Marawan Mohamed Farouk Moharrem
ID : 2205066

Report

This report shall explain what was done in the log file and why when i clicked enter it made a .txt file like a report containing what was done.
When i clicked enter the .txt file was made based on the bash script that I made , so when it runs it makes a report of the things that were read and store them in a .txt file that is saved in my files on Kali

There was a report that appears after i run the script and its explanation is here :
It is a detailed analysis of the Nginx access log file located at `/home/marawan/Downloads/access.log`. The log contains 10,000 HTTP request entries spanning over four days. The purpose of the analysis is to identify usage patterns, detect anomalies, and suggest improvements based on the collected server access data.
The log recorded a total of 10,000 requests. The overwhelming majority of these, 9,952, were HTTP GET requests, which are typically used to retrieve data from the server.

Over the four-day period, 1,753 unique IP addresses interacted with the server. This figure shows the breadth of users accessing the site.

Some IP addresses appeared significantly more frequently than others. The most active IP, `66.249.73.135`, made 482 requests alone. This IP belongs to Google's web crawler,

indicating that search engine indexing is a significant portion of the traffic. Other IPs such as `46.105.14.53`, `130.237.218.86`, and `75.97.9.59` also made hundreds of requests each, possibly indicating either legitimate users or automated scanning tools. High activity from individual IPs should be monitored for potential abuse.

Out of the total 10,000 requests, 220 resulted in client or server errors (HTTP status codes in the 4xx and 5xx range), making up 2% of all requests. This is within a reasonable margin for most servers, but still worth monitoring.

The log spans 4 days, with an average of 2,500 requests per day. This gives a useful baseline for daily server load and can help in identifying anomalies or spikes in traffic on specific days.

Failures were spread across the four days fairly evenly, with the highest number—66 failures—occurring on both May 18 and May 19.

The server saw a fairly consistent spread of requests throughout the day. However, peak activity was concentrated in the afternoon and early evening hours (from 12:00 to 21:00), with the highest hourly traffic reaching nearly 500 requests.

The majority of responses were HTTP 200 (OK), indicating successful request handling. Status 304 (Not Modified) responses were also frequent, which is common when browsers use cached data. A small number of 404 (Not Found) and 500 (Internal Server Error) codes were also observed. These indicate broken links or server-side problems that might require further investigation. Status codes like 301 (Permanent

Redirect) and 206 (Partial Content) were present in expected volumes, typically related to normal web browsing or file downloads.

And at the end Failure requests were spread out, but certain times—especially between 05:00 and 09:00—showed slightly elevated failure counts.

This is the explanation of the report/output when the bash is run and accessed by the log file

And thank you