**Secure System for Person Recognition using Blockchain and Deep Learning**

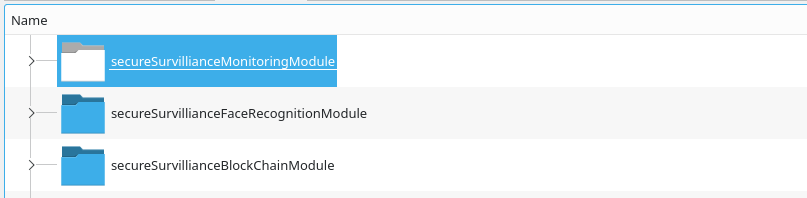
Steps to execute:

1. Install Prerequisite:

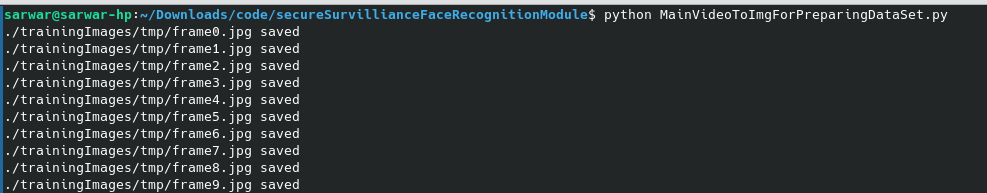
Install below:

* 1. Python 3.10
  2. Anaconda
  3. Pip
  4. Flask~=1.1
  5. requests~=2.22
  6. opencv 4.6.0
  7. jproperties

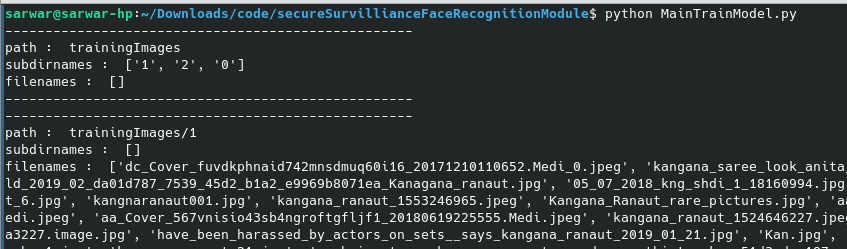
1. System/machine details:
   1. Number of machine >= 2 (to create physical nodes)
   2. OS = Windows, Ubuntu
2. Get the latest code.



1. Train the model:
   1. Go to secureSurvillianceFaceRecognitionModule and run MainVideoToImgForPreparingDataSet.py

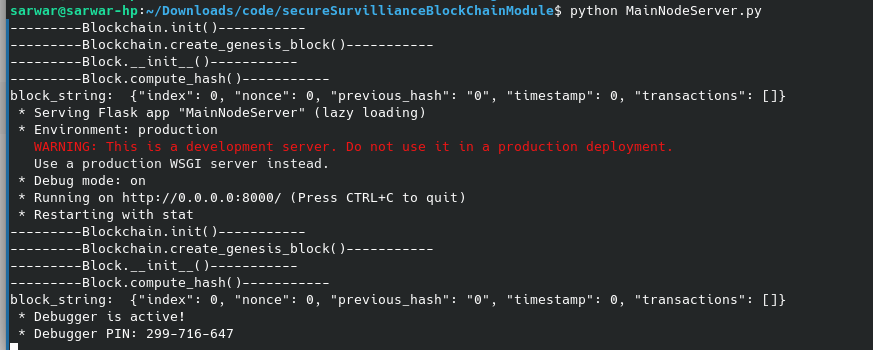


* 1. It will capture the video and save in image form in trainingImages/tmp folder.
  2. Rename this folder by a next id eg 3.
  3. Run MainTrainModel.py to train the model.

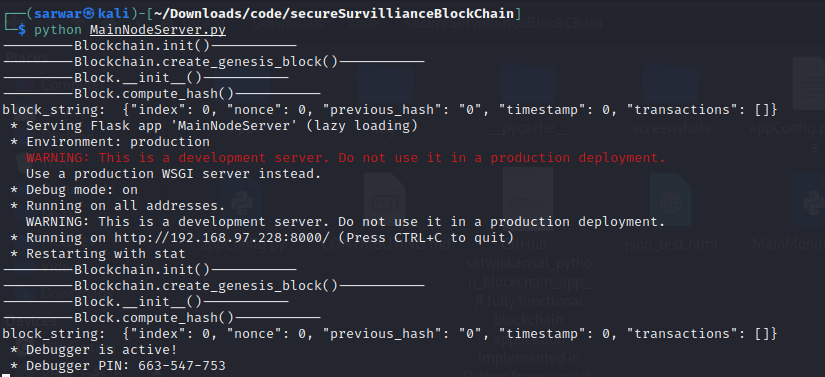


* 1. It will create a YML file with file name “trainingData.yml”.

1. Create node using Node Server module:
   1. Go to secureSurvillianceBlockChainModule and run MainNodeServer.py to create a node.

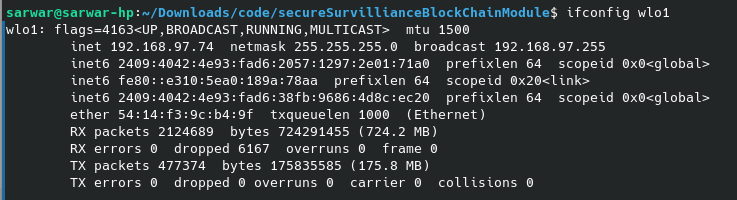


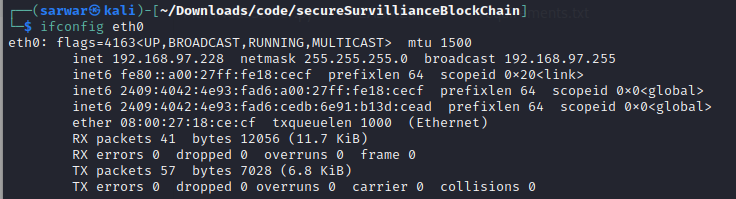
* 1. Apply above steps to other machines also to create a new node in different machine.



* 1. Don’t close it.

1. Synchronize Node:
   1. Note down both machine IP addresses.





* 1. Ping both machines to ensure they are connected.
  2. Run below to sync each other.

curl -X POST

http://<first node ip>:8000/register\_with

-H 'Content-Type: application/json'

-d '{"node\_address": "http://<second node ip>:8000"}'

curl -X POST

http://<second node ip>:8000/register\_with

-H 'Content-Type: application/json'

-d '{"node\_address": "http://<first node ip>:8000"}'

Eg:

curl -X POST \

[http://192.168.97.74:8000/register\_with](http://192.168.97.95:8000/register_with) \

-H 'Content-Type: application/json' \

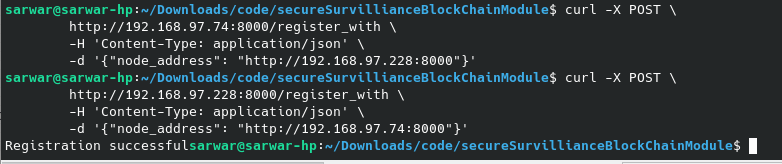
-d '{"node\_address": "[http://192.168.97.95:8000](http://192.168.97.95:8000/)"}'

curl -X POST \

<http://192.168.97.95:8000/register_with> \

-H 'Content-Type: application/json' \

-d '{"node\_address": "[http://192.168.97.74:8000](http://192.168.97.74:8000/)"}'



1. Run Monitoring module:
   1. Goto secureSurvillianceMonitoringModule.
   2. Open application appConfig.properties file and add below line.

MINING\_NODE\_ADDRESS= [http://<Node IP>:8000](http://192.168.97.74:8000/)

Eg.

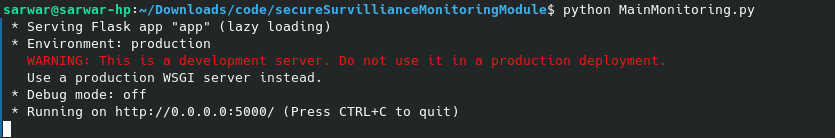
add below first node:

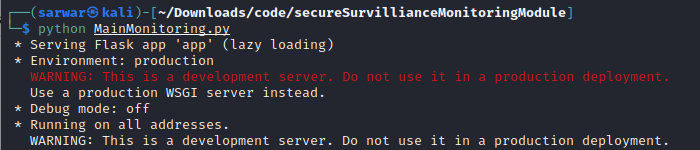
MINING\_NODE\_ADDRESS= http://192.168.97.74:8000

add below in second node:

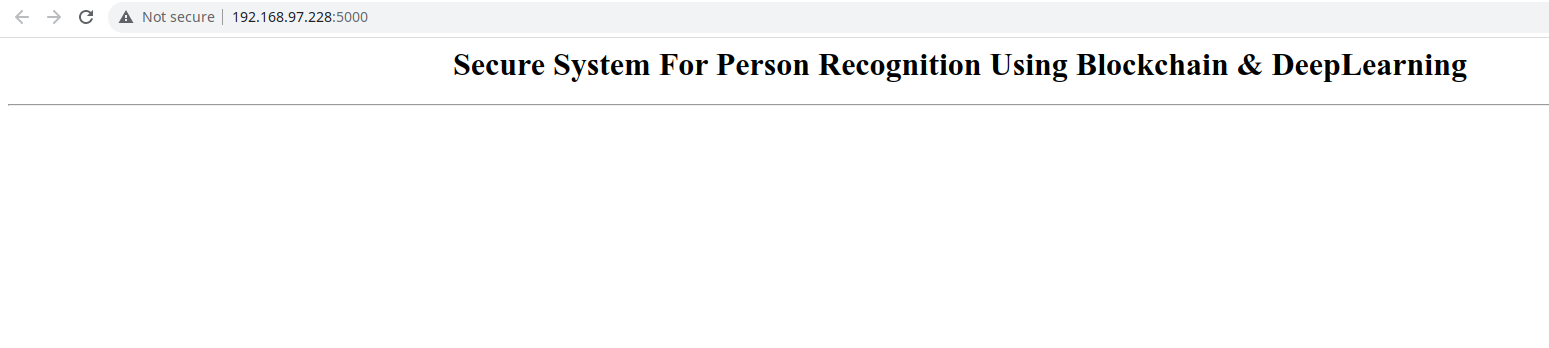
MINING\_NODE\_ADDRESS= [http://192.168.97.228:8000](http://192.168.97.74:8000/)

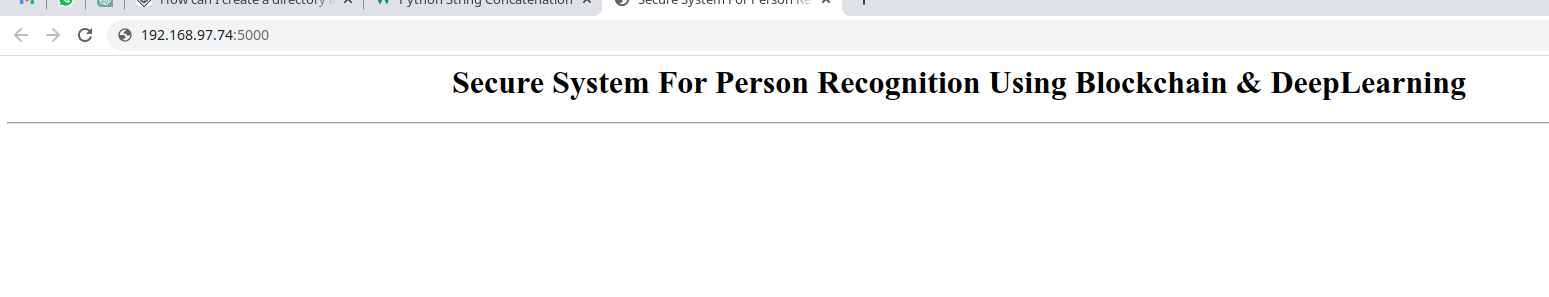
* 1. Run MainMonitoring.py in both machines.





* 1. Don’t close it.
  2. Open [http://192.168.97.74:5000/](http://192.168.97.228:5000/) & <http://192.168.97.228:5000/> in browser for respective machines.





1. Run Face Recognition module:
   1. Go secureSurvillianceFaceRecognitionModule and open appConfig.properties and add below lines.

MINING\_NODE\_ADDRESS= http://<Node IP>:8000

MIN\_TIME\_BETWEEN\_TWO\_TXNS=3

NODE\_LOCATION=<Node/Camera location>

Eg. Add below in first node:

MINING\_NODE\_ADDRESS= http://192.168.97.74:8000

MIN\_TIME\_BETWEEN\_TWO\_TXNS=3

NODE\_LOCATION=DIAT CSE Department (Ubuntu Machine), Pune

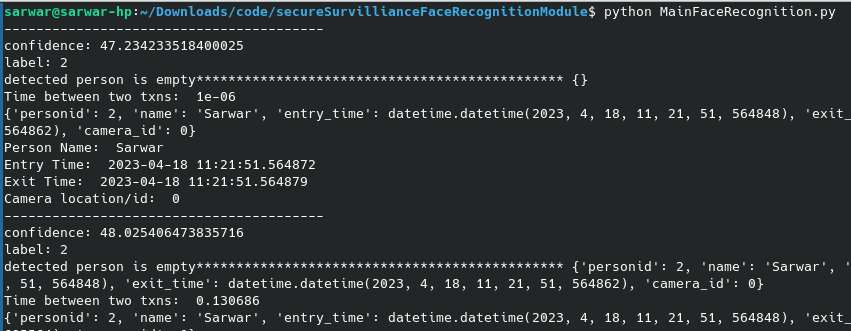
Add below in second node:

MINING\_NODE\_ADDRESS= http://192.168.97.228:8000

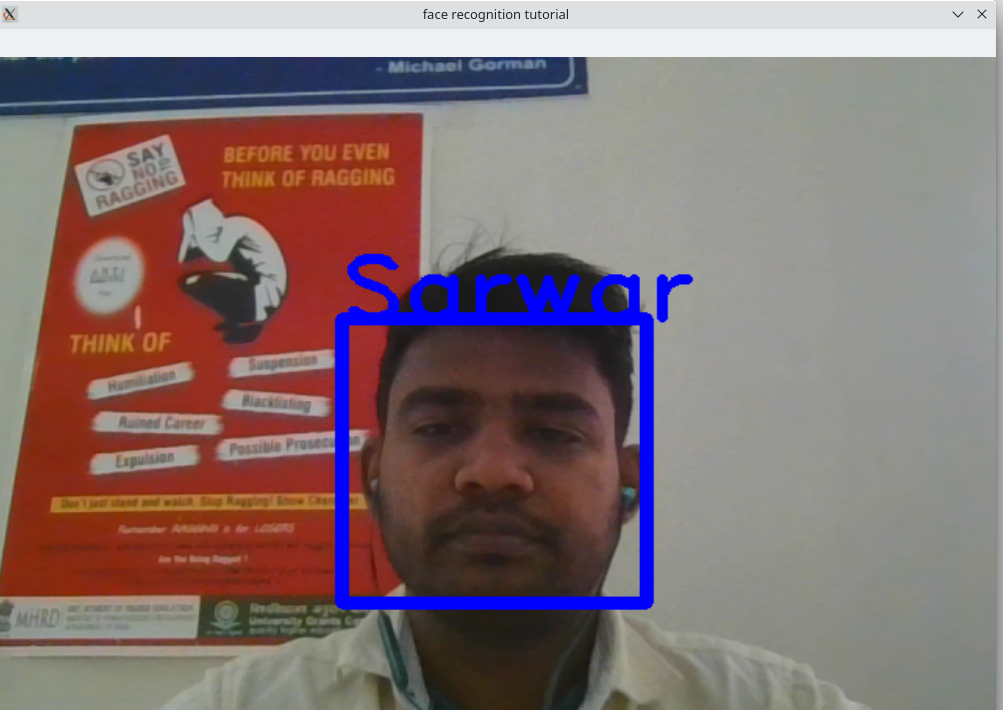
MIN\_TIME\_BETWEEN\_TWO\_TXNS=3

NODE\_LOCATION=DIAT CSE Department (Kali Machine), Pune

* 1. Run MainFaceRecognition.py in both machine.



* 1. It will open window to capture the video. Move your face for few seconds to make multiple entries in blockchain.



* 1. See the same in another machine.
  2. The first machine should make an entry with location “Ubuntu Machine”, and the second machine should make an entry with location “Kali Machine” according to the appConfig.properties file configuration.

1. See the recognized person in Monitoring window:
   1. Go to earlier opened browser and see the updated data.
   2. Go to earlier opened browser of second node and see the data. Both browser should be giving the same output which shows both are synced.