EXTENDED VERSION OF BLOODHOUND

1. Introduction

- **Overview**
  This website provides visualization of a graph database with different features like BloodHound. Besides, it also integrates external BloodHound tools as well as algorithms related to network security (Iterative and AAAI)
- **Website info**
  - Hosting server: ExpressJS
  - Languages: HTML, Javascript
  - Stylesheet: CSS/SCSS
  - Framework: VueJS (version 1)
  - Graph visualization: based on SigmaJS with modifications
  - Database: Neo4J
  - External algorithm: Python (with NodeJS spawn children)

2. Installations

a) NPM

- **ExpressJS**
  npm install -g express-generator
  npm install
- **SCSS**
  npm install -g sass
- **SCSS compilation**
  npm install node-sass nodemon --save-dev
  npm install -g concurrently
- **Neo4J Driver**
  npm install neo4j-driver
- **SigmaJS**
  npm install graphology sigma

b) Python packages

As the website includes many Python tools. Therefore, below are the basic packages that is required. Notice, depending on your Python version as well as pip/pip3, you might have to install some packages before installing our required ones.

- **ShotHound (CornerShot and neo4j package)**
  pip install cornershot neo4j
  When installing CornerShot, you might be required to install impacket first. If so, please have a look at
  https://github.com/SecureAuthCorp/impacket
- **Torch (AAAI)**
  pip3 install
  https://files.pythonhosted.org/packages/d3/91/1b2871d6c8ca079254deae5872af32e02e9a85f07dd0834e8b3489ce138f/torch-0.4.1.post2-cp37-cp37m-manylinux1_x86_64.whl
  pip3 install --user torchvision

3. **DATABASE**
   - **Neo4J:** Please use BloodHound application to load JSON files to Neo4J, or do what you prefer.
   - **Iterative cut:** All our features utilize connection between Neo4J and NodeJS to run query back and forth. However, "Iterative Cut" requires a whole database, meaning that exporting thousands or millions of nodes and edges will potentially crash Neo4J and would take much time, especially when we need to cut multiples times, we have to load the database that much.
     ⇨ Prepare yourself a set of JSON folders (6 files for Computers, Users, Domains, Ous, Groups, GPOs), similar to running SharpHound
     ⇨ Navigate to *path/Research_Express/routes/DATABASE*, there, we provide a set of mock database. If you want to run Iterative Cut on your own database, please replace our 6 files with your files.
   - **AAAI:** This set of agorithms requires different an explicit graph, beyond Neo4J. Therefore, we have provide a separate graph in advance. If you want another, you have to manually build the graph (gpickle) and from there, you can just interact with it at ease. Please follow the below steps:
     - Navigate to *path/Research_Express/routes/AAAI*, open **buildgraph.py**.
       + At line 13, you can change the default name of the graph.

+ At line 14, modify to the absolute path of your database JSON file. Notice, to obtain this JSON file, go to Neo4J, use this query *MATCH p = (a) –[r]->(b) RETURN p* , and store data received in a JSON file.

+ At line 15, replace with your set of DA

+ At line 16, replace with your deleted snowball

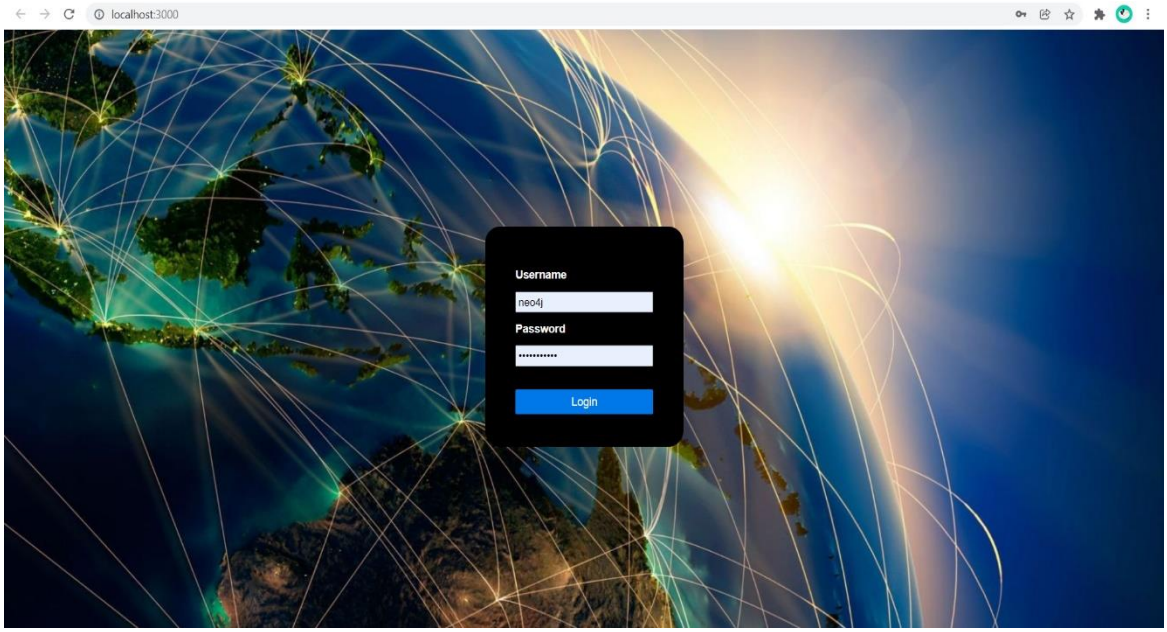
+ In your Command Prompt, run the file.

4. Let's run the Website

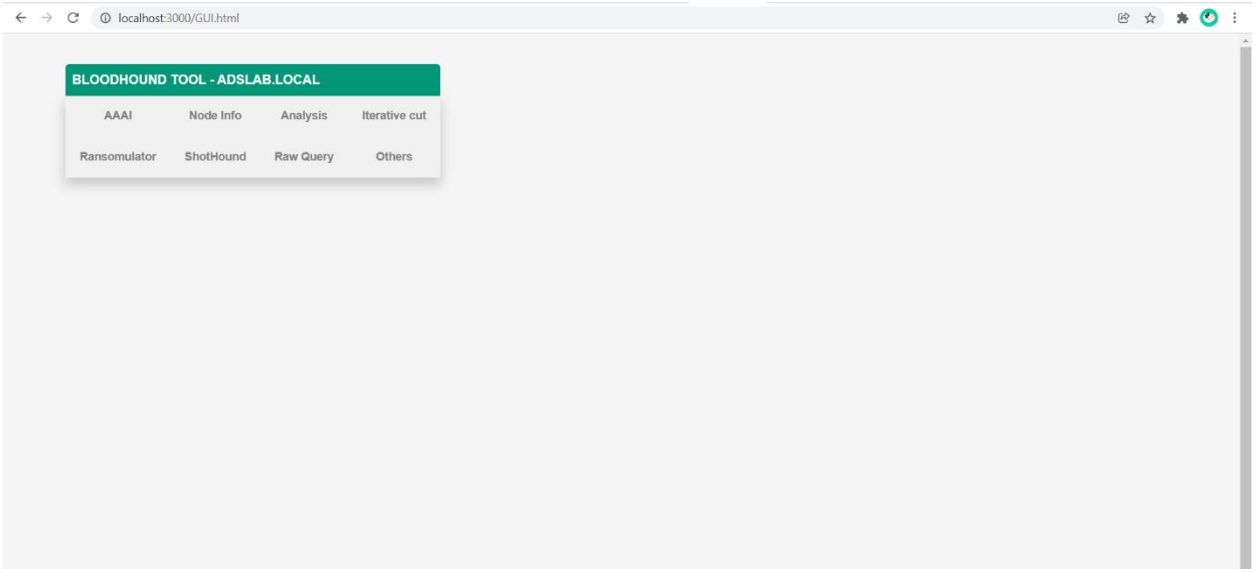
- In your Command Prompt, navigate to *path/Research_Express*
- Run *npm start*, and wait until it renders all, like below:

```
C:\Users\DELL\Documents\Research_Express\routes\AAAI>npm start

> research-express@0.0.0 start
> concurrently "node ./bin/www" "npm run watch-css"

[1]
[1] > research-express@0.0.0 watch-css
[1] > nodemon -e scss -x "npm run build-css"
[1]
[1] [nodemon] 2.0.15
[1] [nodemon] to restart at any time, enter `rs`
[1] [nodemon] watching path(s): *.*
[1] [nodemon] watching extensions: scss
[1] [nodemon] starting `npm run build-css`
[1]
[1] > research-express@0.0.0 build-css
[1] > node-sass --include-path scss ./public/stylesheets/style.scss ./public/stylesheets/style.css
[1]
[1] Rendering Complete, saving .css file...
[1] Wrote CSS to C:\Users\DELL\Documents\Research_Express\public\stylesheets\style.css
[1] [nodemon] clean exit - waiting for changes before restart
```

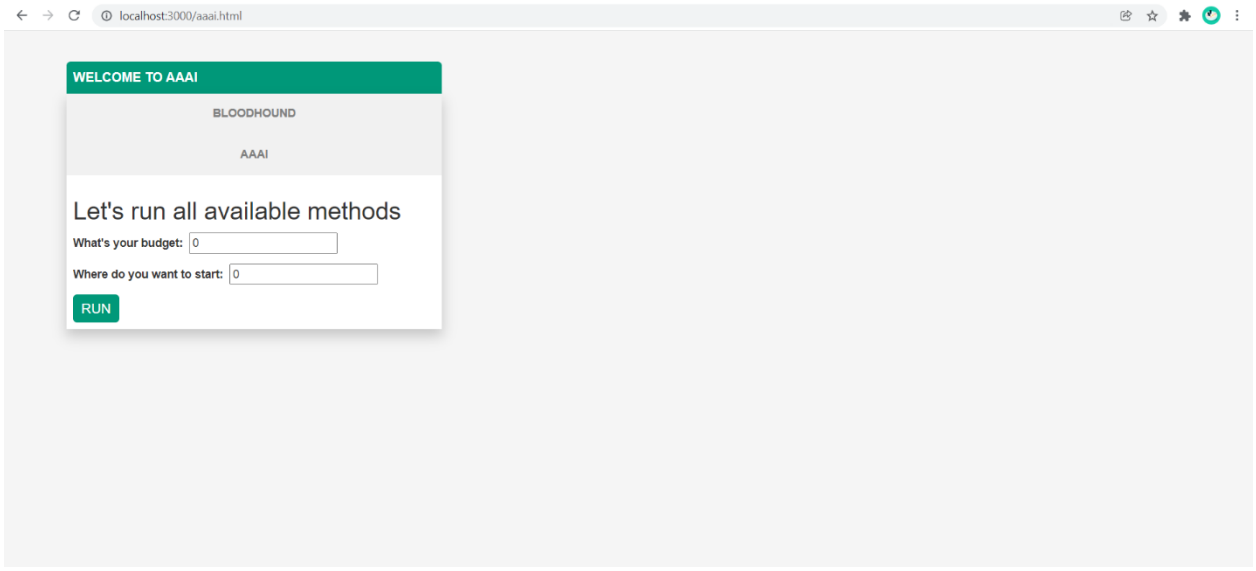
- Open your browser, and open Localhost 3000
- If you see the screen like the picture below, welcome to our Website. Otherwise, if there is error in browser and in terminal, please revise the above steps.



- After login, you should see the default BloodHound:



- If you want to move to AAAI, simply click on AAAI button



- To return back to BloodHound, please click the button BloodHound

- **For Ransomulator and ShotHound**, please use the full name of the object.