

How to deploy the SPIRAL website on an Azure ubuntu 22.04 server

August 3, 2023

1 How to move SPIRAL to a new Azure Ubuntu 22.04 server?

A helpful tutorial: <https://www.rosehosting.com/blog/how-to-install-flask-on-ubuntu-22-04-with-apache-and-wsgi/>.

Note that we store the app at `/var/www/html/`.

1.1 Create a VM and open the 80 (http) and 443 (https) ports (and 5000)

This tutorial may be helpful for this step: https://www.youtube.com/watch?v=HmKUCPRKNos&ab_channel=V

In the Azure portal:

- Follow the tutorial to create a VM. The current deployment uses a Standard E20as v5 (20 vcpus, 160 GiB memory).
- In the Azure portal: VM -> Networking -> Add inbound security rule -> port 80, name it “allowHTTP”
- In the Azure portal: VM -> Networking -> Add inbound security rule -> port 443, name it “allowHTTPS”
- In the Azure portal: VM -> Networking -> Add inbound security rule -> port 5000, name it “allow5000” (only for now)
- In the Azure portal: VM -> Networking -> Make sure that SSH is open for everywhere (only for now)

Important: before beginning the installation, make sure that all the disks are allocated and mounted. The current deployment uses a 1T disk that is used for both the OS and data. However, the more “correct” way is to use a small (32G) OS disk, and a larger (let’s say 1T) Data disk. In this case, you should mount the Data disk on some folder, let’s say `/var2`, and then create a symbolic link from `/var` to `/var2`.

Some useful commands:

- `fdisk -l` shows both the physical and attached disks.
- `lsblk` shows the physical disks and whether they are partitioned and mounted. So if there is “sdb” for example but it is not divided to “sdb1”, “sdb14” etc., then it is not partitioned. If it is divided, but for example “sdb1” has nothing under “MOUNTPOINTS”, then it is not mounted.

- `df -h` shows the actual attached disks (and the available space). So make sure that the “SIZE” fits what you expect.

1.2 Apache2

Open Putty and connect to the VM (public IP) with you name and password.

As in the tutorial:

Before installing the software, we need to update the system packages to the latest versions available.

```
sudo apt-get update -y && sudo apt-get upgrade -y
```

To install the Apache Web server execute the following command:

```
sudo apt install apache2 -y
```

Once installed, start and enable the service.

```
sudo systemctl enable apache2 && sudo systemctl start apache2
```

Check if the service is up and running:

```
sudo systemctl status apache2
```

Open the server IP in a browser. You should see the apache2 ubuntu default page.

1.3 Python

Verify Python installation:

```
python3 -V
```

you should see Python 3.10.x

Install python3-pip, python3-venv, python3-dev:

```
sudo apt-get install python3-pip python3-venv python3-dev
```

1.4 Clone SPIRAL repository, install packages and add needed files

```
cd /var/www/html/
```

As explained in the readme of <https://github.com/hadasbi/SPIRAL.web.tool> (steps 2-7):

```
sudo git clone https://github.com/hadasbi/SPIRAL.web.tool.git
```

```
cd SPIRAL.web.tool
```

```
sudo python3 -m venv spiral_venv
```

Activate the virtual environment:

```
source ./spiral_venv/bin/activate
```

Install required Python packages:

```
sudo ./spiral_venv/bin/pip install -r ./requirements.txt
```

Create analysis directory:

```
cd static
sudo mkdir analysis
cd ..
```

Move required files (that are not in Github for obvious reasons...) to the server: The files are Flask-WTF_encryption_key.txt and mail_password.txt.

In winSCP, move the files from your computer to /home/SpiralAdmin (since you can only write to /var/www/html in sudo). Then move them to their final location by:

```
sudo mv /home/SpiralAdmin/mail_password.txt ./mail_password.txt
sudo mv /home/SpiralAdmin/Flask-WTF_encryption_key.txt ./Flask-WTF_encryption_key.txt
```

Add the server name to list of servers running in production mode (in main.py) In main.py, look for the line:

```
if hostname in ['bi-fiona', 'Spin']: # production (linux)
```

And add <SERVERNAME>:

```
if hostname in ['bi-fiona', 'Spin', '<SERVERNAME>']: # production (linux)
```

Check that the flask app is working: A very important note: app.config['SERVER_NAME'] in \var\www\html\SPIRAL.web.tool\main.py and ServerName in the configuration file /etc/apache2/sites-available/spiral.technion.ac.il.conf (see next section) have to match.

For now, in order to test the app, change app.config['SERVER_NAME'] in \var\www\html\SPIRAL.web.tool\main.py to <VM_PUBLIC_IP>:5000 with the command:

```
sudo nano \var\www\html\SPIRAL.web.tool\main.py
```

Navigate the file and make the change.

To save, type: ctrl+X, then Y, then enter.

Then you can test the app:

Test 1: make sure you have the venv activated and type:

```
python main.py
```

Test 2: make sure you have the venv activated and type:

```
export FLASK_APP=main.py
flask run --host=0.0.0.0
```

For both tests you should see:

Running on all addresses.

WARNING: This is a development server. Do not use it in a production deployment.

Running on http://10.0.0.8:5000/ (Press CTRL+C to quit)

Open the browser and type: <VM_PUBLIC_IP>:5000

You should see SPIRAL's homepage.

Go back to the terminal, press ctrl+C and:

deactivate

1.5 What if it doesn't work?

Go to the last line in main.py and add debug=True to app.run().

Then:

```
python file.py runserver -d
```

2 Pre-configuration step just for Windows

Skip this step. It is here in case SPIRAL would have to work from Windows one day... For Windows, you have to move activate_this.py file to the right location:

```
sudo cp activate_this.py ./spiral_venv/bin/activate_this.py
```

3 WSGI configuration for HTTP

Install mod_wsgi:

```
sudo apt-get install libapache2-mod-wsgi-py3
```

Create SPIRAL configuration file:

```
sudo touch /etc/apache2/sites-available/spiral.technion.ac.il.conf
```

To edit the file, open it with nano:

```
sudo nano /etc/apache2/sites-available/spiral.technion.ac.il.conf
```

And paste this:

```
<VirtualHost *:80>
#ServerName spiral.technion.ac.il
ServerName <VM_PUBLIC_IP>
ServerAlias www.spiral.technion.ac.il
DocumentRoot "/var/www/html/SPIRAL.web.tool/"
```

```
WSGIDaemonProcess spiral user=www-data group=www-data threads=20 python-home="/var/www/html/SP
#python-path=/var/www/html/SPIRAL.web.tool:/var/www/html/SPIRAL.web.tool/spiral_venv/lib/pyth
WSGIScriptAlias / "/var/www/html/SPIRAL.web.tool/app.py"
```

```
ErrorLog /var/log/apache2/spiral.technion.ac.il-error.log
```

```
CustomLog /var/log/apache2/spiral.technion.ac.il-access.log combined
```

```
<Directory "/var/www/html/SPIRAL.web.tool">
WSGIProcessGroup spiral
WSGIApplicationGroup %{GLOBAL}
Order deny,allow
Require all granted
</Directory>
LogLevel warn
</VirtualHost>
```

To save, type: ctrl+X, then Y, then enter.

Then, enable the Apache2 configuration file:

```
sudo a2ensite spiral.technion.ac.il.conf
```

Check the syntax of the Apache2 configuration:

```
sudo apachectl -t
```

You should see:

Syntax OK

Restart Apache:

```
sudo systemctl restart apache2
```

Remove index.html (the default Apache page) from /var/www/html/:

```
sudo mv /var/www/html/index.html /home/SpiralAdmin/index.html
```

Change the ownership of the relevant folder to the user that is supposed to run the app:

```
sudo chown -R www-data:www-data /var/www/
```

Change server name: Change both `app.config['SERVER_NAME']` in `\var\www\html\SPIRAL.web.tool\main.py` and `ServerName` in the configuration file `/etc/apache2/sites-available/spiral.technion.ac.il.conf` to `<VM_PUBLIC_IP>`.

Check if it's working: Restart Apache:

```
sudo systemctl restart apache2
```

Open `http://<VM_PUBLIC_IP>/` in a browser. SPIRAL's homepage should appear.

4 WSGI configuration for HTTPS

Follow this tutorial: <https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-22-04>,

but change the conf file `/etc/apache2/sites-available/spiral.technion.ac.il.conf` to:

```

<VirtualHost *:80>
ServerName <VM_PUBLIC_IP>
ServerAlias www.spiral.technion.ac.il
Redirect / https://<VM_PUBLIC_IP>/
</VirtualHost>

<VirtualHost *:443>
#ServerName spiral.technion.ac.il
ServerName <VM_PUBLIC_IP>
ServerAlias www.spiral.technion.ac.il
DocumentRoot "/var/www/html/SPIRAL.web.tool/"

WSGIDaemonProcess spiral user=www-data group=www-data processes=19 maximum-requests=100 python-
#python-path=/var/www/html/SPIRAL.web.tool/:/var/www/html/SPIRAL.web.tool/spiral_venv/lib/python
WSGIScriptAlias / "/var/www/html/SPIRAL.web.tool/app.py"

ErrorLog /var/log/apache2/spiral.technion.ac.il-error.log
CustomLog /var/log/apache2/spiral.technion.ac.il-access.log combined

<Directory "/var/www/html/SPIRAL.web.tool">

WSGIProcessGroup spiral
WSGIApplicationGroup %{GLOBAL}
Order deny,allow
Require all granted
</Directory>
LogLevel warn

SSLEngine on
SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
</VirtualHost>

```

At this point the website should be working when you type in your browser either http://<VM_PUBLIC_IP> or https://<VM_PUBLIC_IP>. If this is not the case, try to debug (take a look at “Common issues” at the end of this document). Do not proceed until it is working.

5 Ask to set the DNS record

Send an email to dns@technion.ac.il and ask to point www.spiral.technion.ac.il and spiral.technion.ac.il to the IP of the new server (<VM_PUBLIC_IP>).

6 Generate a Technion SSL certificate

To issue an SSL certificate, you first have to generate an CSR file. Use this command:

```
openssl req -new -newkey rsa:2048 -nodes -keyout spiral.technion.ac.il.key -out spiral.technion.ac.il.csr
```

(This tutorial is helpful: <https://phoenixnap.com/kb/generate-openssl-certificate-signing-request>)

- Country Name – use a 2-letter country code (US for the United States): **IL**
- State – the state in which the domain owner is incorporated: **Israel** (not important)
- Locality – the city in which the domain owner is incorporated: **Haifa** (not important)
- Organization name – the legal entity that owns the domain: **Technion** (not important)
- Organizational unit name – the name of the department or group in your organization that deals with certificates: **CIS** (not important)
- Common name – typically the fully qualified domain name (FQDN), i.e. what the users type in a web browser to navigate to your website: ***.spiral.technion.ac.il** (SUPER important)
- Email address – the webmaster’s email address: **hadas.moriah@gmail.com**
- Challenge password: **.**

This process creates two files:

- a private key: **spiral.technion.ac.il.key**
- a csr file: **spiral.technion.ac.il.csr**.

You have to send the csr file to **CyberSecurity@technion.ac.il** (or- **ciso@technion.ac.il**) and ask them to issue an SSL certificate for you. Another option is to ask Dan Bliberg (**ardan@technion.ac.il**) to issue a certificate since he has access to the SECTIGO certificate system (<https://cis.technion.ac.il/sectigo/>). They don’t give access to students. The certificate you need is the “Certificate only, PEM encoded”.

7 When you have the Technion certificate and the DNS has been set:

- Copy the private key to **/etc/ssl/private/spiral.technion.ac.il.key**
- Copy the certificate that was issued for you (a ‘cer’ or ‘crt’ file) to **/etc/ssl/certs/_spiral_technion_ac_il_cert.cer**
- Change in **/var/www/html/SPIRAL.web.tool/main.py** the configuration of **ServerName** to **spiral.technion.ac.il**:

```
sudo nano /var/www/html/SPIRAL.web.tool/main.py
```

and then change so that:

```
app.config['SERVER_NAME'] = 'spiral.technion.ac.il'
```

- Make some changes in the SPIRAL configuration file:

```
sudo nano /etc/apache2/sites-available/spiral.technion.ac.il.conf
```

and then change the values of **ServerName**, **ServerAlias**, **Redirect**, **SSLCertificateFile**, **SSLCertificateKeyFile** in both port 80 and port 443:

```
ServerName spiral.technion.ac.il
```

```
ServerAlias www.spiral.technion.ac.il
```

```
Redirect / https://spiral.technion.ac.il/
```

```
SSLCertificateFile /etc/ssl/certs/__spiral_technion_ac_il_cert.cer
```

```
SSLCertificateKeyFile /etc/ssl/private/spiral.technion.ac.il.key
```

That should be enough... But since ServerAlias in port 443 did not work for me (that is, https://www.spiral.technion.ac.il did not find the website), I resorted to multiplying the port 443 section, ultimately creating the following final SPIRAL configuration file:

```
<VirtualHost *:80>
```

```
ServerName spiral.technion.ac.il
```

```
ServerAlias *.spiral.technion.ac.il
```

```
Redirect / https://spiral.technion.ac.il/
```

```
</VirtualHost>
```

```
<VirtualHost *:443>
```

```
ServerName spiral.technion.ac.il
```

```
#ServerAlias www.spiral.technion.ac.il
```

```
DocumentRoot "/var/www/html/SPIRAL.web.tool/"
```

```
WSGIDaemonProcess spiral user=www-data group=www-data processes=19 maximum-requests=100 python
```

```
WSGIScriptAlias / "/var/www/html/SPIRAL.web.tool/app.py"
```

```
ErrorLog /var/log/apache2/spiral.technion.ac.il-error.log
```

```
CustomLog /var/log/apache2/spiral.technion.ac.il-access.log combined
```

```
<Directory "/var/www/html/SPIRAL.web.tool">
```

```
WSGIProcessGroup spiral
```

```
WSGIApplicationGroup %{GLOBAL}
```

```
Order deny,allow
```

```
Require all granted
```

```
</Directory>
```

```
LogLevel warn
```

```
SSLEngine on
```

```
SSLCertificateFile /etc/ssl/certs/__spiral_technion_ac_il_cert.cer
```

```
SSLCertificateKeyFile /etc/ssl/private/wildcard.spiral.technion.ac.il.key
```

```
</VirtualHost>
```

```
<VirtualHost *:443>
```

```
ServerName www.spiral.technion.ac.il
```

```
Redirect / https://spiral.technion.ac.il/
```

```
</VirtualHost>
```


- Sometimes you need to add the row

```
ServerName 127.0.0.1
```

to the end of the Apache2 configuration file `/etc/apache2/apache2.conf`.

- Restart Apache2:

```
sudo systemctl restart apache2
```

8 Some last tasks...

- In the Azure portal: remove the port 5000 rule.
- In the Azure portal: Make sure that SSH is only open for a list of IPs (enter your IP). Remember this... next time you want to connect via SSH you'll need to first go to the Azure portal and enter your IP in the list of allowed IPs.
- And you're done! Hallelujah!

9 Common issues...

9.0.1 What to do if something isn't working?

Check the error logs (both SPIRAL's and Apache's):

```
cat /var/log/apache2/spiral.technion.ac.il-error.log
```

```
cat /var/log/apache2/error.log
```

Azure portal: make sure that your NIC and your subnet are under the same NSG.

Don't install nginx, as it can create issues by making the ports unavailable for Apache2.

If nginx is installed you have to completely remove it (see <https://www.cyberciti.biz/faq/remove-uninstall-nginx-from-ubuntu-debian-linux-apt-get-command/>).

[]:

```
[9]: # Check response time
import requests, time

for url in ['http://www.spiral.technion.ac.il', 'http://spiral.technion.ac.il',
↳ 'http://4.227.224.198/',
           'https://www.spiral.technion.ac.il', 'https://spiral.technion.ac.
↳ il', 'https://4.227.224.198/']:
    t0 = time.time()
    response = requests.get(url, verify=False)
    t1 = time.time()
    total = t1-t0
    print(url)
    print("Simple get request took " , total)
```

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified HTTPS request is being made to host 'spiral.technion.ac.il'. Adding certificate verification is strongly advised. See:
<https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings>
InsecureRequestWarning,

<http://www.spiral.technion.ac.il>
Simple get request took 0.7313344478607178

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified HTTPS request is being made to host 'spiral.technion.ac.il'. Adding certificate verification is strongly advised. See:
<https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings>
InsecureRequestWarning,

<http://spiral.technion.ac.il>
Simple get request took 0.7394886016845703
<http://4.227.224.198/>
Simple get request took 0.2856614589691162

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified HTTPS request is being made to host 'www.spiral.technion.ac.il'. Adding certificate verification is strongly advised. See:
<https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings>
InsecureRequestWarning,

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified HTTPS request is being made to host 'spiral.technion.ac.il'. Adding certificate verification is strongly advised. See:
<https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings>
InsecureRequestWarning,

<https://www.spiral.technion.ac.il>
Simple get request took 0.9088315963745117

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified HTTPS request is being made to host 'spiral.technion.ac.il'. Adding certificate verification is strongly advised. See:
<https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings>
InsecureRequestWarning,

<https://spiral.technion.ac.il>
Simple get request took 0.44493865966796875
<https://4.227.224.198/>
Simple get request took 0.44376707077026367

c:\program files (x86)\microsoft visual studio\shared\python37_64\lib\site-packages\urllib3\connectionpool.py:1020: InsecureRequestWarning: Unverified

```
HTTPS request is being made to host '4.227.224.198'. Adding certificate
verification is strongly advised. See:
https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings
InsecureRequestWarning,
```

[]: