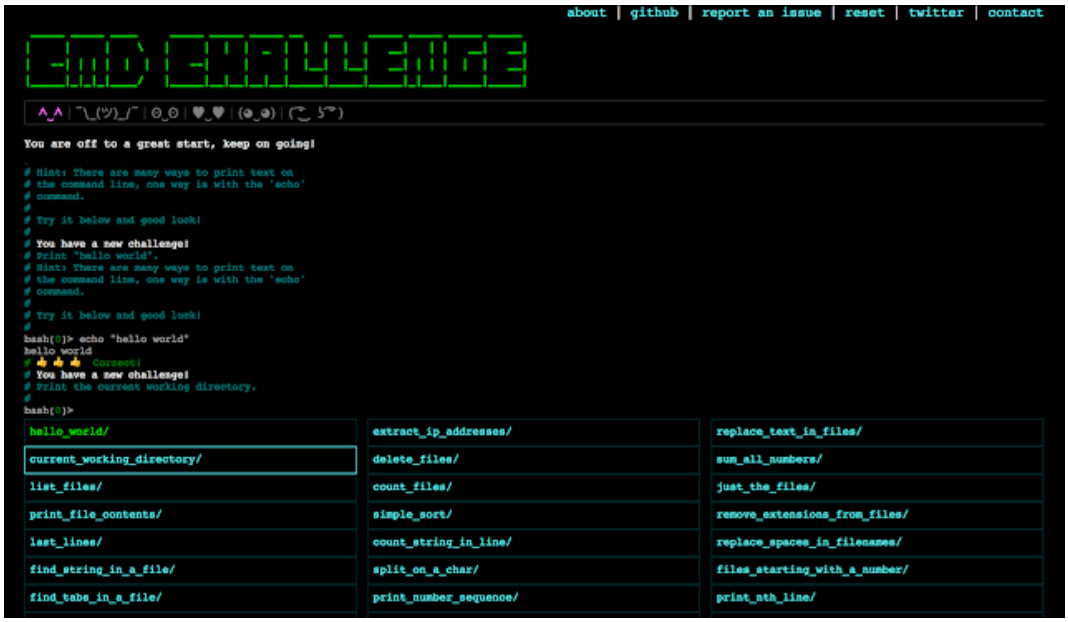
Commandline Challenge (bash) writeup

POSTED BY VICENTE MOTOS ON FRIDAY, FEBRUARY 10, 2017 TAGS: BASH , CHALLENGES , WRITEUPS

I recently read one entry on the blog of our fellow Cyberhades about **Commandline Challenge,** a platform with different challenges that will test our skill at the hands of a console bash. It is also an open source project that you can obtain at Github , where you will even find instructions to create your own.

If you want to start "playing" you just have to visit the URL https://cmdchallenge.com/and try to do each of the exercises:



As you will see it starts from the basics and then the difficulty level will *crescendo* . Here you have the complete write up just in case you get stuck in any:

SOLUTIONARY / WRITE-UP Are you up for the command line challenge? Solve the tasks printed below in a single line of bash. **hello_world** /

—

```
# Print "hello world".
# Hint: There are many ways to print text on
# the command line, one way is with the 'echo'
# command.
#
# Try it below and good luck!

#
bash (☠)> echo "hello world"
hello world
```

```
# 👍 👍 👍 Correct!
```

current_working_directory /

```
# You have a new challenge!
#  Print the current working directory.
#
bash (0)>  pwd
/ var / challenges / current_working_directory
# 👍 👍 👍 Correct!
```

list_files /

```
# You have a new challenge!
# List names of all the files in the current
# directory, one file per line.
#
bash (0)>  ls
README
# 👍 👍 👍 Correct!
```

print_file_contents /

```
# You have a new challenge!
# There is a file named "access.log" in the
# current directory. Print the contents.
#
bash (0)>  cat access.log
163.56.115.58 - - [09 / Jan / 2017: 22: 29: 57 +0100] "GET / posts / 2 / display HTTP / 1.0"
200 3240
75.113.188.234 - - [ 09 / Jan / 2017: 22: 30: 43 +0100] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 200 1116
69.16.40.148 - - [09 / Jan / 2017: 22: 34: 33 +0100] "GET / pages / create HTTP / 1.0" 500
3471
225.219.54.140 - - [09 / Jan / 2017: 22: 35: 30 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0" 500 2477
207.243. 19.2 - - [09 / Jan / 2017: 22: 38: 03 +0100] "GET / bar / create HTTP / 1.0" 200
1116
199.37.62.156 - - [09 / Jan / 2017: 22: 42: 18 +0100] "GET / posts / 1 / HTTP / 1 display.
55.74.240.123 - - [09 / Jan / 2017: 22: 44: 25 +0100] "POST / posts / 1 / HTTP display /
1.0" 200 3471
251.111.109.143 - - [09 / Jan / 2017: 22: 49: 02 +0100] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 200 2477
101.163.230.250 - - [09 / Jan / 2017: 22: 52: 31 +0100] "DELETE / posts / 2 / HTTP display /
1.0 "404 2477
200.19.168.148 - - [09 / Jan / 2017: 22: 57: 11 +0100]" GET / posts / foo? AppID = xxxx HTTP
```

```
/ 1.0 "200 3471
# 👍 👍 👍 Correct!
```

last_lines /

```
# You have a new challenge!
# Print the last 5 lines of "access.log".
#
bash (0)>  tail -n 5 access.log
199.37.62.156 - - [09 / Jan / 2017: 22: 42: 18 +0100] "GET / posts / 1 / HTTP display / 1.0"
200 2477
55.74.240.123 - - [09 / Jan / 2017: 22: 44: 25 +0100] "POST / posts / 1 / HTTP display /
1.0" 200 3471
251.111.109.143 - - [09 / Jan / 2017: 22: 49: 02 +0100 ] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 200 2477
101.163.230.250 - - [09 / Jan / 2017: 22: 52: 31 +0100] "DELETE / posts / 2 / display HTTP /
1.0" 404 2477
200.19.168.148 - - [09 / Jan / 2017: 22: 57: 11 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0" 200 3471
# 👍 👍 👍 Correct!
```

find_string_in_a_file /

```
# You have a new challenge!
# There is a file named "access.log" in the
# current working directory. Print all lines
# in this file that contains the string "GET".
#
bash (0)>  cat access.log | grep GET
163.56.115.58 - - [09 / Jan / 2017: 22: 29: 57 +0100] "GET / posts / 2 / display HTTP / 1.0"
200 3240
75.113.188.234 - - [09 / Jan / 2017: 22: 30:43 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0" 200 1116
69.16.40.148 - - [09 / Jan / 2017: 22: 34: 33 +0100] "GET / pages / create HTTP / 1.0 "500
3471
225.219.54.140 - - [09 / Jan / 2017: 22: 35: 30 +0100]" GET / posts / foo? AppID = xxxx HTTP
/ 1.0 "500 2477
207.243.19.2 - - [09 / Jan / 2017: 22: 38: 03 +0100] "GET / bar / create HTTP / 1.0" 200
1116
199.37.62.156 - - [09 / Jan / 2017: 22: 42: 18 +0100] "GET / posts / 1 / HTTP display / 1.0"
200 2477
251.111.109.143 - - [09 / Jan / 2017: 22: 49: 02 +0100] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 200 2477
200.19.168.148 - - [09 / Jan / 2017: 22: 57: 11 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0 "200 3471
# 👍 👍 👍 Correct!
```

find_tabs_in_a_file /

```
# You have a new challenge!
# How many lines contain tab characters in
# the file named "file-with-tabs.txt" in the
# current directory.
#
bash (0)> cat file-with-tabs.txt | grep $ '\ t' | wc -l
3
# 👍 👍 👍 Correct!
```

search_for_files_containing_string /

```
# You have a new challenge!
# Print all files in the current directory,
# one per line (not the path, just the filename)
# that contain the string "500".
#
bash (0)> find. | grep -rl 500
access.log
access.log.1
README
# 👍 👍 👍 Correct!
```

search_for_files_by_extension /

```
# You have a new challenge!
# Print the relative file paths, one path
# per line for all filenames that start with
# "access.log" in the current directory.
#
bash (0)> find. -name "access.log *"
./access.log.2
./access.log
./access.log.1
# 👍 👍 👍 Correct!
```

search_for_string_in_files_recursive /

```
# You have a new challenge!
# Print all matching lines (without the filename
# or the file path) in all files under the current
# directory that start with "access.log" that
# contain the string "500". Note that there are no
# files named "access.log" in the current directory,
# you will need to search recursively.
#
bash (0)> find. -name "access.log *" | xargs grep -h 500
```

```
69.16.40.148 - - [09 / Jan / 2017: 22: 34: 33 +0100] "GET / pages / create HTTP / 1.0" 500
3471
225.219.54.140 - - [09 / Jan / 2017: 22 : 35: 30 +0100] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 500 2477
2.71.250.27 - - [09 / Jan / 2017: 22: 41: 26 +0100] "GET / pages / create HTTP /1.0 "
```

extract_ip_addresses /

```
# You have a new challenge!
# Extract all IP addreses from files
# that start with "access.log" printing one
# IP address per line.
#
bash (0)> find. -name "access.log *" | xargs grep -Eo '^ [^] +'
./var/log/httpd/access.log:163.56.115.58
./var/log/httpd/access.log:75.113.188.234
./var/log/httpd/access .log: 69.16.40.148
./var/log/httpd/access.log:225.219.54.140
./var/log/httpd/access.log:207.243.19.2
./var/log/httpd/access.log:199.37. 62.156
./var/log/httpd/access.log:55.74.240.123
./var/log/httpd/access.log:251.111.109.143
./var/log/httpd/access.log:101.163.230.250
./var/ log / httpd / access.log: 200.19.168.148
./var/log/httpd/access.log.1:108.68.174.15
./var/log/httpd/access.log.1:17.2.20.139
./var/log/httpd/access.log.1:28.151. 137.59
./var/log/httpd/access.log.1:199.150.241.179
./var/log/httpd/access.log.1:2.71.250.27
./var/log/httpd/access.log.1:17.137 .186.194
./var/log/httpd/access.log.1:151.84.119.34
./var/log/httpd/access.log.1:4.180.204.195
./var/log/httpd/access.log.1: 9.230.96.54
./var/log/httpd/access.log.1:157.143.233.21
# 👍 👍 👍 Correct!
```

delete_files /

```
# You have a new challenge!
# Delete all of the files in this challenge
# directory including all subdirectories and
# their contents.
#
bash (0)> find. -delete
# 👍 👍 👍 Correct!
```

count_files /

```
# You have a new challenge!
# Count the number of files in the current
# working directory. Print the number of
# files as a single integer.
#
bash (0)> ls | wc -l
2
# 👍 👍 👍 Correct!
```

simple_sort /

```
# You have a new challenge!
# Print the contents of access.log
# sorted.
#
bash (0)> sort access.log
101.163.230.250 - - [09 / Jan / 2017: 22: 52: 31 +0100] "DELETE / posts / 2 / HTTP display /
1.0" 404 2477
163.56.115.58 - - [ 09 / Jan / 2017: 22: 29: 57 +0100] "GET / posts / 2 / HTTP display /
1.0" 200 3240
199.37.62.156 - - [09 / Jan / 2017: 22: 42: 18 +0100] "GET / posts / 1 / HTTP display / 1.0
"200 2477
200.19.168.148 - - [09 / Jan / 2017: 22: 57: 11 +0100]" GET / posts / foo? appID = xxxx HTTP
/ 1.0 "200 3471
207.243. 19.2 - - [09 / Jan / 2017: 22: 38: 03 +0100] "GET / bar / create HTTP / 1.0" 200
1116
225.219.54.140 - - [09 / Jan / 2017: 22: 35: 30 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0" 500 2477
251.111.109.143 - - [09 / Jan / 2017: 22: 49: 02 +0100] "GET / posts / foo? AppID = xxxx
HTTP / 1.0" 200 2477
55.74.240.123 - - [09 / Jan / 2017: 22: 44:25 +0100] "POST / posts / 1 / HTTP display / 1.0"
200 3471
69.16.40.148 - - [09 / Jan / 2017: 22: 34: 33 +0100] "GET / pages / create HTTP / 1.0" 500
3471
75.113.188.234 - - [09 / Jan / 2017: 22: 30: 43 +0100] "GET / posts / foo? AppID = xxxx HTTP
/ 1.0" 200 1116
# 👍 👍 👍 Correct!
```

count_string_in_line /

```
# You have a new challenge!
# Print the number of lines
# in access.log that contain the string
# "GET".
#
bash (0)> grep GET access.log | wc -l
8
# 👍 👍 👍 Correct!
```

split_on_a_char /

```
# You have a new challenge!
# The file split-me.txt contains a list of
# numbers separated by a ';' character.
# Split the numbers on the ';' character,
# one number per line.
#
bash (0)> cat split-me.txt | sed s / \; / \\ n / g
1
2
3
4
5
6
7
8
9
10
# 👍 👍 👍 Correct! print_number_sequence /



# You have a new challenge!
# Print the numbers 1 to 100 separated
# by spaces.
#
bash (0)> echo {1.100}
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89
90 91 92 93 94 95 96 97 98 99 100
# 👍 👍 👍 Correct!
```

remove_files_with_a_dash /

```
# You have a new challenge!
# There are some files in this directory that
# start with a dash in the filename.
# Remove those files.
#
bash (0)> find ./ -iname '- *' -exec rm {} \;
# 👍 👍 👍 Correct!
```

remove_files_with_extension /

```
# You have a new challenge!
# There are files in this challenge with
```

```
# different file extensions.
# Remove all files with the .doc extension
# recursively in the current working directory.
#
bash (0)> find. -name "* .doc" -delete
# 👍 👍 👍 Correct!
```

remove_files_without_extension / `# You have a new challenge! # There are files in this challenge with # different file extensions. #` **`Remove all files without the .txt and .exe extensions # recursively in the current working directory.`**`#  bash (2)> find. -type f! -regex ". * /. * \. \ (txt \ | exe \)" -delete # 👍 👍 👍 Correct!`

replace_text_in_files /

```
# You have a new challenge!
# This challenge has text files (with a .txt extension)
# that contain the phrase "challenges are difficult".
# Delete this phrase recursively from all text files.
# Note that some files are in subdirectories so you will
# need to search for them.
#
bash (1)> find ./ -iname "* .txt" -print0 | xargs -0 sed -i 's / challenges are difficult //
g'
# 👍 👍 👍 Correct!
```

sum_all_numbers /

```
# You have a new challenge!
# The file sum-me.txt has a list of numbers,
# one per line. Print the sum of these numbers.
#
bash (0)> cat sum-me.txt | xargs | sed -e 's / \ / + / g' | bc
42
# 👍 👍 👍 Correct!
```

just_the_files /

```
# You have a new challenge!
# Print all files in the current directory
# recursively without the leading directory path.
```

```
#
bash (0)> find. -type f -printf "% f \ n"
error.doc
corporis.xls
odit.doc
animi.doc
necessitatibus.doc
totam
beatae.flac
readme
libero.xls
# 👍 👍 👍 Correct!
```

remove_extensions_from_files /

```
# You have a new challenge!
# Rename all files removing the extension from
# them in the current directory recursively.
#
bash (0)> find `pwd` -type f -exec bash -c 'mv" $ 1 "" $ {1%. *} "' - '{}' \;
mv: '/ var / challenges / remove_extensions_from_files / totam' and '/ var / challenges /
remove_extensions_from_files / totam' are the same file
mv: '/ var / challenges / remove_extensions_from_files / README' and '/ var / challenges /
remove_extensions_from_files / README' are the same file
# 👍 👍 👍 Correct!
```

replace_spaces_in_filenames /

```
# You have a new challenge!
# The files in this challenge contain spaces.
# List all of the files (filenames only) in the
# current directory but replace all spaces with
# a '.' character.
#
bash (0)> find. -type f -printf "% f \ n" | xargs -0 -I {} echo {} | tr '' '.'
Thomas.Parks
Mr.
James.Lopez
Amy.Anderson
Claudia.Mccormick
Kevin.Price
James.Harper
Lynn.Robinson
Thomas.Washington
Corey.Bird
Robert.Hill
Angel.Saunders
Tamara.Anderson
John.Nguyen
```

```
Luke.Mason Mr..Shawn. Martin
Crystal.Valdez
Jared.Hill.DVM
Sheri.Bishop
Allison.Brown
Christine.Valdez
Briana.Wilson
Mallory.Peterson
Tammy.Galloway
Brad.Michael
Alexis.Stein
Kimberly.Parker
Lori.Macias
Crystal.Dunn
Terri.Young
Robert.Gregory
Karen.Ramirez
Michaela.Hobbs
Olivia.Irwin
Yvonne.Myers
Mrs. .Jade.Clark
Christopher.Miller
Adam.Simpson
Tiffany.Clark
Parker.Gilbert
James.Roberts
Matthew.Romero
Molly.Stevens
Marie.Gutierrez
Carrie.Alexander
Sarah.Hill
Joseph.Hurst
Jorge.Ross
Erica.Richardson
README
Courtney.Miller
Scott.Rice
# 👍 👍 👍 Correct!
```

files_starting_with_a_number /

```
# You have a new challenge!
# There are a mix of files in this directory
# that start with letters and numbers. Print
# the filenames (just the filenames) of all
# files that start with a number recursively
# in the current directory.
#
bash (0)> find. -name '[0-9] *' -type f -printf "% f \ n"
```

```
540Katherine Jones
78Michelle Spencer
757Robert Marquez
293Linda Bennett
335John Joseph
593Brett Martin
04Carrie Alexander
436Teresa Owens
639Charles Ferguson
477Thomas Pierce MD
778Holly Archer
3maxime.mp3
388Andrew Carter
682Terri Jones
42 Robert Hill
511Tammy Welch
402Nancy Henson
48Thomas Allen
974Michael Bowman
25Brandon McDonald
132Rebecca Rubio
670James Jacobs
99blanditiis.avi
737Jeffrey Davis
# 👍 👍 👍 Correct!
# You have a new challenge!
```

print_nth_line /

```
# Print the 25th line of the file faces.txt
#
bash (0)> sed '25q; d' faces.txt
¯ \ _ (ツ) _ / ¯
# 👍 👍 👍 Correct!
```

remove_duplicate_lines /

```
# You have a new challenge!
# Print the file faces.txt, but only print the first instance of each
# duplicate lin e, even if the duplicates don't appear next to each other.
# Note that order matters so don't sort the lines before removing duplicates.
#
bash (0)> awk '! seen [$ 0] ++' faces.txt
( ◕‿◕ )
(^ ‿^)
⊙‿⊙
ಠ _ ಠ
ಠ ‿ ಠ
ಠ ‿ ಠ
( ⊙‿⊙ )
```

```
( ಠ _ ಠ )
¯ \ _ ( ツ ) _ / ¯
( ಠ ‿ ಠ
ಠಠ ‿ ಠ )
( ಠ ‿ ಠ )
٩ ͜◔◔۶
ヽ༼ ຈ ل ͜ ຈ ༽ ﾉ
♥ ‿ ♥
◔̯◔
⊙ ﹏⊙
(¬_¬)
( ; ─ _ ─ )
( ͡ ° ͜ʖ ͡ °)
( ° ͜ʖ °)
¯ \ (° _o) / ¯
( ﾟ ヮ ﾟ )
( ︶︿︶ )
```

```
# 👍 👍 👍   Correct!
```

corrupted_text / # You have a new challenge! # The following excerpt from War and Peace is saved to # the file 'war_and_peace.txt': # # She is betraying us!Russia alone must save Europe. # Our gracious sovereign recognizes his high vocation # and will be true to it. That is the one thing I have # faith in!Our good and wonderful sovereign has to # perform the noblest role on earth, and he is so virtuous # and noble that God will not forsake him. He will fulfill # his vocation and crush the hydra of revolution, which # has become more terrible than ever in the person of this # murderer and villain! #

```
# The file however has been corrupted, there are random '!'
# marks inserted throughout.  Print the original text.
#
bash (0)> <war_and_peace.txt tr -s '!' | sed 's /! \ ([az] \) / \ 1 / g' |sed 's /! \ ([az] \) / \ 1 / g' | sed 's /!\.!/./ g' | sed 's /! / / g'
She is betraying us! Russia alone must save Europe.
Our gracious sovereign recognizes his high vocation
and will be true to it. That is the one thing I have
faith in! Our good and wonderful sovereign has to
perform the noblest role on earth, and he is so virtuous
```

and noble that God will not forsake him. He will fulfill
his vocation and crush the hydra of revolution,

Murderer and Villain!
👍 👍 👍 Correct!

print_common_lines / # You have a new challenge! # access.log.1 and access.log.2 are http server logs. **Print the IP # addresses common to both files** , one per line. # bash (0)> IPS1 = `cat access.log.1 | awk '{print $ 1}' '; IPS2 = `cat access.log.2 | awk '{print $ 1}' '; for ip in $ IPS1; do if [[["$ IPS2" = ~ "$ ip"]]; then echo $
ip; fi; done 108.68.174.15 28.151.137.59 2.71.250.27 17.137.186.194 # 👍 👍 👍 Correct!