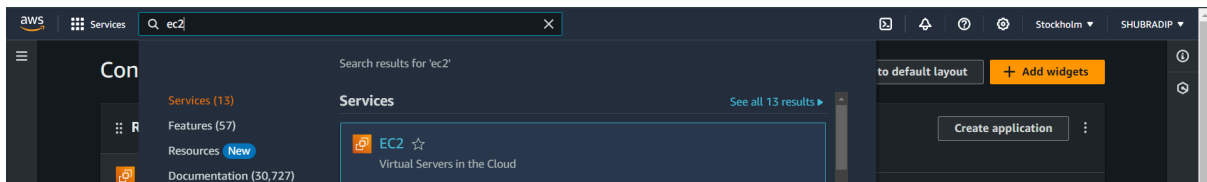


Assignment No:12

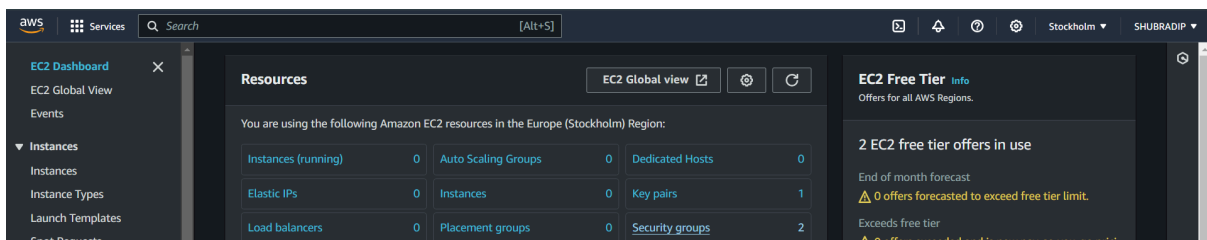
Problem Statement: Deploy and run the project in AWS without using the port.

The steps are as follows: -

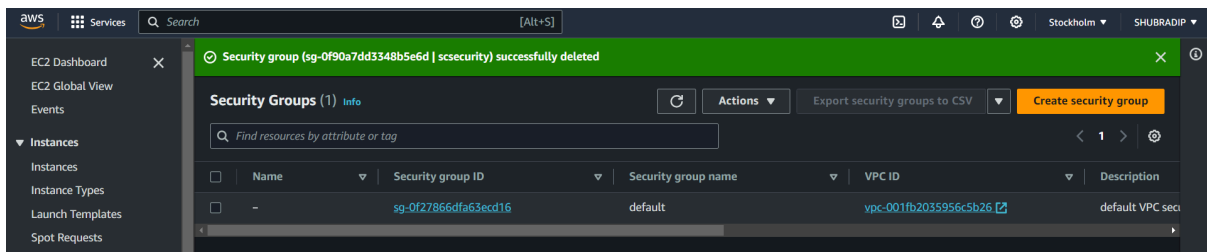
1. Navigate to AWS and locate the EC2 service. Choose the first option listed under the EC2 services.



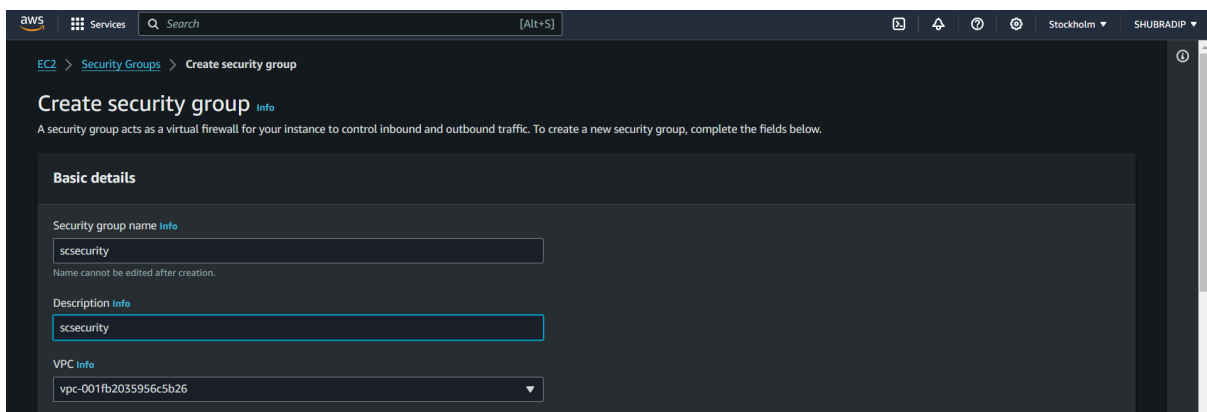
2. Select the "Security group" option.



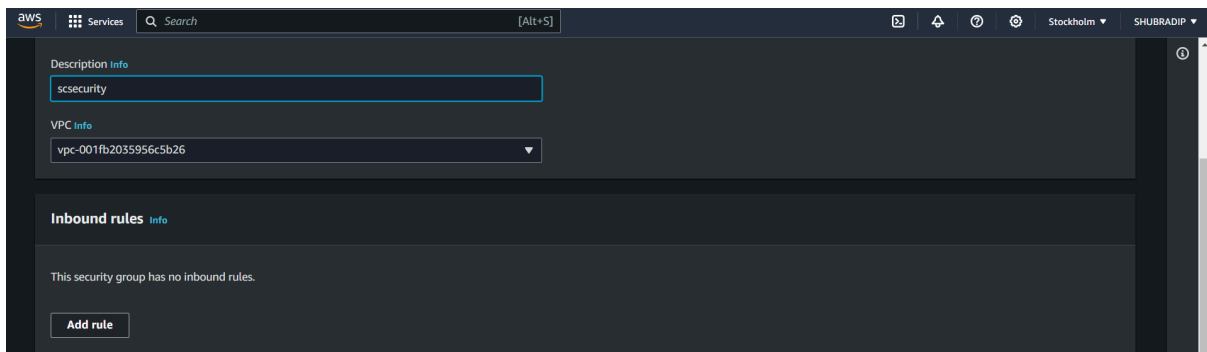
3. Now, select "create security group".



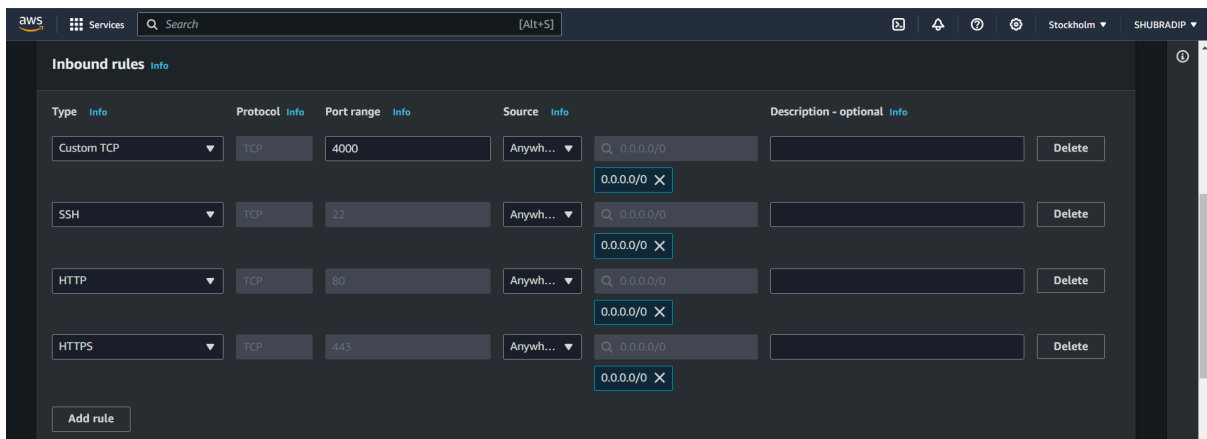
4. Choose an appropriate and valid name for the group (for instance, we've used "scsecurity" here). You can also copy the same name into the 'Description' field, or provide any other relevant information.



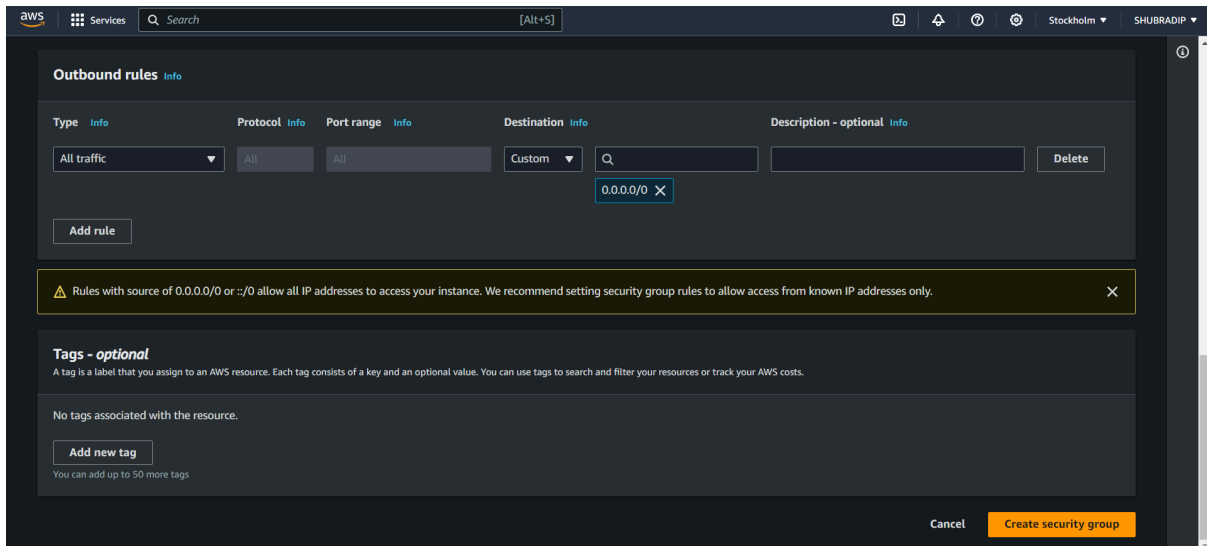
5. Within the '**Inbound rules**' section, select '**Add rule**'.



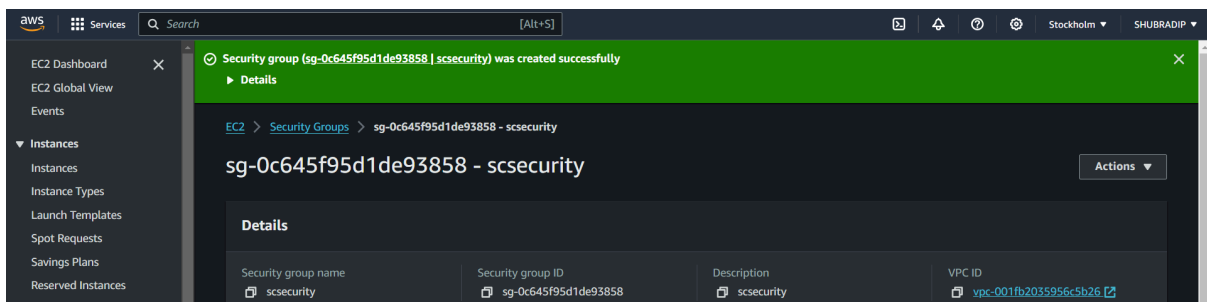
6. Since '**Custom TCP**' protocol is already chosen, input '**4000**' (as specified in the index.js) in the Port range field and select '**0.0.0.0/0**'. Then include the three protocols **SSH**, **HTTP**, and **HTTPS** from the dropdown list and select '**0.0.0.0/0**' for while adding each of them.



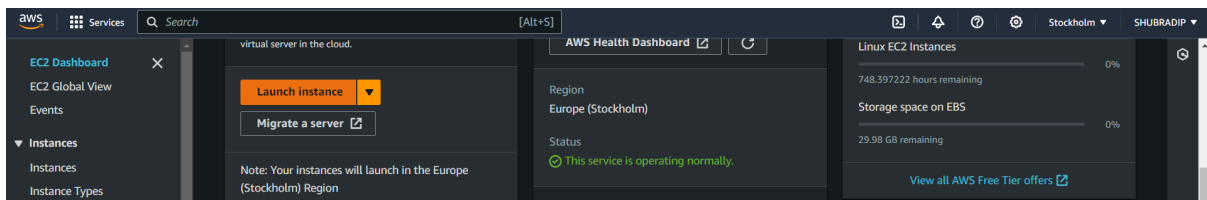
7. Scroll down without making any changes to the Outbound rules, then proceed to click on "**Create security group**".



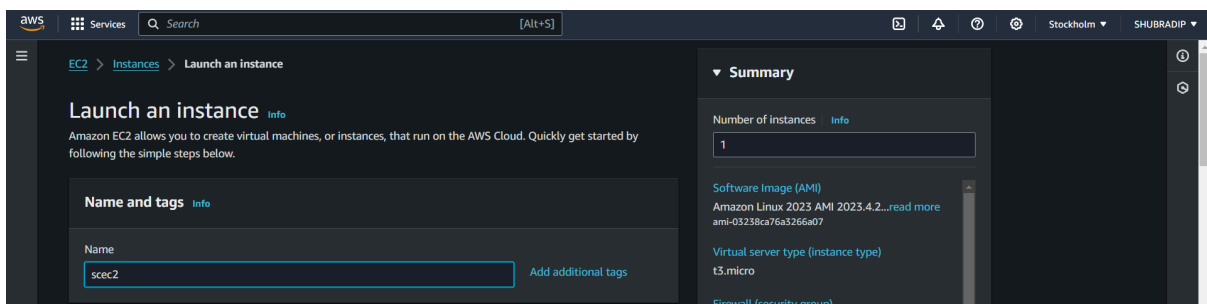
8. The security group is created successfully.



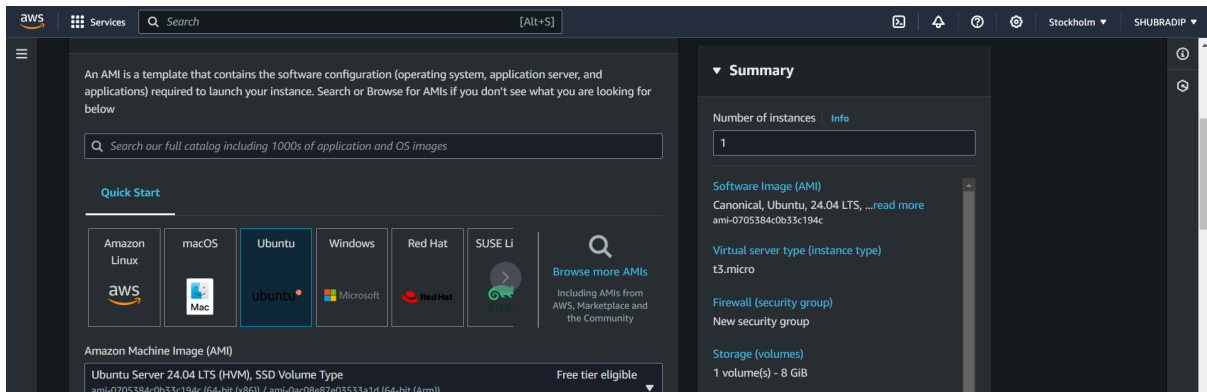
9. Return to the "EC2 dashboard" and select "Launch Instance".



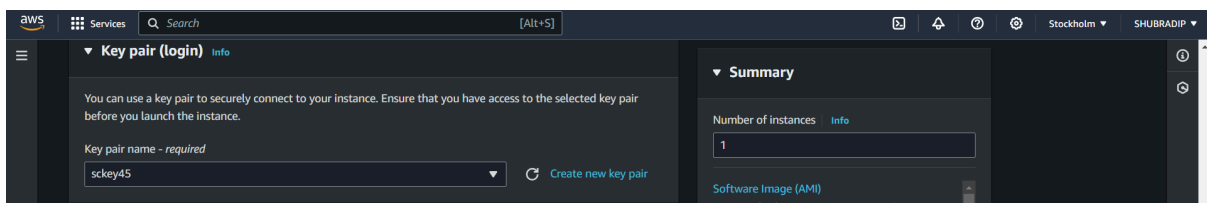
10. Enter a suitable and valid name for the instance (for instance, "soumiec2" in this example).



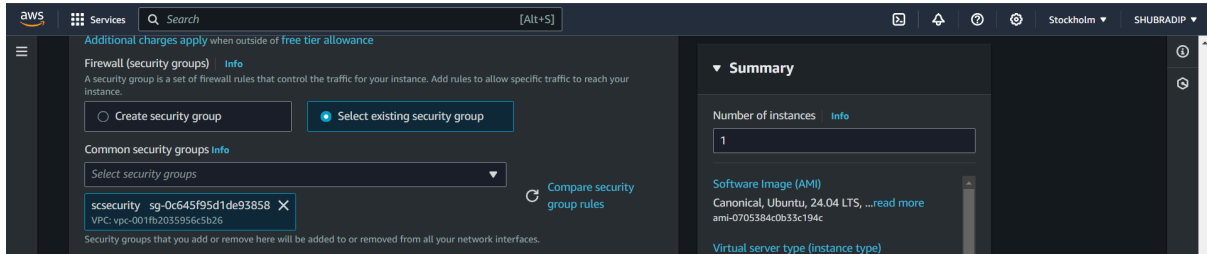
11. Choose "Ubuntu" as the AMI from the available options.



12. Choose an existing key pair, or alternatively, create a new key pair if necessary. In this case, the existing key pair named "**sckey45**" is utilized.



13. Next, click on the "**Select existing security group**" option. From the dropdown menu, choose the existing security group that was created as outlined in the previous steps.

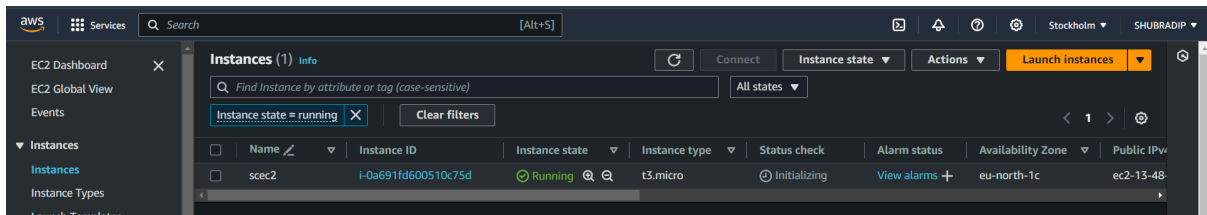


14. Then, proceed to click on "**launch instance**".

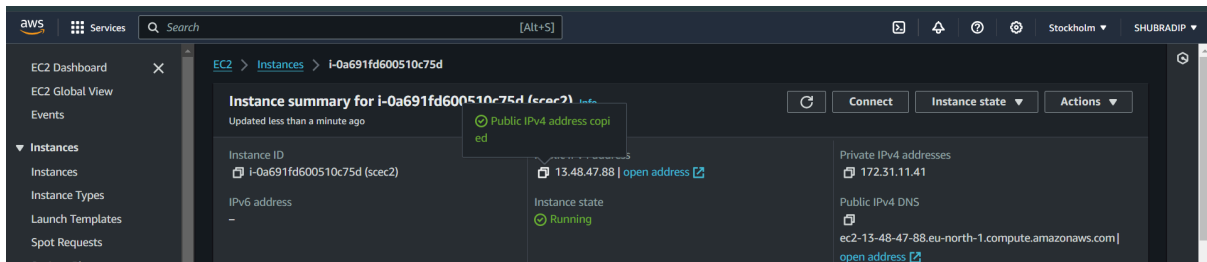
15. The instance has been successfully created utilizing the existing security group.



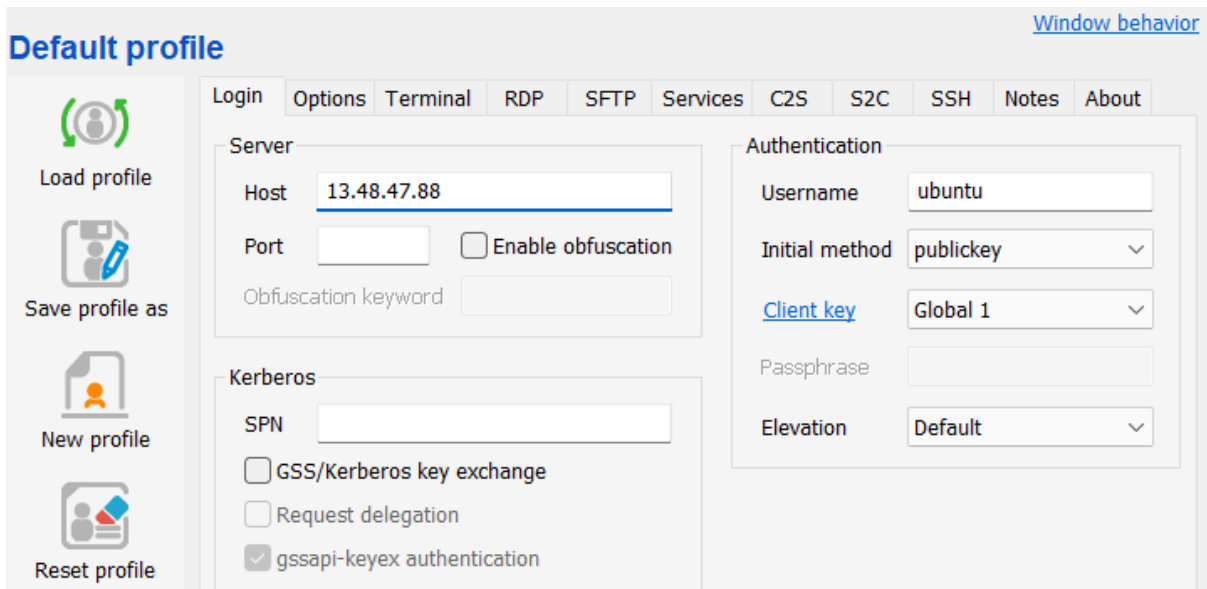
16. Navigate to the "**Instances**" section in the left pane. Then, identify and click on the instance ID of the newly created instance.



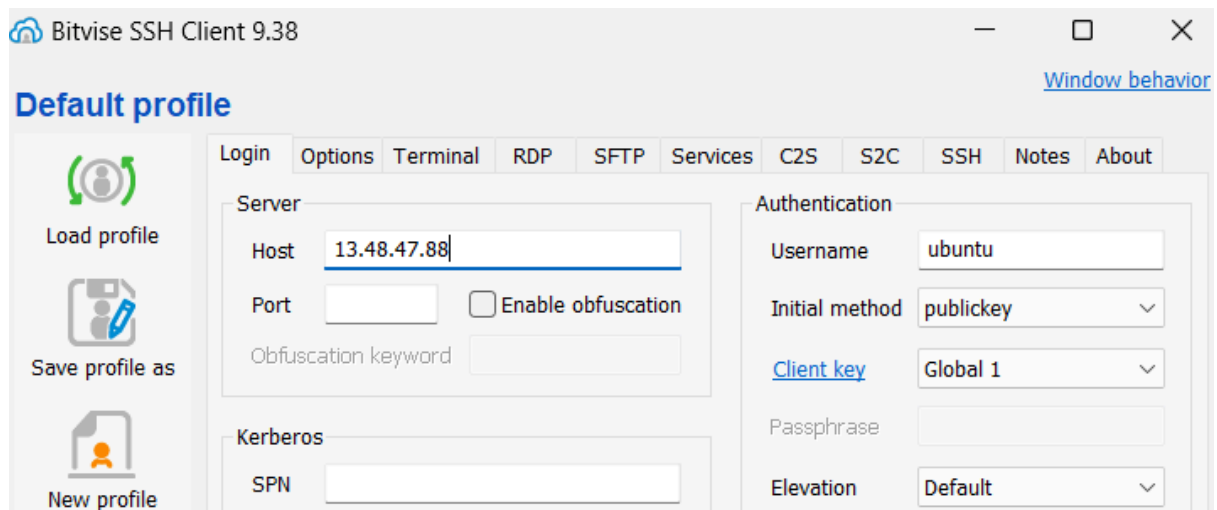
17. Copy the **Public IPv4 address**.



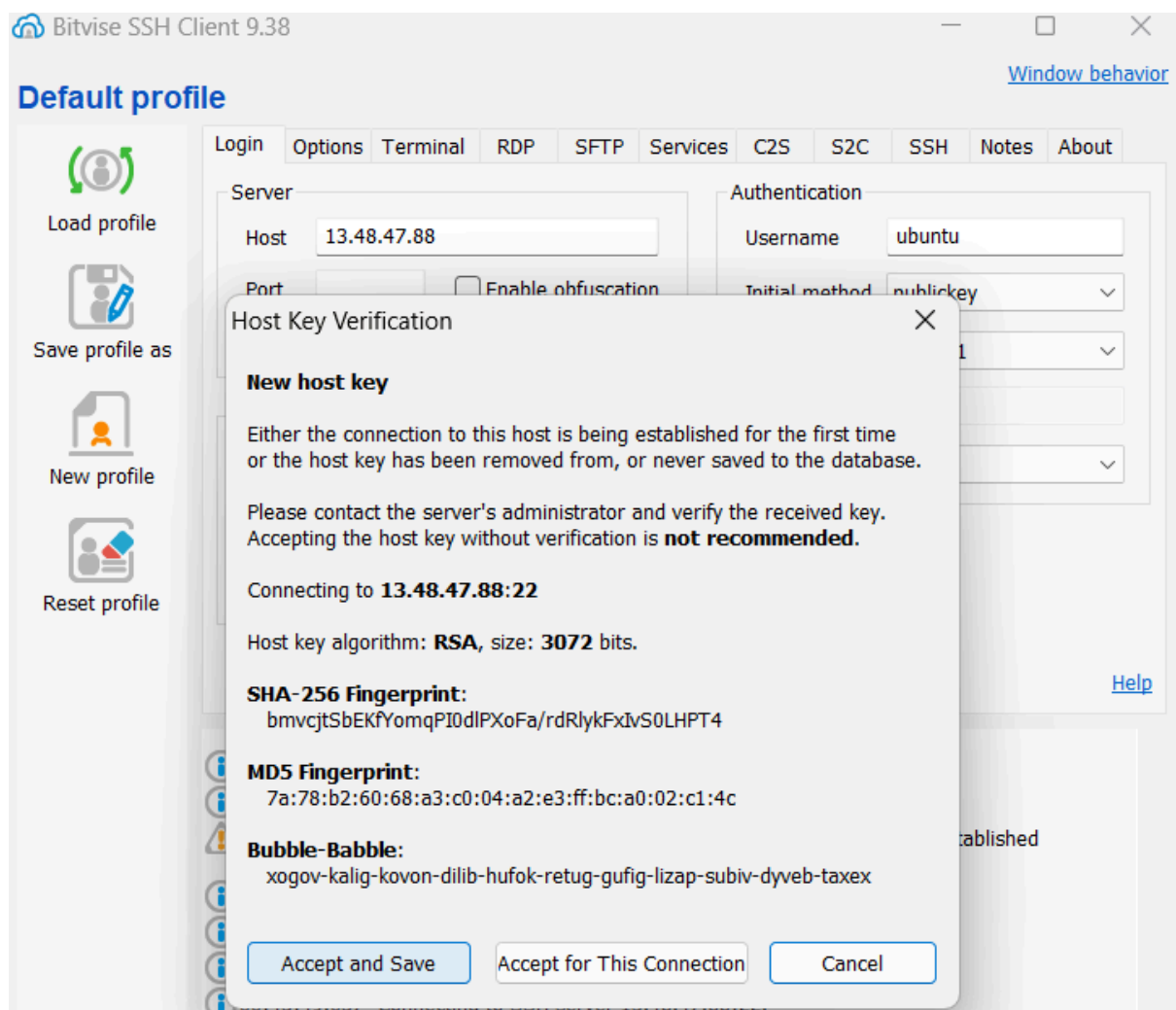
18. Launch Bitwise SSH Client and proceed to click on "**Login**" in the menu bar.



19. Paste the IPv4 address into the host field and enter "**ubuntu**" as the username. Also navigate to the client key manager to verify if the correct key is being used. If not, remove the existing key and import the correct one.



20. After logging in, click on "**Accept and Save**". Subsequently, the following window will appear.



21. Now click on "**New terminal console**". The terminal console opens.

```
ubuntu@13.48.47.88:22 - Bitvise xterm - ubuntu@ip-172-31-11-41: ~

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Fri May 24 19:16:29 UTC 2024

System load:  0.12           Temperature:    -273.1 C
Usage of /:   23.2% of 6.71GB Processes:      113
Memory usage: 26%          Users logged in: 0
Swap usage:   0%           IPv4 address for ens5: 172.31.11.41

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-11-41:~$
```

22. Proceed by entering the commands as listed below.

->***pwd***

```
ubuntu@ip-172-31-11-41:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-11-41:~$
```

-> ***sudo apt-get update***

```
ubuntu@ip-172-31-11-41:~$ sudo apt update
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [89.7 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [89.7 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [89.7 kB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 Packages [1401 kB]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main Translation-en [513 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:9 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [32.2 kB]
```

-> ***sudo apt-get upgrade***

```
ubuntu@ip-172-31-11-41:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following upgrades have been deferred due to phasing:
  dhcpcd-base initramfs-tools initramfs-tools-bin initramfs-tools-core
The following packages will be upgraded:
  apport apport-core-dump-handler curl distro-info-data gir1.2-glib-2.0 klibc-utils less libc-bin
  libc-dev-bin libc-devtools libc6 libc6-dev libcurl3t64-gnutls libcurl4t64 libglib2.0-0t64
  libglib2.0-bin libglib2.0-data libgnutls30t64 libklibc libnghttp2-14 libssl3t64 locales openssl
  python3-apport python3-distupgrade python3-idna python3-problem-report
  ubuntu-release-upgrader-core
```

->**sudo apt-get install nginx**

```
ubuntu@ip-172-31-11-41:~$ sudo apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  nginx-common
Suggested packages:
  fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  nginx nginx-common
0 upgraded, 2 newly installed, 0 to remove and 4 not upgraded.
Need to get 552 kB of archives.
After this operation, 1596 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

->**curl -SL https://deb.nodesource.com/setup_16.x | sudo -E bash -**

```
=====
DEPRECATION WARNING

Node.js 16.x is no longer actively supported!

You will not receive security or critical stability updates for this version.

You should migrate to a supported version of Node.js as soon as possible.
Use the installation script that corresponds to the version of Node.js you
wish to install. e.g.

* https://deb.nodesource.com/setup_16.x - Node.js 16 "Gallium" (deprecated)
* https://deb.nodesource.com/setup_18.x - Node.js 18 "Hydrogen" (Maintenance)
* https://deb.nodesource.com/setup_19.x - Node.js 19 "Nineteen" (deprecated)
* https://deb.nodesource.com/setup_20.x - Node.js 20 LTS "Iron" (recommended)
* https://deb.nodesource.com/setup_21.x - Node.js 21 "Iron" (current)

Please see https://github.com/nodejs/Release for details about which
version may be appropriate for you.

The NodeSource Node.js distributions repository contains
information both about supported versions of Node.js and supported Linux
distributions. To learn more about usage, see the repository:
https://github.com/nodesource/distributions

=====
Continuing in 10 seconds ...
```

->**sudo apt install nodejs**

```
ubuntu@ip-172-31-11-41:~$ sudo apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  nodejs
0 upgraded, 1 newly installed, 0 to remove and 4 not upgraded.
Need to get 27.5 MB of archives.
After this operation, 128 MB of additional disk space will be used.
Get:1 https://deb.nodesource.com/node_16.x nodistro/main amd64 nodejs amd64 16.20.2-1nodesource1 [
27.5 MB]
Fetched 27.5 MB in 1s (43.0 MB/s)
Selecting previously unselected package nodejs.
```


->**git clone <your repository path>**

```
ubuntu@ip-172-31-11-41:~$ git clone https://github.com/shubradip23/Rep45.git
Cloning into 'Rep45'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 4), reused 4 (delta 1), pack-reused 0
Receiving objects: 100% (16/16), 50.74 KiB | 1.24 MiB/s, done.
Resolving deltas: 100% (4/4), done.
```

->**Now go inside the repository using cd and ls.**

```
ubuntu@ip-172-31-18-55:~$ cd Rep45
ubuntu@ip-172-31-18-55:~/Rep45$ cd CSE10
ubuntu@ip-172-31-18-55:~/Rep45/CSE10$ cd Repo1
ubuntu@ip-172-31-18-55:~/Rep45/CSE10/Repo1$
```

->**npm install**

```
ubuntu@ip-172-31-11-41:~/Rep45$ npm install
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 258 packages, and audited 259 packages in 7s

18 packages are looking for funding
  run `npm fund` for details

12 vulnerabilities (10 moderate, 2 critical)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
npm notice
npm notice New major version of npm available! 8.19.4 -> 10.8.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.0
npm notice Run npm install -g npm@10.8.0 to update!
npm notice
```

->**cd /**

->**pwd**

```
ubuntu@ip-172-31-11-41:~/Rep45$ cd /
ubuntu@ip-172-31-11-41:/$ pwd
/
```

-> **cd etc/nginx/sites-available/**

```
ubuntu@ip-172-31-11-41:/$ cd etc/nginx/sites-available/
ubuntu@ip-172-31-11-41:/etc/nginx/sites-available$
```

->**sudo nano default**

```
GNU nano 7.2                                default
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
# https://wiki.debian.org/Nginx/DirectoryStructure
#
# In most cases, administrators will remove this file from sites-enabled/ and
# leave it as reference inside of sites-available where it will continue to be
# updated by the nginx packaging team.
#
# This file will automatically load configuration files provided by other
# applications, such as Drupal or Wordpress. These applications will be made
# available underneath a path with that package name, such as /drupal8.
#
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
```

->Scroll down and track “location” and comment the line using “#” along with two remaining lines.

```
server {
    # location / {
    #     # First attempt to serve request as file, then
    #     # as directory, then fall back to displaying a 404.
    #     try_files $uri $uri/ =404;
    # }
}
```

->Copy the code provided and write it below “Server name”.

```
GNU nano 7.2                                default *
server_name _;
location / {
    proxy_pass http://localhost:4000;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection 'upgrade';
    proxy_set_header Host $host;
    proxy_cache_bypass $http_upgrade;
}
```

->Now press “CTRL+x” and write “y” then press enter.

```
# deny access to .htaccess files, if Apache's document root
Save modified buffer?
Y Yes
N No      ^C Cancel
```

-> Now go inside the repository using cd.

->sudo systemctl restart nginx

->node index.js

```
ubuntu@ip-172-31-11-41:/etc/nginx/sites-available$ cd
ubuntu@ip-172-31-11-41:~$ cd Rep45
ubuntu@ip-172-31-11-41:~/Rep45$ sudo systemctl restart nginx
ubuntu@ip-172-31-11-41:~/Rep45$ node index.js
Started server
```

23. Open a new browser window and enter the IPv4 address.

