Apache Log File Analysis

Name: Rana Ashraf ***ID:2205019***

OVERVIEW & PURPOSE

This report presents a detailed analysis of web server activity based on Apache logs collected over **4 days** (from **17 May 2015** to **20 May 2015**). The purpose is to understand user behavior, traffic patterns, failure trends, and potential security concerns, providing insights to improve system performance and reliability.

1. Request Counts

- **Total Requests:**
  The total number of HTTP requests made to the server was **10,000**.

- **GET Requests:**
  A vast majority of these were GET requests (**9,952**). This suggests that most users or bots were retrieving resources rather than submitting data.

- **POST Requests:**
  Only **5 POST requests** were recorded. This is a very small fraction, which could indicate that form submissions or API calls are either minimal or non-existent in the current traffic.

- **Other Methods:**
  There were **43 requests** using other HTTP methods (likely HEAD, OPTIONS, or others). This could come from bots probing capabilities or specific diagnostic tools.

2. Unique IP Addresses

Examining unique visitors by their IP helps in understanding how distributed your traffic is.

- **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, and automated scripts.

- **GET/POST by IP:**
  Each IP's usage pattern was broken down to show how many GET and POST requests were made.
  For example:

    - IP **100.43.83.137** made 84 GETs.

    - IP **66.249.73.135** made 482 GETs and was the most active.

3. Failure Requests (4xx/5xx)

Failures are a critical metric for identifying problems in server configuration, application logic, or broken links.

- **Client Errors (4xx):**
  There were **217 client-side errors**, mostly 404 Not Found. These often indicate broken or outdated links on the site or on referring pages.

- **Server Errors (5xx):**
  Only **3 server errors** (e.g., 500 Internal Server Error) were recorded. These are fewer in number but can signal issues with backend logic or server overload.

- **Total Failure Requests: 220**

    - **Failure Rate: 2.20%** of all traffic resulted in some form of error.

4. Top User (Most Active IP)

1. **IP:** `66.249.73.135`

2. **Total Requests:** **482 GETs**
   This IP address is likely a Googlebot or another search engine crawler. Its high activity rate can be considered normal if the server is open to indexing.

5. Daily Request Averages

- **Analysis Window:** **4 days (17–20 May 2015)**

- **Average Requests per Day:** **2,500**

6. Failure Analysis

Failures were not distributed evenly across the days:

Date	Number of Failures
19 May 2015	66
18 May 2015	66
20 May 2015	58
17 May 2015	30

Peak Failure Day: 18–19 May (both with 66 failures)

7. Request Patterns by Hour

A clear daily traffic pattern emerged:

- Peak Hours: **14:00 – 16:00**
    - Around **130** requests/hour
- Low Traffic: **00:00 – 09:00**
    - Minimal to no requests

This suggests that the site is accessed during standard working hours, possibly indicating its usage in business or academic contexts.

8. Request Trends

- Most Active Day: **19 May 2015** with **2,896** requests
- Traffic increased sharply from **17 May** to **18 May** (**+1,261** requests)

Date	Requests	Difference from Previous
17 May	1,632	—
18 May	2,893	+1,261
19 May	2,896	+3
20 May	2,579	–317

9. Status Code Breakdown

Status Code	Count	Meaning
200	9,126	OK
304	445	Not Modified (cached)
404	213	Not Found
301	164	Moved Permanently
206	45	Partial Content (streaming)
500	3	Server Error
416	2	Range Not Satisfiable
403	2	Forbidden

10. Most Active Users by Method

- Most GETs: IP `66.249.73.135` with **482 GET** requests
- Most POSTs: IP `78.173.140.106` with only **3 POST** requests

POST usage is minimal, reinforcing the idea that this is a content-centric site rather than a form- or API-heavy service.

11. Patterns in Failure Requests

By Hour:

- **Most failures occurred around 09:00, 05:00, and 06:00.**

By Day:

- **18 and 19 May saw significantly higher failure rates than the rest.**

RECOMMENDATIONS

Based on the above findings, here are actionable suggestions:

1. Reduce Failures

1. **Eliminate broken links (404s)**
    - **213 of 220 failures are “404 Not Found” .**
    - **Action: Audit your most-requested missing URIs; restore those resources, correct internal links, or put in 301 redirects to valid pages.**
2. **Handle range requests correctly (416s)**
    - **Two “416 Range Not Satisfiable” errors indicate mis-handled partial-content requests .**
    - **Action: Ensure your server either supports byte-range responses properly (for video/audio streaming) or strips unsupported Range headers and returns full content (200).**
3. **Harden against server errors (5xx)**
    - **Although only 3 total, any 500s point to unhandled exceptions .**
    - **Action: Add try/catch around critical endpoints, improve logging of stack traces, and configure alerts when any 5xx occurs.**

2. Focus on High-Attention Days & Times

1. **Early-morning failure spike (05:00–09:00)**
    - **Peak failure hour is 09:00 (18 failures), with 05:00 (15) and 06:00 (14) also high .**
    - **Action: Review cron jobs, health checks, or crawler schedules running in that window; correct mis-configured paths or credentials.**
2. **Critical failure days (18–19 May 2015)**
    - **Both days saw 66 failures each—30% of all failures .**
    - **Action: Correlate with deployments or configuration changes made just before 18 May; consider canary releases or staged roll-outs on high-traffic days.**

3. Address Security Concerns & Anomalies

1. **High-volume IPs**
    - **IP 66.249.73.135 alone made 482 requests (4.8% of traffic) .**
    - **Action: Verify its user-agent (likely Googlebot). If legitimate, allow; if not, apply rate-limiting or CAPTCHAs.**
2. **Repeated client errors (403/416)**
    - **A handful of 403s and 416s may signal probing.**
    - **Action: Log these occurrences with full request headers and consider tightening firewall/WAF rules on endpoints returning 403/416.**

4. System & Performance Improvements

1. **Offload static content to a CDN**
    - **Peak traffic hours (14:00–16:00, ~130 req/hr) strain origin servers .**
    - **Action: Move images, CSS/JS, and other static assets to a CDN to reduce latency and origin-server load.**
2. **Autoscaling & load-balancing**
    - **Ensure additional application instances spin up automatically during midday peaks.**
3. **Enhanced real-time monitoring**
    - **Track status-code distributions by hour/day with alerts on >1% failure rate.**
    - **Visualize trends in a dashboard (Grafana/Datadog) for rapid anomaly detection.**
4. **User-friendly error pages**
    - **Replace generic 404/500 responses with branded pages offering navigation/search to reduce user frustration.**

Conclusion

Over four days, the server handled 10,000 requests with a 2.2% error rate driven by broken links and a morning failure spike. Traffic peaks at 14:00–16:00, and a handful of IPs generate disproportionate load. By fixing 404s, properly handling ranges, hardening error paths, tuning early-morning jobs, scaling for midday peaks, and tightening security on noisy clients, we can reduce failures, improve performance, and strengthen reliability.