



Power Snacks

Challenge information

Are you the very best PowerShell user? Try this challenge to get better acquainted with PowerShell's functionality. You need to come up with commands that result in a specific output. You can check your output by piping the result to the "Check" function. E.g. `Get-Content words | do something | Check`

Note: do not use any `format-` function before piping to the Check function. Also, note that the checker may not understand all sorts of inputs. Try piping your output to the `Out-String` function first, or make sure your output matches more closely to the given example. Lastly, instead of `write-host`, use `write-output` to be able to pipe to the checker.



1> 42

Write a script that writes out all numbers (1 per line) from 1 to 1337, inclusive. However, if the number is divisible by 42, instead, print the string "Life, the universe, and everything". Example excerpt given below: ...4041Life, the universe, and everything43.

Points [25 points]



Solution

- We are supposed to make a Powershell script which will print all the numbers from 1 to 1337 but will print the given text if the number is divisible by 42, after some research I was able to make the script given below
- Usage :- Print-inRange | Check

```
function Print-inRange
{
for ($num = 1 ; $num -le 1337 ; $num++ )
{if (($num % 42) -eq 0) {"Life, the universe, and everything"} else {"$num"}}
}
```

- Flag 1 :- CTF{using_your_powers_for_powershell}
- Reference Link :- https://www.reddit.com/r/PowerShell/comments/frifjt/writing_a_script_to_show_a_number_as_odd_or_even/

2> Scrabble

You're playing scrabble with your friends. You have the letters "iydhalao". Which are the words you can form? First, sort them by increasing size, then alphabetically. Only include words of two letters and more. Make use of the dictionary file "dictionary" in /workdir. Example excerpt is given below for different letters

```
....
pad
pea
aped
deaf
....
```

Solution

```
($($Select-String -Path "dictionary" "^[holiday']{2,8}$" -CaseSensitive | Select-String -NotMatch -Pattern "h.*h|o.*o|l.*l|i.*i|d.*d|a.*a|y.
```

Flag:-CTF{using_your_holidays_for_learning_powershell}

3> TSV

In the tab-separated file "passwords.tsv" you get an overview of often-used passwords. Can you give us an overview of the number of passwords per category? Sort the result by descending count. Example except given below.

Name	Count
------	-------

...	..
nerdy	30
animal	19
...

Solution

```
Import-Csv -Delimiter "t" -Path passowrds.tsv | Group-Object category | sort Count -Descending | Select-Object Name, Count | check
```

Flag:- CTF{powerful_password_filtering}

4> Names

Given the passwords file, supply a list of passwords from the 'names' category ordered first by ascending passwords length, then alphabetically. Example excerpt given below.

.....
scott
steve
albert
alexis
.....

Solution

```
Import-Csv -Delimiter "t" -Path passowrds.tsv | Where-Object ($_.category -eq "names") | Select-Object password | sort {$_.password.length
```

Flag :- CTF {im_a_powershell_pro}