Lab 6: Apache Logging

Introduction: In this lab you will examine the 2 main log files generated by the Apache2 web server **access.log** and **error.log**, then create a custom Apache log using CLF/LogFormat.

```
LogFormat "%h %l %u %t \"%r\" %>s %b" common
```

This will require one server running a deb-based Linux distribution and an Apache server. You should start the server and check that Apache is running. Ensure there is an index page visable on the localhost address from the machines web browser. Create a logbook that demonstrates you have completed each of the tasks here.

Part 1: Access Log

1.1 In Unix-like operating systems, files that change (variable files) are found in **/var**, such as a web server's html and log files. Config files are usually found in:

/etc/{PROGRAM_NAME}/{PROGRAM_NAME}.conf

1.2 Become root and change directory to **/var/log**, list out all of the logs that the system is currently maintaining, there should be an Apache directory, **cd** to there and list that out.

1.3 Use **cat** to look at the contents of the access and the error logs. You should be able to see that the entries show a timestamp and various other information, also the http status code for a successful connection to the web server should be 200.

1.4 Concatenate (**cat**) the access log and then pipe this into the word counter, using the word count command with the *lines* option, count the number of entries in the log each line is the result of one access to a page running on the server.

```
cat /var/log/apache2/access.log | wc -l
```

1.5 Word count is a useful command, read the man page then find out how many *characters* are in the log in total.

1.6 Go to the websites default page in the browser, now hit refresh, now run word count lines again. The number of lines in the log should increase every time the page is refreshed.

Part 2: Error Log

2.1 Look at the error log now, there shouldn't be too much in there, count the lines, there should be less here than the access log.

```
cat /var/log/apache2/error.log
cat /var/log/apache2/error.log | wc -l
```

2.2 You would have to look in here if something was broken, for example and the Apache config test (**apache2ctl configtest**) wasn't giving enough information, here would be a good place to start troubleshooting the error.

Part 3: Custom Log for example.com

3.1 As root make a new directory for a new site example.com, and then make an index page.

```
mkdir -p /var/www/example.com/
rm var/www/html/index.html
vi /var/www/example.com/index.html

<html>
      <head><title>example.com</title></head>

      <body><h1> Welcome to Example.Com </h1></body>
</html>
```

Type users and **chown** the file to one of those users, in my case the non-root user is michael.

```
chown -R michael:michael /var/www/example.com/
chmod -R 755 /var/www
```

Copy the default config file to a new example.com.conf file to make a new config file, this is all one line, just doesn't fit on the page.

```
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-
available/example.com.conf
```

3.2 Now edit example.com.conf, first thing to do is remove every line that starts with a #, also delete the ServerAdmin line, there should only be 3 lines left inside the VitrualHost directive now, DocumentRoot, ErrorLog and CustomLog. Now it should look like this:

```
<VirtualHost *:80>
    DocumentRoot /var/www/example.com/
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

3.3 Enable the new site and disable the default site with these commands, then restart.

```
a2ensite example.com.conf
a2dissite 000-default.conf
systemctl restart apache2
```

3.3 Go to the localhost ip of the server in it's web browser, and check that example.com is up, and showing as created in part 3.1.

3.4 **cd** to **/var/log/apache2/** and list out the contents, there should be no changes. Edit **/etc/apache2/sites-available/example.com.conf**, change the custom log name from to generate a new log file.

```
CustomLog ${APACHE_LOG_DIR}/example_access.log combined
```

3.5 Restart apache2, then go to the browser, refresh the page several times, then go to the log directory and you should see the new custom log.

Part 4: The Magic Log

4.1 In this section we create a new LogFormat called magic log, for the purposes of demonstration we will only put 1 thing in this log to start off then add to it.

4.2 We are going to create this log in apache's global configuration file, then we can use this on all/or any virtual hosts by referring to the log's name in their VirtualHost directives, edit the global configuration file:

```
vi /etc/apache2/apache2.conf
```

4.3 In normal mode in vi use :/LogFormat to search for LogFormat, if you press n it will take you to the next match, if you press N it will take you to the previous match, press i to edit here. That saves looking through 200 lines to find the log section.

4.3.2 Add a new line under the last Logformat with common log format syntax (CLF), this will create a log with only the ip address (host/h) and the time (t). Magic is the nickname we are giving this so we can refer to it in the VirtualHost file.

```
LogFormat “%h %t” magic
```

4.4 Edit the example site file to refer to the new log format, change the custom log line. We already created a new example_access log, now we need to specify which log format to use.

```
vi/etc/apache2/sites-available/example.com.conf
CustomLog ${APACHE_LOG_DIR}/example_access.log magic
```

4.5 Now cd to the access logs in **/var/log/apache2/** type **ls**, then **cat** the example log, then go to the browser and refresh 5 or 6 times then cat the log again. Do you see the new magic log in operation?

4.6 Now for SEO, Marketing, Sales, Reports, Testing, Analysis and a load of other things these logs can be extremely useful. You want as much information as possible so you can see when pages were busy what browsers folk were using, geolocate ip's... Return to the global configuration file and make the magic log look like this:

```
LogFormat "%h %l %u %t \"%r\" %>s %b %D \"%{Referer}i\" \"%{User-
Agent}i\"" magic
```

4.7 Restart the server, now refresh the page another 5 times or so and then **cat** the example log, there should be lots of things in there.

Part 5: The Magic Log

Upload evidence of this to the Moodle in PDF format.