Initially, I scraped the information by extracting the HTML source code (refer to ranking\_methods.py). However, I later realised that I could simply extract the JSON files sent over the network to the website (refer to YesWeHack\_rankings.py). The following analysis will be made from the latter code.

Throughout the code, I occasionally used the *tqdm* module to track the code running.

I first verified that the lists of hunters from *any time period* are all subsets of the ‘Top 100’ hunters. Hence I did the analysis using the Top 100 list.

1. Number of Private/Public Profiles:

* Only public profiles have further information, thus I identified them first.
* I then scraped the public profile websites.

Graphical user interface, text

Description automatically generated

1. Number of Public Profiles with GitHub:

* I am more interested in a programmer’s GitHub profile as compared to other social media.

Text

Description automatically generated

1. Contributing High Impact:

* I noticed that some profiles had a higher *Impact* than Hunter of Rank #1.
* I found the hunters with a higher impact and noticed that the impact does not correlate with his/her points and rank.
* Since duplicate reports do not contribute to impact, it can be hypothesised that Hunter #1 may have submitted many duplicates, thus accumulating high points but low impact.

A picture containing text

Description automatically generated  
A picture containing text

Description automatically generated

1. Top 5 Profiles (public):

* I wanted to search into the details of their Hacktivities. Given the large amounts of data, I considered the statistics of the Top 5 profiles by rank.

Graphical user interface, text

Description automatically generated

1. Hacktivity Count:

* Reports are not equal to Hacktivity. In fact, Hacktivity 2Reports.
* Possibly, each report submitted needed to be modified at least once more.

Text

Description automatically generated

1. Top Bugs:

* Based on the Hacktivity, I found the Top Bug Type of the top 5 profiles, and the percentage occurrence of the Bug Type.
* From the results, it can be inferred that Hunter “freesec” is extremely specialised, with 53.6% of bug types being “”.
* As compared to Hunter “freesec”, the other top hunters are much less specialised, although they have their own focus as well. The other top hunters’ top bug percentage frequencies are less than half of that of Hunter “freesec”.

Text

Description automatically generated

1. Top Status:

* I was curious about the status of reports submitted by the top 5. Given that so many reports have been submitted, how many reports have yet to be processed?
* From the results, it can be inferred that the top hunters are very active, even till date. Almost half of their reports still have a status of *‘new’*.
* Top hunters are submitting reports faster than the reports can be processed.

A screenshot of a computer

Description automatically generated with low confidence

Further possible analysis include:

* Number of reports over the time periods, using a line graph to observe possible trends
* Top Bug Tags in each time period
* Percentage change in Hacktivity count for some of the Top Bug Tags over the time periods