**AWK TEXT ANALYSIS TOOLS CHEAT SHEET**

**Glossary:**

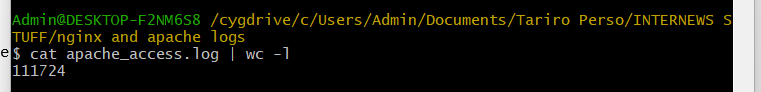
|  |  |  |
| --- | --- | --- |
| Name | Symbol | What it does |
| Pipe | | | connecting the output of one command to the input of another command  This allows you to chain multiple commands together in a single line to perform more complex operations efficiently |

**Example 1: Identify the total number of requests recorded in the access log**

**Command Syntax:**

cat apache\_access.log | wc -l

**Output:**

****

**What it does:**

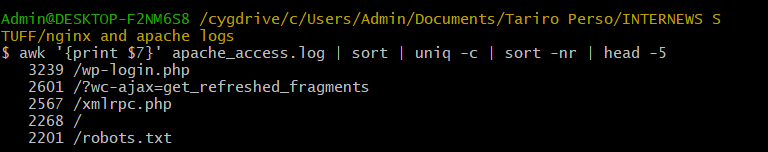
* cat is a command that is used to concatenate and display the contents of files. In this case, it displays the contents of the file
* The vertical bar **|** is a pipe character that is used to redirect the output of one command as input to another command
* wc is a command that is used to count words, lines, and characters in a file. In this command, **wc -1** counts the number of lines in the input it receives from the cat command

**Example 2: Determine the most frequently requested URLs**

**Command Syntax:**

awk '{print $7}' apache\_access.log | sort | uniq -c | sort -nr | head -5

**Output:**

****

**What this command does:**

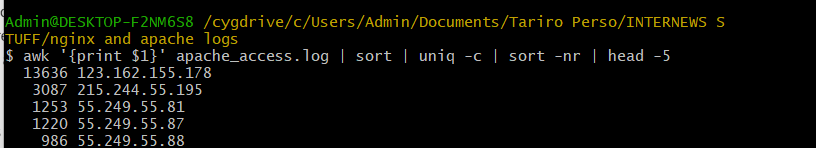
* This awk command will print the seventh column from each line of the log then pipe the output of the previous awk command into the sort command.
* Sort is used to sort the lines of text alphabetically or numerically.
* By default, it sorts in ascending order.
* After sorting the output with sort, the uniq -c command is used to count the occurrences of each unique line in the sorted output.
* The sort -nr command is used to sort the output numerically (-n) in reverse order (-r). This means that the lines are sorted based on their numerical values, with the highest values appearing first.
* The head -5 command is used to display the first 5 lines of the input

**Example 3: Find out the top 5 IP addresses making requests to the server**

**Command Syntax:**

awk '{print $1}' apache\_access.log | sort | uniq -c | sort -nr | head -5

**Output:**



**Remember:**

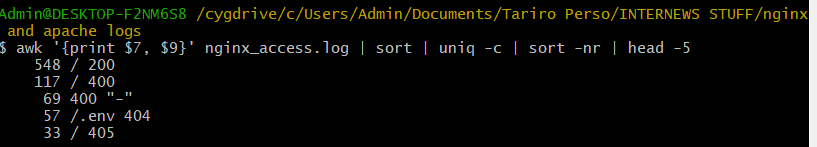
* Sort is used to sort the lines of text alphabetically or numerically.
* By default, it sorts in ascending order.
* After sorting the output with sort, the uniq -c command is used to count the occurrences of each unique line in the sorted output.
* The sort -nr command is used to sort the output numerically (-n) in reverse order (-r). This means that the lines are sorted based on their numerical values, with the highest values appearing first.
* The head -5 command is used to display the first 5 lines of the input

**Example 4: Identify the most requested URLs and their corresponding status codes**

**Command Syntax:**

awk '{print $7, $9}' nginx\_access.log | sort | uniq -c | sort -nr | head -5

**Output:**

****

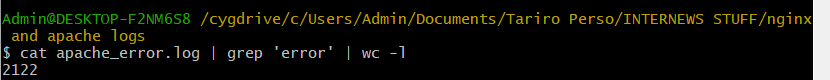
**EXAMPLES USING APACHE ERROR LOGS**

**Example 1: Count the total number of error entries in the log**

**Command Syntax:**

cat apache\_error.log | grep 'error' | wc -l

**Output:**

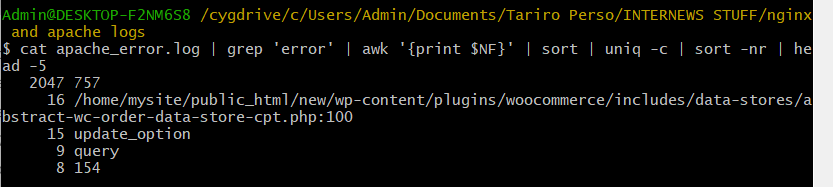
****

**Example 2: Identify the most common types of errors:**

**Command Syntax:**

cat apache\_error.log | grep 'error' | awk '{print $NF}' | sort | uniq -c | sort -nr | head -5

**Output:**

****

**A little bit more info:**

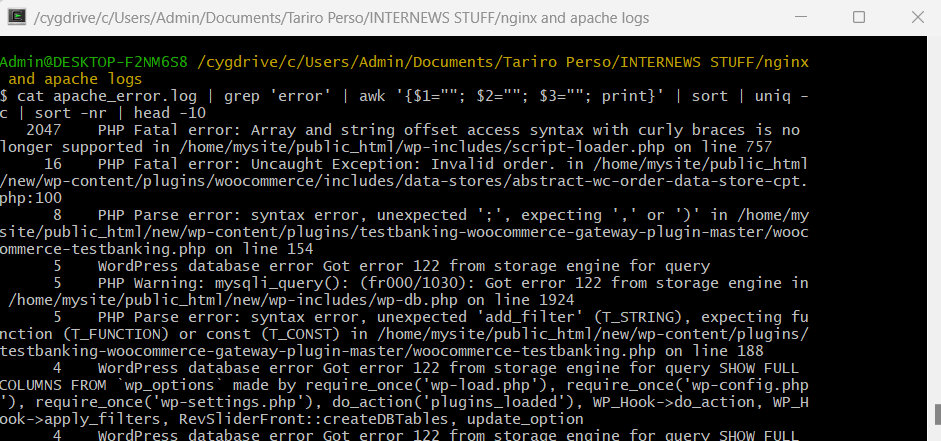
* awk '{print $NF}' reads each line of input data, splits it into fields (separated by whitespace by default), and then prints the value of the last field from each line
* The number at the beginning of each line shows how many times a particular error occurred in the log. In this case, “2047” means that the error with the last field “757” occurred 2047 times.
* The last field represents different things in each line. It could be a file path, a specific action, or some other identifier related to the error.
* For instance, “757” or “154” could be error codes or unique identifiers, while “/home/mysite/public\_html/new/wp-content/plugins/woocommerce/includes/data-stores/abstract-wc-order-data-store-cpt.php:100” could be a file path and line number where the error occurred

**Example 3: Investigate any recurring error patterns so that you can propose potential solutions**

**Command Syntax:**

cat apache\_error.log | grep 'error' | awk '{$1=""; $2=""; $3=""; print}' | sort | uniq -c | sort -nr | head -10

**Output:**



**A little bit more info:**

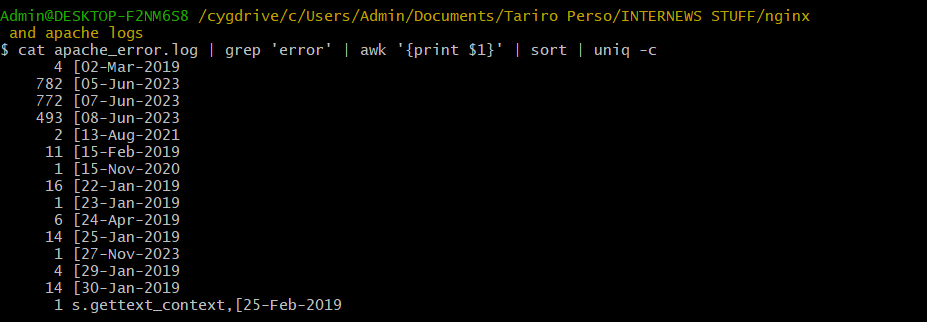
* {$1=""; $2=""; $3="";}: This part of the awk command sets the first three fields (date, time, and timezone information) to empty strings

**Example 4: Analyze the distribution of errors by date or time**

**Command Syntax:**

cat apache\_error.log | grep 'error' | awk '{print $1}' | sort | uniq -c

**Output:**

****