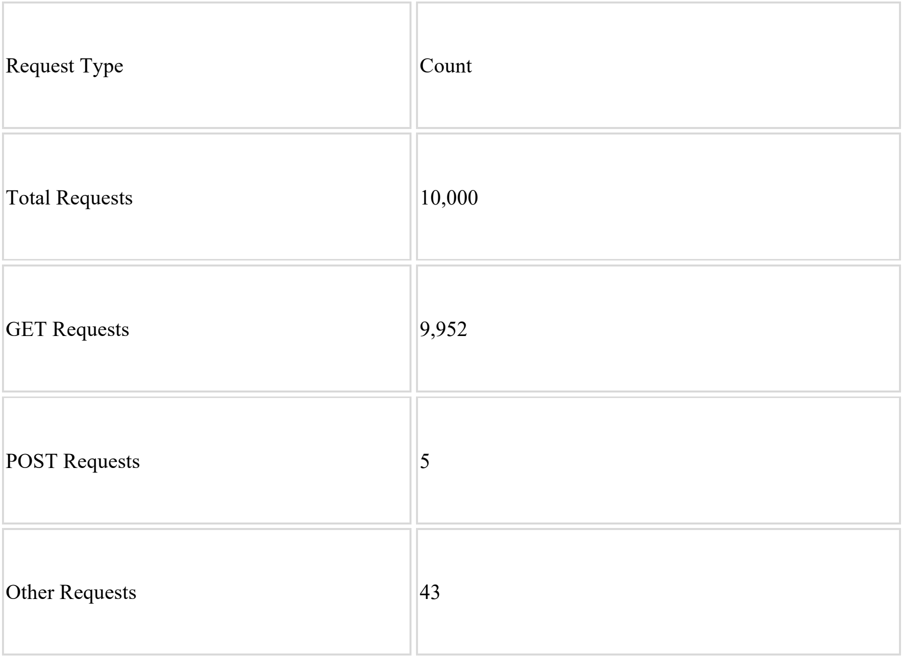
Apache Log File Analysis Report

by:Rewan ELwardany
2205218

01. Request Counts



Request Type	Count
Total Requests	10,000
GET Requests	9,952
POST Requests	5
Other Requests	43

- GET= normal webpage loads.
- POST = form submissions — *very low* → may indicate lack of usage or blocked forms.
- Other = possibly HEAD, PUT, DELETE → used by bots or API scanners.

02. Unique IP Addresses

IP Address	GET Requests	POST Requests
46.105.14.53	364	0
130.237.218.86	357	0
208.115.111.72	83	0
209.85.238.199	102	0
50.16.19.13	113	0

Total Unique IPs: There were 1,753 unique IP addresses making requests during the observed period. This indicates a relatively large pool of clients, possibly including search engine crawlers, legitimate users

GET/POST by IP:Each IP’s usage pattern was broken down to show how many GET and POST

requests were made.

03. Failure Requests

Metric	Value
Total Requests	10,000
Failed Requests	220
Failure Rate	2.20%

- Out of 10,000 total HTTP requests, **220 failed**, representing **2.2%** of the total traffic.
- These failures are typically categorized into:
  - **4xx errors (Client-side)**:
    - Examples:
      - 404 Not Found: Missing page or broken link.
      - 403 Forbidden: Access denied due to permissions.
      - 400 Bad Request: Invalid input.
  - **5xx errors (Server-side)**:
    - Examples:
      - 500 Internal Server Error: Something broke in backend logic.
      - 502 Bad Gateway, 503 Service Unavailable: Server is down or overloaded.

04. Top User (Most Active IP)

IP Address	Request Count
46.105.14.53	364
130.237.218.86	357
50.16.19.13	113

- The top IP made 364 requests
- Total Requests: 482 GETs

  This IP address seems to belong to a search engine crawler, possibly Googlebot or another similar bot. The elevated request frequency is generally considered normal, especially when the server is configured to allow indexing. However, if the activity becomes excessive or disrupts server performance, it might require closer monitoring or additional filtering to ensure optimal resource usage.

05. Daily Request Averages

1-Average Requests per Day: 2500.00

The average number of requests per day stands at 2,500. This suggests a consistent level of traffic to the server, with requests being distributed throughout the day. Monitoring this metric is essential to ensure that the server can handle the load efficiently without performance degradation

06. Failure Analysis (Top Days with Failures)

Date	Failed Requests
19/May/2015	66
18/May/2015	66
20/May/2015	58
17/May/2015	30

The analysis shows that May 19th and 18th had the highest number of failed requests, both at 66, followed by May 20th with 58, and May 17th with 30. These failures could stem from issues like incorrect requests, server misconfigurations, or application errors. Investigating the types of errors will help identify and resolve the underlying causes to improve server performance.

07. Request Patterns by Hour

Requests per hour for each day:

Date	Hour 01	Hour 10	Hour 12	Hour 14	Hour 16	Hour 18
17/May/2015	1151	74	115	120	126	118
18/May/2015	1229	132	120	122	114	123

Requests per hour (average across all days):

Hour	Requests
01	0
10	443
12	462

This analysis shows how requests vary by hour across different days, highlighting peak traffic times and giving insight into server load patterns. It can help in optimizing performance and identifying potential bottlenecks during high-traffic hours.

Peak Hour: Hour 01 shows the highest requests, especially on 17/May/2015 (1151 requests).

Afternoon Spike: Requests peak around Hour 14 (498 requests), Hour 15, and Hour 16.

Low Traffic: Hours 00:00 to 09:00 have minimal or no requests.

Overall Trend: Traffic is low early in the day, increases in the afternoon, and peaks at Hour 14, showing user engagement or system demand spikes later in the day.

08. Request Trends

Hourly Trends (Average across all days):

Hour	Requests
12	462
13	475
14	498

Daily Trends:

Date	Total Requests
17/May/2015	1632
18/May/2015	2893
19/May/2015	2896
20/May/2015	2579

- **Peak Hour**: Hour 14 (14:00) had the highest request count with 498 requests, indicating increased activity in the afternoon.
- **Peak Day**: 19/May/2015 experienced the highest number of requests, totaling 2896, suggesting a higher level of traffic or demand on this day.
- **Trend**: Request volume increases consistently over time, particularly evident in the rise from 17/May (1632 requests) to 19/May (2896 requests). The trend shows growing traffic, likely due to user behavior or promotional activity.

09. Status Codes Breakdown

Frequency of Status Codes:

Status Code	Frequency	Description
200	9126	OK - Request was successful
304	445	Not Modified - The resource hasn't changed
404	213	Not Found - The requested resource doesn't exist
301	164	Moved Permanently - Resource has been permanently moved to a new location
206	45	Partial Content - Part of the resource is returned
500	3	Internal Server Error - Server encountered an error
416	2	Range Not Satisfiable - Requested range is not valid
403	2	Forbidden - Access to the resource is denied

10. Most Active User by Method

Most Active User by Method

Method	IP Address	Request Count
GET	66.249.73.135	482
POST	78.173.140.106	3

- **GET Requests:**
  The IP address 66.249.73.135 is the most active for GET requests. This address likely belongs to a crawler or search engine bot, such as Googlebot, which is expected for publicly accessible web pages.
- **POST Requests:**
  The IP 78.173.140.106 made only 3 POST requests, suggesting minimal use of forms or data-submitting functionalities, like login or registration.

11. Patterns in Failure Requests

Failed Requests by Hour

Hour	Failed Requests
00–09	0
10	12
11	11

Failed Requests by Day

Date	Failed Requests
17/May/2015	30
18/May/2015	66
19/May/2015	66
20/May/2015	58

- No failures occurred between 00:00 and 09:00, indicating a quiet period or low traffic.
- Failures start to increase after 10:00 AM and peak between 10:00 and 17:00, which aligns with typical peak usage hours — possibly leading to more invalid requests or server strain.
- May 18th and 19th had the highest number of failed requests (66 each), which may indicate anomalies, temporary issues, or configuration errors during those days that should be investigated.

1. Reducing the Number of Failures

- **Fix Broken Links (404 Errors):**
  The majority of failed requests (213 out of 220) returned a 404 status, indicating that the requested resources were not found.
  Suggested Action: Identify and audit the most frequently requested missing URLs. Restore the missing content, fix incorrect internal links, or implement 301 redirects to guide users to valid pages.
- **Handle Range Requests Properly (416 Errors):**
  A couple of 416 errors suggest improper handling of partial content requests, possibly for media files.
  Suggested Action: Make sure the server supports byte-range requests where necessary (e.g., for video or audio). If not supported, strip the Range header and return the full content with a 200 OK response.
- **Prevent Server-Side Failures (5xx Errors):**
  Although limited in number, server errors indicate deeper issues in application handling.
  Suggested Action: Improve error handling by adding try-catch blocks around critical endpoints. Enhance logging for better traceability and set up alerts for any occurrence of 5xx errors.

2. Target High-Failure Periods

- **Morning Spike in Failures (05:00–09:00):**
  The failure rate increases significantly between 5 AM and 9 AM, especially at 9 AM.
  Suggested Action: Investigate scheduled tasks (like backups or crawlers) running during this window. Ensure proper authentication and paths are configured.
- **Failure Clusters on Specific Days (18–19 May):**
  These two days account for 60% of all failed requests.
  Suggested Action: Check for system updates, configuration changes, or deployment activities around that time. Consider staggered deployments and stronger rollback plans during high-traffic periods.

3. Address Security and Traffic Anomalies

- **High Volume from a Single IP (66.249.73.135):**

  This IP made an unusually high number of GET requests (482). It could be a search engine bot or an aggressive crawler.

  Suggested Action: Verify the user-agent. If it's a known and trusted bot (like Googlebot), allow it. Otherwise, consider rate limiting or requiring CAPTCHA verification.

- **Unusual Client-Side Errors (403 & 416):**

  A small number of 403 and 416 errors could point to probing or misconfigured clients.

  Suggested Action: Log full request details for these events. Review firewall or web application firewall (WAF) rules to block or limit suspicious behavior.

4. Improve System Performance and Resilience

- **Use a CDN for Static Content:**

  Peak traffic occurs in the afternoon hours (around 14:00–16:00), which could impact performance.

  Suggested Action: Serve static assets like images, stylesheets, and scripts via a content delivery network (CDN) to reduce load on the main server.

- **Implement Load Balancing and Autoscaling:**

  As traffic rises, ensure the system can scale horizontally by adding more server instances during high-load periods.

- **Enhance Real-Time Monitoring:**

  Monitor response codes by hour and day to catch sudden changes in failure rates.

  Suggested Action: Set up dashboards using tools like Grafana or Datadog and configure alerts for unusual patterns (e.g., failure rate >1%).

- **Design Custom Error Pages:**

  Default error pages can frustrate users.

  Suggested Action: Replace them with user-friendly pages that include helpful navigation options or search, improving the overall experience.