Test your server**

Figure 38: Select it

c. Enter your website domain and hit **Submit**

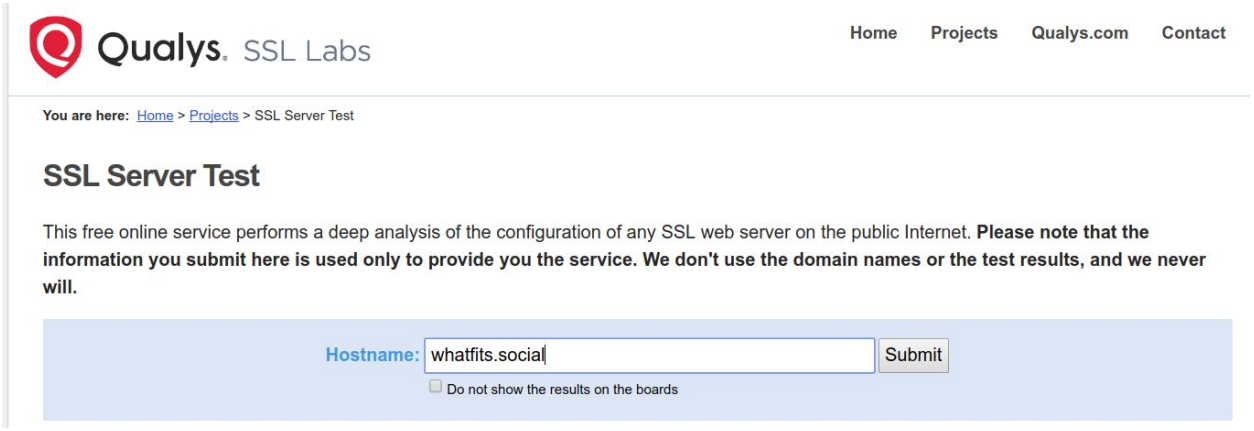


Figure 39: Enter it

d. If you have the SSL Certificate installed you should see a B rating

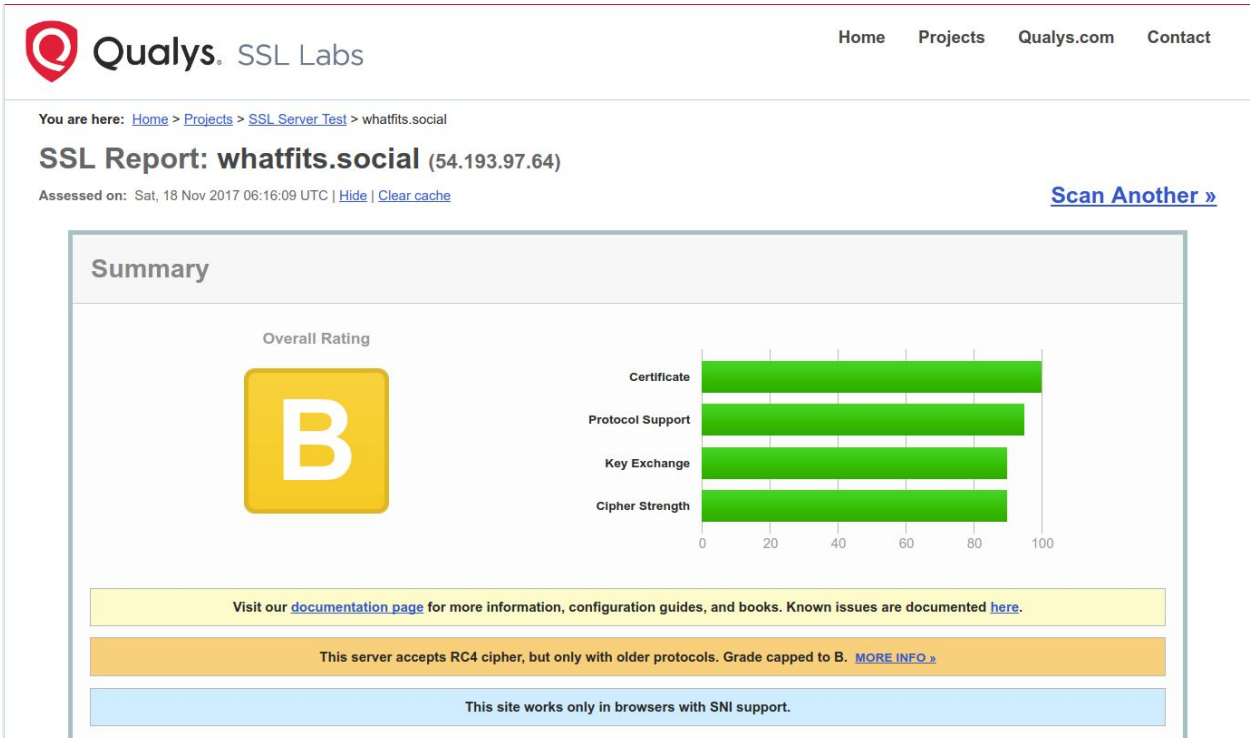


Figure 40: This page tells you what improvements you can make to your server

e. The result returns improvements that could be made on the server.
f. Disabling Ciphers/Cipher Suites/Hashes
    i. Download **IISCrypto** from https://www.nartac.com/Products/IISCrypto
    ii. Run the Application
    iii. Uncheck the following for **Protocols**
        1. Multi-Protocol Unified Hello
            a. PCT 1.0
            b. SSL 2.0
            c. SSL 3.0
        2. Uncheck the following for **Ciphers**

a. Everything but AES 256/256

3. Uncheck the following for **Hashes**
    a. MD5
    b. SHA
4. Leave **Key Exchanges** alone

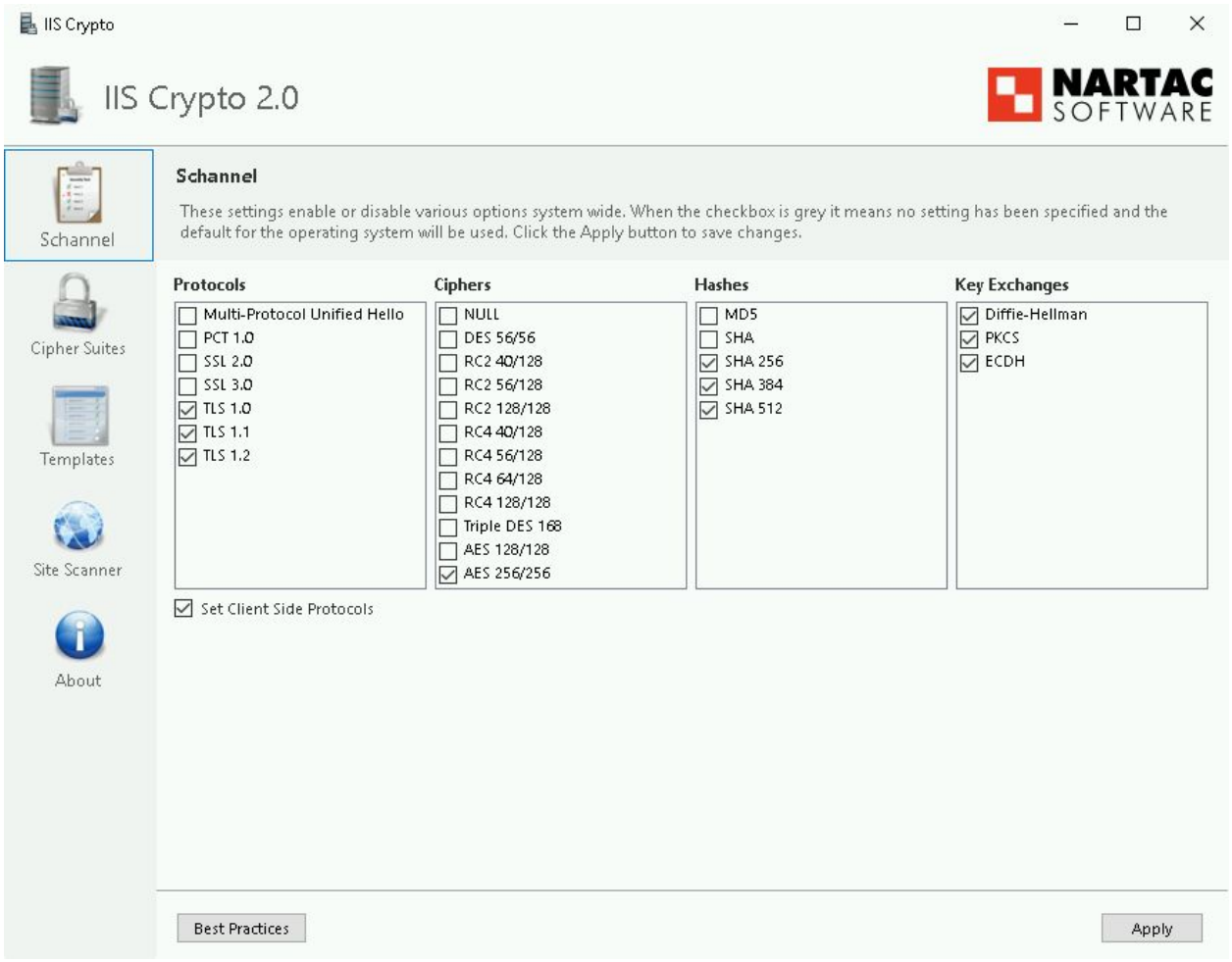


Figure 41: Yours should look like this

iv. Click on **Cipher Suites**

v. Uncheck everything but the Following:

1. TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
2. TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
3. TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
4. TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
5. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
6. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
7. TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
8. TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
9. TLS_RSA_WTH_AES_256_GCM_SHA484
10. TLS_RSA_WTH_AES_128_GCM_SHA256

11. TLS_RSA_WTH_AES_CBC_SHA256
12. TLS_RSA_WTH_AES_CBC_SHA256

Figure 42: Should look like this after

vi. Click **apply**, you will be prompted to restart your server. Restarting the server once that has happened will remove those ciphers/Cipher Suites/Hashes/etc from your system

vii. Go to your project folder and find the Web.config file that was created when you created the HTTP URL REDIRECT.

viii. Add the following code on there

```
<httpProtocol>
        <customHeaders>
            <add name = "Strict-Transport-Security"
value="max-age=63072000"/>
        </customHeaders>
</httpProtocol>
```

ix. So at this point your web.config file should look like this:

```
<?xml version="1.0" encoding="UTF-8"?>

<configuration>
    <system.webServer>
    <httpProtocol>
        <customHeaders>
            <add name = "Strict-Transport-Security" value="max-age=63072000"/>
        </customHeaders>
    </httpProtocol>
        <httpRedirect enabled="false" destination="https://whatfits.social"
exactDestination="false" childOnly="false" />
        <rewrite>
            <rules>
                <rule name="https" enabled="true" patternSyntax="Wildcard"
stopProcessing="true">
                    <match url="*" />
                    <conditions>
                        <add input="{HTTPS}" pattern="off" />
                    </conditions>
                    <action type="Redirect" url="https:{HTTP_HOST}{REQUEST_URI}"
redirectType="Found" />
                </rule>
            </rules>
        </rewrite>
    </system.webServer>
</configuration>
```

x. Save the file and restart the server so the changes take place

xi. Once the server has rebooted, rerun SSL Labs server test to see your score

xii. At this point you should have a A+ rating on your server

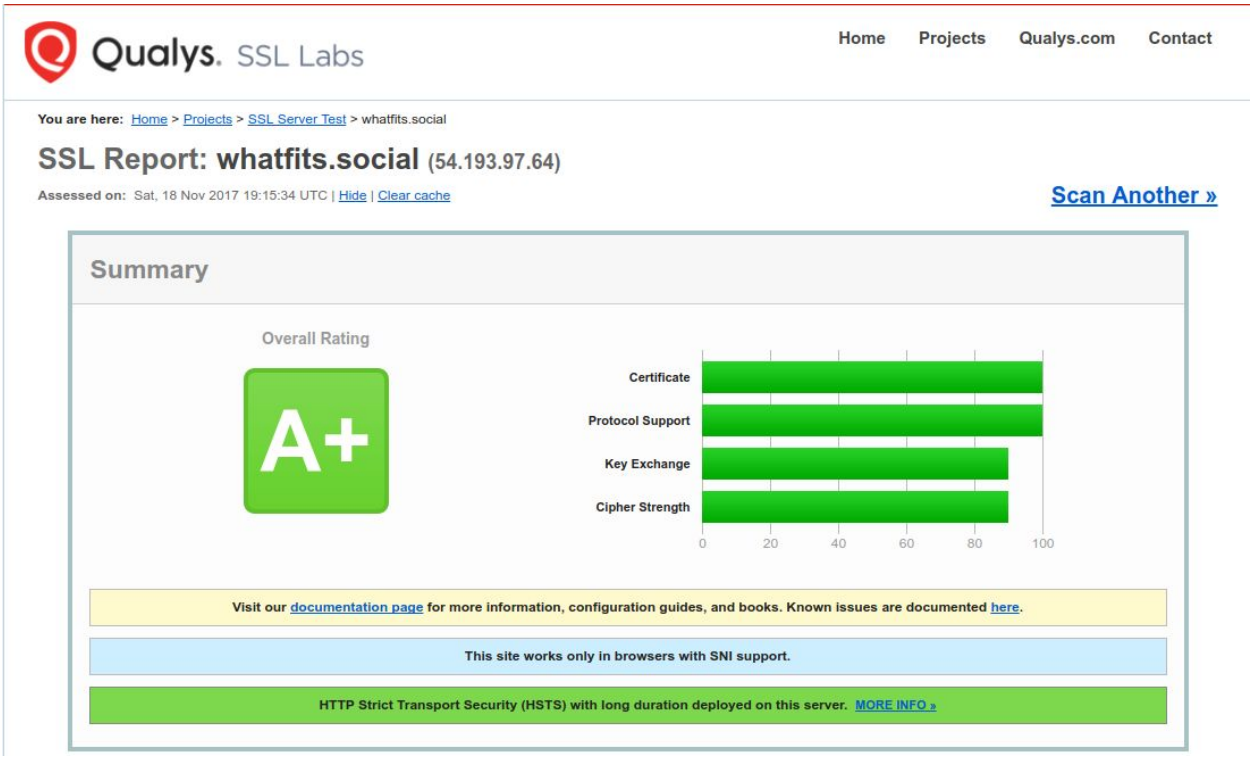


Figure 43: A screenshot of our SSL Report

Installing and Setting up ASP.NET 4.6

1. Open up Server Manager
2. Click on Add Roles and Features Wizard
3. Click Next until you reach Server Roles
4. On Server Roles -> Expand Web Server(IIS) -> Expand Web Server -> Expand Application Development

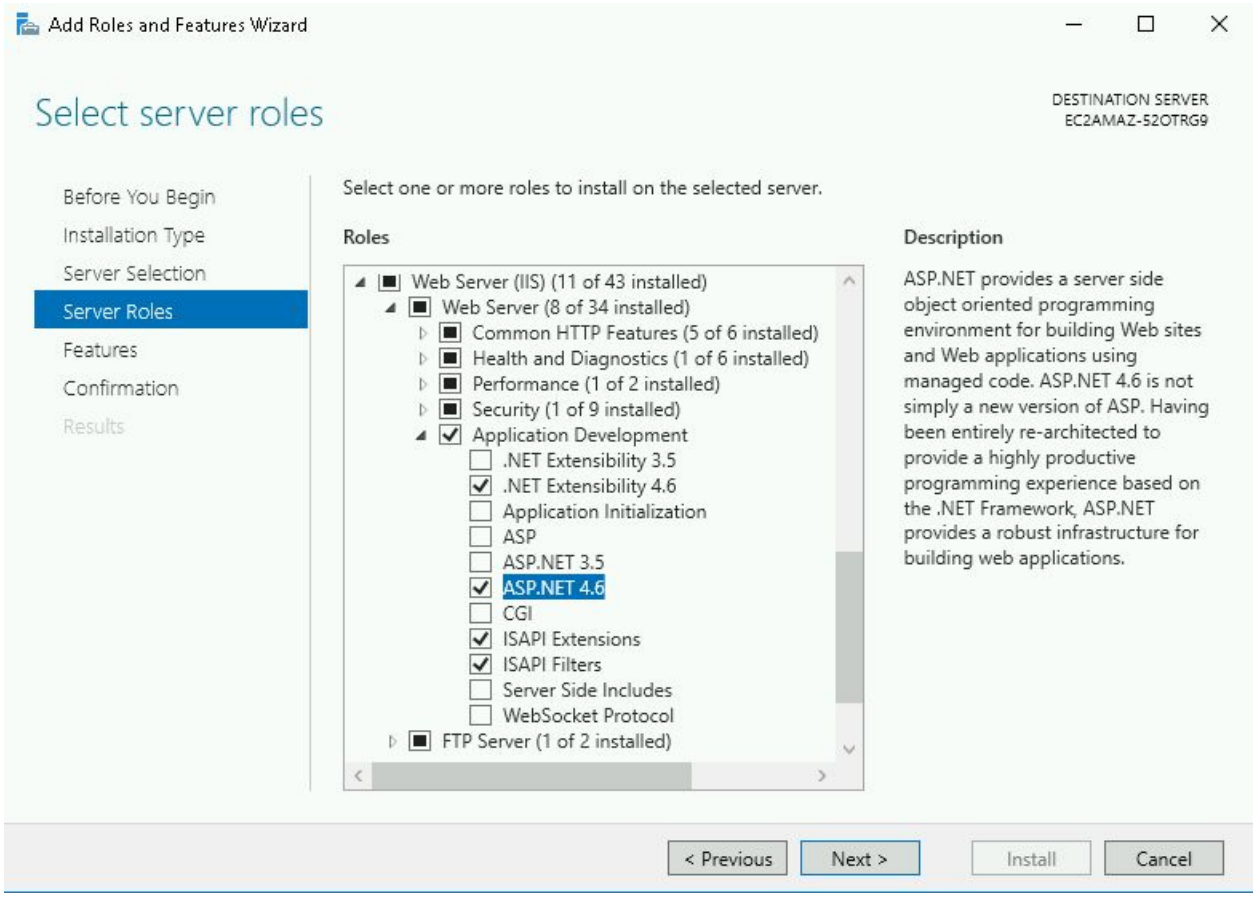


Figure X: Selecting what to install for server roles

5. Select ASP.NET 4.6 and .NET Extensibility 4.6
6. Optional (CONTAINERS)
7. Click on next until your reach the Confirmation page

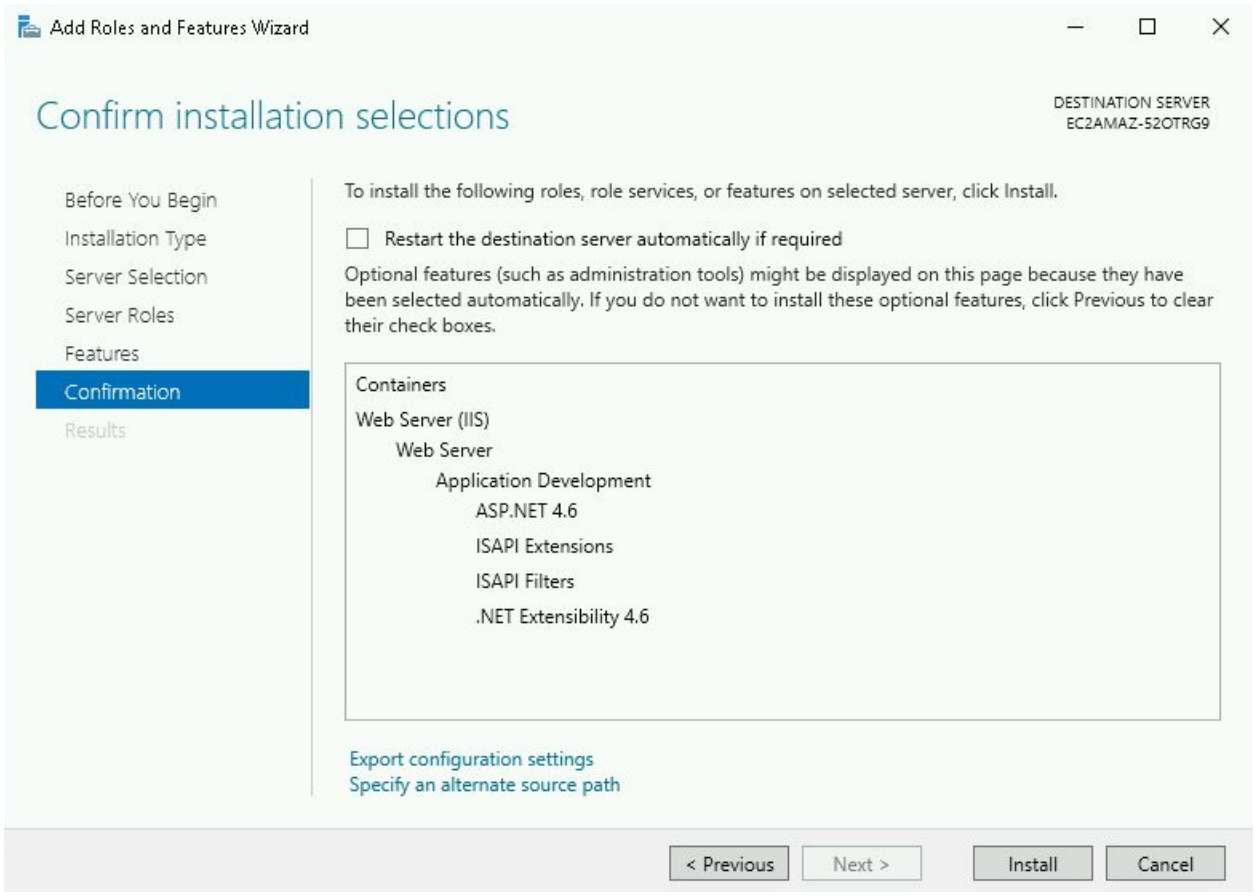


Figure:

8. Click Install
9. You will be prompted to restart your server. Do it.

Installing Misc. Software

Software that we installed

Misc. Software

1. Github Desktop Client **Version 1.0.6 (64-bit)**
    a. https://desktop.github.com
2. Mozilla Firefox **Version 56.0.2 (32-bit)**
    a. https://www.nartac.com/Products/IISCrypto/Download
3. Letsencrypt-win-simple **Version 1.9.7.2**
    a. https://www.nartac.com/Products/IISCrypto/Download
4. IIS Crypto GUI **Version 2.0**
    a. https://www.nartac.com/Products/IISCrypto/Download

Microsoft Software (1-4 can be downloaded through Windows Server Manager)

1. Windows Server 2016
2. Microsoft Internet Information Services 10 - Application Server Manager
3. .NET Framework **Version 4.6**
4. ASP.NET **Version 4.6**
5. Containers
6. VSCode **Version 1.18.0 (64-bit)**
7. IIS URL Rewrite **Version 2.1**
8. SQL Server 2017
9. SQL Server Management Studio (SSMS) **Version 14.7**