SERVER LOGS

Access and Error logs

COS216
AVINASH SINGH
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF PRETORIA

LOGS – OVERVIEW

- Logs are simple text files containing info for the developer and system admin
- The are typically ordered sequentially according to a timestamp
  - Oldest entries at the start of the file
  - Newest entries at the end of the file
- Webservers have different kinds of logs
  - We focus on Apache logs
    - - Access Logs
    - - Error Logs

LOGS – OVERVIEW

- By default, Apache keeps track of two kinds of logs
- Error Log
  - Errors and warnings are thrown by the web server
- Access Log
  - Keeps track of all the requests made to the server

ERROR LOGS – OVERVIEW

- Keeps track of script and other webserver errors
  - Errors and warnings thrown by PHP (or other languages)
  - Errors manually printed using the PHP error_log() function
  - Errors when non-existing files are being accessed
- Typically error log file names
  - error.log
  - error_log

ERROR LOGS – FORMAT

- The format of the error log file

[date information] [error severity] [process id] [client ip] message

- Note that the **process ID** and **client IP** is often not recorded

ERROR LOGS – EXAMPLE

[Mon Apr 20 14:44:58 2015] [error] [client 137.215.33.103]

PHP Fatal error: Call to a member function fetch_assoc( ) on a non-object in /home/cs/students/u.../Assignment4/profiles.php on line 43, referer: http://wheatley.cs.up.ac.za/u.../Assignment4/login.php

ERROR LOGS – EXAMPLE


u_ex150803.log - Notepad

File Edit Format View Help

```
date time s-ip cs-method cs-uri-stem cs-uri-query s-port cs-username c-ip cs(User-
Agent) sc-status sc-substatus sc-win32-status time-taken
2015-08-03 12:40:57 209.133.7.95 GET /course-eligibility.asp - 80 - 115.118.114.159
Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Ubuntu
+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0 1234
2015-08-03 12:40:58 209.133.7.95 GET /css/font-awesome.min.css - 80 -
115.118.114.159 Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like
+Gecko)+Ubuntu+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0
578
2015-08-03 12:40:58 209.133.7.95 GET /images/ftrlogo.png - 80 - 115.118.114.159
Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Ubuntu
+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0 312
2015-08-03 12:40:58 209.133.7.95 GET /css/styles.css - 80 - 115.118.114.159
Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Ubuntu
+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0 609
2015-08-03 12:40:58 209.133.7.95 GET /js/modernizr.custom.86080.js - 80 -
115.118.114.159 Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like
+Gecko)+Ubuntu+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0
281
2015-08-03 12:40:58 209.133.7.95 GET /css/bootstrap.min.css - 80 - 115.118.114.159
Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Ubuntu
+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0 1171
2015-08-03 12:40:58 209.133.7.95 GET /js/bootstrap.min.js - 80 - 115.118.114.159
Mozilla/5.0+(X11;+Linux+x86_64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Ubuntu
+Chromium/37.0.2062.120+Chrome/37.0.2062.120+Safari/537.36 200 0 0 593
```

ERROR LOGS – SEVERITY

- Possible error_severity values, in order of declining severity

Level	Description	Example
emerg	Emergency, system is unusable	"Child cannot open lock le. Exiting"
alert	Action must be taken immediately	"getpwuid: couldn't determine user name from uid"
crit	Critical Conditions	"socket: Failed to get a socket, exiting child"
error	Error conditions	"Premature end of script headers"
warn	Warning conditions	"child process 1234 did not exit, sending another SIGHUP"
notice	Normal but significant condition	"httpd: caught SIGBUS, attempting to dump core in ..."
info	Informational	"Server seems busy, (you may need to increase StartServers, or Min/MaxSpareServers)..."
debug	Debug-level messages	"Opening config file ..."

ACCESS LOGS – OVERVIEW

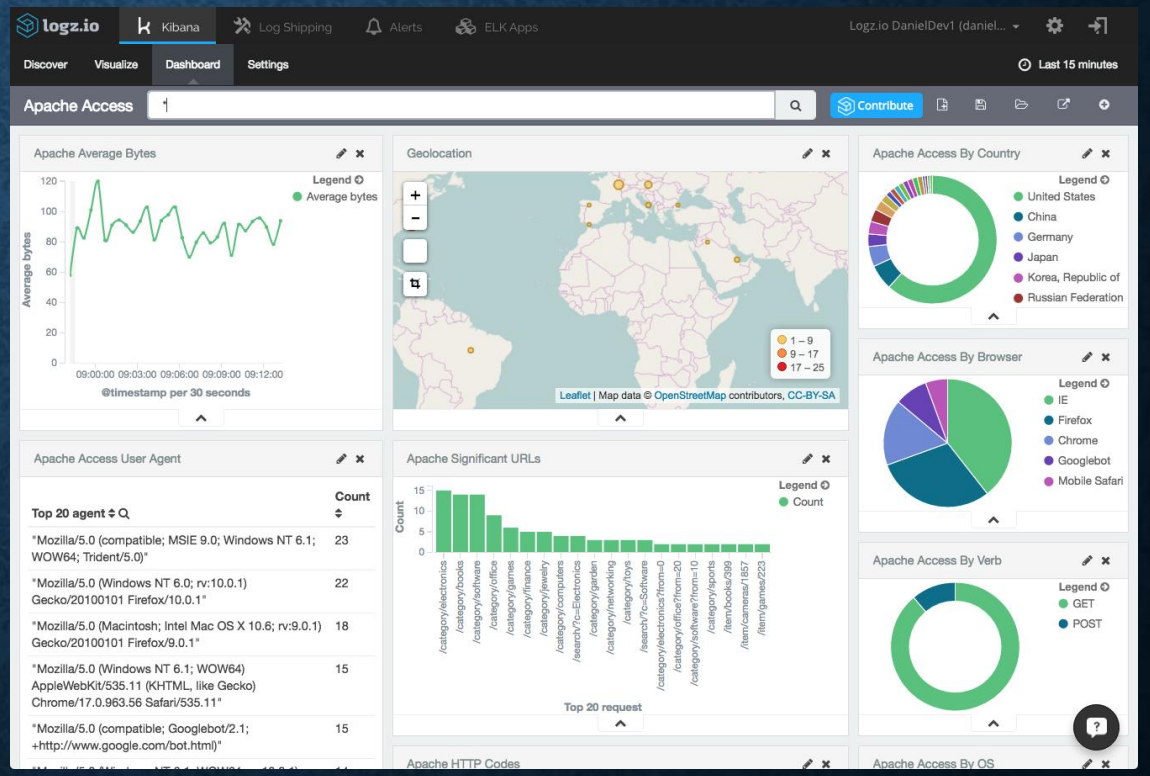
- Access logs store the info of all requests made to the server
- Includes info of the client
  - Client's IP address and browser details
- Includes info of the request
  - Script details, HTTP methods and codes
- Written in the Common Log Format (CLF)
  - Allows developers to write code to interpret the logs
- Typically error log file names
  - access.log
  - access_log

ACCESS LOGS – OVERVIEW

- Access logs store the IP address of each user making a request to the server
- Using free tools, the geographical location of the user can be pinpointed
  - Accurate to a few kilometres
  - In large cities, the accuracy can be a few hundred meters
- Some countries have restrictions on keeping track of user data
  - Forces companies to keep access logs for a certain period of time
  - Forces companies to delete access logs after a certain period of time
  - The EU has laws to anonymize IP addresses if they are stored for longer periods of time
    - The last octet in an IP address is replace with a 0
    - Example: 41.174.52.0

ACCESS LOGS – TOOLS

- Access logs are used by various 3rd-paty tools
- Provides the website admin with useful info about the users



ACCESS LOGS – FORMAT

- The format of the access log file

[client_ip] [client_identity] [user_id] [date_information]

[requestline_information] [status_code_to_client] [object_size_to_client]

ACCESS LOGS – FORMAT

- A hyphen (RFC1413) indicates the requested information is not available
- client_ip with value ::1 represents localhost in IPv6
- user_id should not be trusted if the status_code_to_client is 401
  - If the value is a hyphen, then the user has not been authenticated with a password
- date_information is the time of the request
  - Format: dd/Mon/yyyy:hh:mm:ss +|-zzzz
  - Example: 20/Apr/2019:12:22:52 +0200
- requestline_information specifies the HTTP method (GET or POST)
  - Along with the resource the client requested and the protocol used by the client
- status_code_to_client are defined in RFC2616 Section 10

ACCESS LOGS – STATUS

Category	Code	Range	Example
Informational	1xx	100-101	100 Continue, 101 Switching Protocols
Successful	2xx	200-206	200 OK
Redirection	3xx	300-307	301 Moved Permanently, 302 Found, 304 Not Modified
Client Error	4xx	400-417	401 Unauthorized, 403 Forbidden, 404 Not Found
Server Error	5xx	500-505	500 Internal Server Error

ACCESS LOGS – FORMAT

- Redirection, not modified

137.215.37.114 - u... [20/Apr/2015:14:44:54 +0200] "GET /u.../Assignment4/logo.png HTTP/1.1" 304 187 "http://wheatley.cs.up.ac.za/u.../Assignment4/signup.php" "Mozilla/5.0 (X11; Linux x86_64; rv:34.0) Gecko/20100101 Firefox/34.0"

ACCESS LOGS – FORMAT

- Successful, OK

137.215.37.114 - u... [20/Apr/2015:14:34:04 +0200] "GET /u.../ HTTP/1.1" 200 334 "http://wheatley.cs.up.ac.za/u.../Assignment4/login.php" "Mozilla/5.0 (X11; Linux x86_64; rv:34.0) Gecko/20100101 Firefox/34.0"

ACCESS LOGS – TOOLS

- Tools exist to analyse logs, specially access logs
- Tools exist to summarize the logs
- Tools exists to graphically represent the data
  - From where in the world the website was accessed
  - Dates and times when users were on the site
  - The browsers and devices they used
  - Which pages they accessed
  - How long they were on certain pages

ACCESS LOGS – TAIL

- View the last n entries in the log

```
tail –n 10 –f filename
```

- View the first n entries in the log

```
head –n 10 –f filename
```

ACCESS LOGS – AWK

- To extract specific and summarized information, use awk
- Awk has the following command structure

```
condition {actions}
```

- The condition specifies when the actions are to be performed
  - The default condition (blank or null pattern) is performed on every line
  - A condition highlighted by the keywords BEGIN and END will be applied before and after any lines

ACCESS LOGS – AWK

- The awk command

awk ‘{action(s)}' filename <optional piping commands>

ACCESS LOGS – AWK

- List the error severities that have occurred in the error log

```
awk '{print $6}' error.log
```

- $6 accessed the 6th column of each line
  - Comma or tab delimited

ACCESS LOGS – AWK

- Count the occurrences of each of the severities in the file

```
awk '{print $6}' error.log | sort | unique -c | sort
```

ACCESS LOGS – AWK

- Very complex analysis of logs can be done using awk

https://www.the-art-of-web.com/system/logs/