

## Details:

Please find the codebase at: [https://github.com/TMonale/IEUK\\_Engineering\\_SSP](https://github.com/TMonale/IEUK_Engineering_SSP).

## Brief:

To conduct the analysis of the `sample-log.log` file, I began with writing a python script, `log_to_csv_conversion.py`, to convert the file into a csv format, `output.csv`. I then performed a series of analysis using python, `data_analysis.py`.

## Process:

Out of 432,096 data entries, initial analysis showed unusually high traffic from specific European and North American IPs. Below are some key findings:

IP Address	Frequency
45.133.1.2	5400
45.133.1.1	5400
35.185.0.156	3600
194.168.1.2	1859
194.168.1.6	1855

Countries	Frequency
UK	54215
US	43175
SE	39740
NL	39723
DE	39629

Upon manually reviewing the website's traffic logs, it was found that the above IP addresses had made several requests in rapid succession. This unhuman-like behaviour suggested that bots were accessing the website.

I then researched [bot detection](#) and found a list of [common bot user agents](#). I created a summary of all user agents used in the traffic logs and filtered them by the list of known bot agents. A total of 18,676 bot requests were found from 8 IP addresses originating from 5 countries. This accounts for 4% of total website traffic.

## Recommendations:

Below are my three recommendations for bot management. I've assumed that the company is using a basic web server with no dedicated bot mitigation.

1. Using Cloudflare's Free or Pro Plan: implement their WAF (web application firewall) for bot protection, rate limiting, and DDoS mitigation. It is affordable and offloads traffic from their servers.
2. User agent blocking: blocking the identified IPs and user agents directly on your web servers (Nginx, Apache config, etc.). This is an immediate solution.
3. Basic rate limiting: configuring your web server to limit requests from single IPs or suspicious patterns to prevent flooding the site.