

## Modulo: 2

### Esercitazione Kali - Imparare i comandi

L'esercitazione, sotto forma di avventura, ci guida ed invoglia ad imparare i comandi di Linux per procedere con le varie missioni.

Il 'gioco' inizia e nelle prime missioni ci viene chiesto di spostarci da una directory all'altra per completare delle missioni e pian piano dovremo affrontare "mostri" sempre più difficili.

Per completare le prime missioni necessitiamo dei seguenti comandi:

**cd:** per spostarci fra le varie directory

**ls:** per verificare il contenuto della cartella in cui ci troviamo

**pwd:** ci mostra dove ci troviamo attualmente ed il percorso che abbiamo fatto per arrivarci.

```
[mission 2] $ cd
[mission 2] $ pwd
/home/kali/gameshell/World/nicola/scuola
[mission 2] $ ls
Castle Forest Garden Mountain Stall
[mission 2] $ cd Castle
[mission 2] $ ls
Cellar Great_hall Main_building Main_tower Observatory
[mission 2] $ cd Cellar
[mission 2] $ ls
barrel_of_apples
[mission 2] $ gsh check
```

Congratulations, mission 2 has been successfully completed!

```
[mission 3] $ ls
Castle Forest Garden Mountain Stall
[mission 3] $ cd
[mission 3] $ cd Castle/Main_building/Throne_room
[mission 3] $ gsh check
```

Congratulations, mission 3 has been successfully completed!

La storia va avanti e ci ritroveremo a realizzare directory, sotto directory, spostare i file e rimuoverli.

Per completare queste missioni utilizzeremo:

**mkdir:** crea delle directory vuote

**rm:** rimuove una directory, aggiungendo il **-r** eliminiamo anche i file contenuti al suo interno

**mv**: utilizzato per spostare i file da una directory ad un'altra.

```
~/Forest
[mission 4] $ mkdir Hut/nti/nicola/scuola)
~/Forest
[mission 4] $ cd Hut
~/Forest/Hut
[mission 4] $ mkdir Chest
```

```
~/Forest/Hut/nti/nicola/scuola)
[mission 5] $ cd or directory: ../nicola/studenti
~/Castle/Cellar (~:/studenti/nicola/scuola)
[mission 5] $ cd Castle/Cellar
~/Castle/Cellar (~:/studenti/nicola)
[mission 5] $ ls
barrel_of_apples  bat_1  bat_2  spider_1  spider_2  spider_3
~/Castle/Cellar (~:/studenti/nicola/scuola)
[mission 5] $ rm spider_1 spider_2 spider_3
~/Castle/Cellar (~:/studenti/nicola/scuola)
[mission 5] $ ls
barrel_of_apples  bat_1  bat_2  ../studenti
~/Castle/Cellar (~:/studenti/nicola/scuola)
[mission 5] $ gsh check

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

```
~/Garden
[mission 6] $ mv coin_1 coin_2 coin_3 /home/kali/gameshell/World/Forest/Hut/Chest
~/Garden
[mission 6] $ cd
~/Forest/Hut/Chest (~:/studenti/nicola/scuola)
[mission 6] $ cd Forest/Hut/Chest
~/Forest/Hut/Chest (~:/studenti/nicola/scuola)
[mission 6] $ ls
coin_1  coin_2  coin_3
~/Forest/Hut/Chest
[mission 6] $ gsh check

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

```
~/Garden
[mission 7] $ mv .15798_coin_3 .32409_coin_2 .7539_coin_1 /home/kali/gameshell/World/Forest/Hut/Chest/
~/Garden
[mission 7] $ cd
~/Forest/Hut/Chest (~:/studenti/nicola/scuola)
[mission 7] $ cd Forest/Hut/Chest
~/Forest/Hut/Chest (~:/studenti/nicola/scuola)
[mission 7] $ ls -a
.  ..  .15798_coin_3  .32409_coin_2  .7539_coin_1  coin_1  coin_2  coin_3
~/Forest/Hut/Chest (~:/studenti/nicola/scuola)
[mission 7] $ gsh check

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

```
~/Castle/Cellar
[mission 8] $ rm 12691_spider_8 17413_spider_35 20262_spider_7 23551_spider_47 26273_spider_38 28601_spider_3
587_spider_33 30974_spider_22 4086_spider_19 5424_spider_20 9368_spider_23 12130_spider_5 27875_spider_32 31309
~/Castle/Cellar
[mission 8] $ ls
10722_bat_5 14431_bat_1 19526_bat_2 21905_bat_3 26172_bat_4 barrel_of_apples
~/Castle/Cellar
[mission 8] $ gsh check
Congratulations, mission 8 has been successfully completed!
```

Nelle missioni successive andiamo ad utilizzare i seguenti comandi:

**Is -a:** verifichiamo il contenuto delle directory e con **-a** visualizziamo i file nascosti

```
~/Castle/Cellar
[mission 9] $ ls -a
. .11385_bat_1 .13245_spider_24 .15475_spider_50 19526_bat_2 21905_bat_3 .25453_spider_19 .26752_spider_29 .28195_spider_21 .31271_spider_10 .4414_spider_12 .5_spider_27 .9304_spider_17
.. .11608_spider_29 .13247_spider_33 .16185_spider_8 .19085_bat_2 .23088_spider_38 .25615_spider_25 .26886_spider_46 .28518_spider_9 .31283_spider_3 .5144_spider_36 .7593_spider_1 .9502_spider_41
10984_spider_18 .11652_spider_4 .14112_spider_34 .16328_spider_13 .20552_bat_3 .23378_spider_6 .25813_spider_49 .26945_spider_36 .29558_spider_14 .31938_bat_5 .5564_spider_11 .825_spider_22 barrel_of_apples
10722_bat_5 .12461_spider_31 14431_bat_1 .16958_spider_5 .21136_spider_44 .24557_spider_28 26172_bat_4 .28086_spider_37 .31217_bat_4 .32195_spider_43 .5921_spider_32 .861_spider_26
.11012_spider_2 .13159_spider_39 .15104_spider_48 .17089_spider_42 .21736_spider_16 .25172_spider_48 .26366_spider_47 .28103_spider_20 .31221_spider_35 .3683_spider_45 .5947_spider_15 .8658_spider_7

~/Castle/Cellar
[mission 9] $ rm .*_spider.*
~/Castle/Cellar
[mission 9] $ ls -a
. .10722_bat_5 .11385_bat_1 14431_bat_1 19526_bat_2 .19085_bat_2 .20552_bat_3 21905_bat_3 26172_bat_4 .31217_bat_4 .31938_bat_5 barrel_of_apples
..
~/Castle/Cellar
[mission 9] $ gsh check
Congratulations, mission 9 has been successfully completed!
```

**cp:** viene utilizzato per copiare dei file in un'altra directory inserendo i nomi dei file e il percorso di destinazione.

```
~/Castle/Great_hall
[mission 10] $ cp standard_1 standard_2 standard_3 standard_4 /home/kali/gameshell/World/Forest/Hut/Chest
```

Alla missione 13 ci viene chiesto di stampare un calendario:

```
~/Forest/Hut/Chest
[mission 13] $ cal
      July 2023
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
```

**cat:** ci permette di stamparlo a schermo nell'anno corrente

mentre affiancondoli un anno affinaco avremo a schermo il calendario di quell'anno.

In questo caso io ho dovuto cercare l'anno 1909 per sapere quale era il giorno corretto che mi chiedeva la domanda:

```

      July                August                September
Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa
  1 2 3                1 2 3 4 5 6 7          1 2 3 4
 4 5 6 7 8 9 10       8 9 10 11 12 13 14      5 6 7 8 9 10 11
11 12 13 14 15 16 17 15 16 17 18 19 20 21    12 13 14 15 16 17 18
18 19 20 21 22 23 24 22 23 24 25 26 27 28    19 20 21 22 23 24 25
25 26 27 28 29 30 31 29 30 31                26 27 28 29 30

      October            November            December
Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa
  1 2 3                1 2 3 4 5 6          1 2 3 4
 3 4 5 6 7 8 9        7 8 9 10 11 12 13      5 6 7 8 9 10 11
10 11 12 13 14 15 16 14 15 16 17 18 19 20    12 13 14 15 16 17 18
17 18 19 20 21 22 23 21 22 23 24 25 26 27    19 20 21 22 23 24 25
24 25 26 27 28 29 30 28 29 30                26 27 28 29 30 31
31

~/Forest/Hut/Chest
[mission 13] $ gsh check
What was the day of the week for the 09-16-1909?
 1 : Monday
 2 : Tuesday
 3 : Wednesday
 4 : Thursday
 5 : Friday
 6 : Saturday
 7 : Sunday
Your answer: 4

Congratulations, mission 13 has been successfully completed!

```

Adesso ci verrà chiesto di realizzare un Alias.

```

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.
Define the synonym
la
for the command
ls -A

```

Alias **la='ls -A'** è il comando che abbiamo eseguito che va ad associare ad la il comando **'ls -A'**.

Quindi andando a lanciare la avremo lo stesso risultato.

```

~/Forest/Hut/Chest
[mission 14] $ alias la='ls -A'

~/Forest/Hut/Chest
[mission 14] $ ls -A
.15798_coin_3      1828_tapestry_01  23920_tapestry_04  .32409_coin_2    50258_tapestry_08  60717_tapestry_02
17779_tapestry_05 18834_tapestry_03 25025_tapestry_09  48087_tapestry_06 55219_tapestry_10 61009_tapestry_07

~/Forest/Hut/Chest
[mission 14] $ gsh check

Congratulations, mission 14 has been successfully completed!

```

Successivamente andremo a realizzare un file nella nostra directory Chest per i nostri appunti e dovremo fare in modo che possa essere lanciato da ovunque noi ci troviamo.

```
()=(  
    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.  
  
    Details  
  
    ``nano`` is a command-line text editor. You can use it whenever you need to edit a file from the shell.
```

Qui abbiamo realizzato il file e gli abbiamo impostato l'alias in tal modo da poter richiamare il nostro file. Abbiamo utilizzato il comando **alias** e ovviamente il comando **nano** che ci permette di aprire un file esistente o crearne uno nuovo eventualmente non dovesse esserci.

```
~  
[mission 16] $ alias journal="cat /home/kali/gameshell/World/Forest/Hut/Chest/journal.txt"  
  
~  
[mission 16] $ journal  
Hi,this is my personal space of work.  
  
~  
[mission 16] $ alias journal="nano /home/kali/gameshell/World/Forest/Hut/Chest/journal.txt"  
  
~  
[mission 16] $ journal  
  
~  
[mission 16] $ gsh check  
Congratulations, mission 16 has been successfully completed!
```

Qui il gioco ci chiede di mandare il comando **xeyes** per far apparire degli occhi che seguono perennemente il puntatore, aggiungendo una **&** prima del comando saremo in grado anche di **metterlo in background** (nello screen lo vediamo guardare il codice in modo curioso)

```
~/Castle/Cellar/.Lair_of_the_spider_queen TLKIGKMRUVpikgXt IskVZwxMzzymAIMD  
[mission 18] $ xeyes  
^C  
~/Castle/Cellar/.Lair_of_the_spider_queen TLKIGKMRUVpikgXt IskVZwxMzzymAIMD  
[mission 18] $ xeyes&  
[1] 255443  
~/Castle/Cellar/.Lair_of_the_spider_queen TLKIGKMRUVpikgXt IskVZwxMzzymAIMD  
[mission 18] $ exeyes6
```

Nelle missioni successive continuiamo a cercare e spostare cartelle e file con i comandi visti prima:

```
~/Garden/Maze/f4298612059410ee/7208186db07e9fa2d21bc35/85125caec8dc2e87b802146ee  
[mission 20] $ ls  
00000_silver_coin_00000  
  
~/Garden/Maze/f4298612059410ee/7208186db07e9fa2d21bc35/85125caec8dc2e87b802146ee  
[mission 20] $ mv 00000_silver_coin_00000 /home/kali/gameshell/World/Forest/Hut/Chest/  
  
~/Garden/Maze/f4298612059410ee/7208186db07e9fa2d21bc35/85125caec8dc2e87b802146ee  
[mission 20] $ gsh check  
Congratulations, mission 20 has been successfully completed!  
  
You are back at the entrance of the maze...
```

Abbiamo trovato il gettone d'oro, d'argento e adesso per ricercare quello d'oro dovremo utilizzare un metodo differente.

Per ricercarlo utilizziamo il comando **find -iname \*gold\***.

Quando i file sono troppi possiamo utilizzare il **find** (ricerca) per ricercare un testo (**-iname**) il quale al suo interno abbia la parola gold. Gli asterischi in questo caso sono utilizzati per "dire" che possono esserci caratteri prima o dopo gold.

Successivamente andiamo a spostarlo nella nostra directory solita.

```
[mission 21] $ cd Forest/Hut/Chest/
~/Forest/Hut/Chest
[mission 21] $ ls
17779_tapestry_05  18834_tapestry_03  25025_tapestry_09  50258_tapestry_08  60717_tapestry_02  coin_1  coin_3  00000_copper_coin_00000  painting_TBTFWBeU  standard_2  standard_4
1828_tapestry_01  23920_tapestry_04  48087_tapestry_06  55219_tapestry_10  61009_tapestry_07  coin_2  journal.txt  00000_silver_coin_00000  standard_1  standard_3

~/Forest/Hut/Chest
[mission 21] $

~/Garden/Maze
[mission 21] $ find -iname *gold*
./1d981c87f5246/6678b9d70ef376/71b8b9a92714604170b4a6627865bf/Gold_Coin_2
./5f39ee9fb7e74a23ba8015516eabaa9/8a0a59e46bf847d7b/3f5cc7d3f6fc4a93a8/gold_coin_1

~/Garden/Maze
[mission 21] $ mv ./1d981c87f5246/6678b9d70ef376/71b8b9a92714604170b4a6627865bf/Gold_Coin_2 ./5f39ee9fb7e74a23ba8015516eabaa9/8a0a59e46bf847d7b/3f5cc7d3f6fc4a93a8/gold_coin_1 /home/kali/gameshell.1/World/Forest/Hut/Chest/
~/Garden/Maze
[mission 21] $ gsh check
Congratulations, mission 21 has been successfully completed!
```

Adesso, entrando nella caverna troveremo un vecchietto a cui dare una mano (aprendo, leggendo ed eseguendo file)

```
~/Garden/Maze
[mission 22] $ gsh goal

Mission goal

A forgetful old hermit called Servillus has set up camp in a cave with his old, leather-bound potion book.
Go to the cave and help him remember the recipe of his famous herbal tea.

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 recipe (including its title), but nothing else.

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

Qui ci viene chiesto di visualizzare un file, ma soltanto le parti inerenti alla ricetta del tè.

Abbiamo già visto il comando **cat** per concatenare e stampare, ma a noi serve solo una parte del file di testo.

Qui ci viene incontro **head** che come vediamo ci permette di definire il numero di righe da visualizzare del file aggiungendoci percorso e il nome file.

```
~/Mountain/Cave
[mission 22] $ head -7 Book_of_potions/page_07
Herbal tea

1) Boil water.
2) Add herbs from the forest.
3) Let it sit for five minutes and drink while hot.

~/Mountain/Cave
[mission 22] $ gsh check
Congratulations, mission 22 has been successfully completed!
```

Qui invece osserviamo un comando opposto a **head**, ovvero **tail** che inizia a contare le righe dalla fine.

Abbiamo così rimosso righe superficiali e titoli.

```
~/Mountain/Cave
[mission 23] $ 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 23] $ gsh check

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

Possiamo ulteriormente concatenare i comandi per ottenere un risultato più preciso e delineato, inerente alle richieste della missione.

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

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

Qui vediamo effettivamente quanto è utile il comando **cat** per concatenare uno o più file, necessario per riuscire a vedere interamente le ricette del libro.

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

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



Anche se nello screen non è molto leggibile è stato usato il comando **kill** per terminare un processo. Precisamente il comando utilizzato è **kill -9 (PID)** e viene utilizzato rispetto al **kill** normale per terminare tutti i processi evitando che i processi si riavviino in loop.

Per visualizzare i processi attivi abbiamo utilizzato il comando **ps**.

```
~/Castle/Cellar
[mission 29] $ ps
  PID TTY          TIME CMD
  7114 pts/1        00:00:00 zsh
  7562 pts/1        00:00:00 bash
  7591 pts/1        00:00:02 bash
 215147 pts/1        00:00:00 nice_fairy
 215148 pts/1        00:00:00 mischievous_imp
 215155 pts/1        00:00:00 spell
 215156 pts/1        00:00:00 spell
 215157 pts/1        00:00:00 spell
 215158 pts/1        00:00:00 tail
 215173 pts/1        00:00:00 tail
 251072 pts/1        00:00:00 sleep
 251121 pts/1        00:00:00 sleep
 251171 pts/1        00:00:00 sleep
 251179 pts/1        00:00:00 ps
```

```
~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 28] $ kill
          *#@*
          6_*/~
          !$-#

-9 100279

~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 28] $ gsh goal
```

Successivamente ci verrà chiesto di nascondere alla lettura dei file e lo facciamo unendo a **ls** il comando **--hide** in modo che da leggere tutto il resto per poi visualizzare il file con **less**.

```
~/Castle/Main_building/Library/Merlin_s_office
[mission 32] $ ls --hide=candle --hide=Drawer > Drawer/inventory.txt

~/Castle/Main_building/Library/Merlin_s_office
[mission 32] $ cd Drawer/

~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 32] $ less inventory.txt

~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 32] $ gsh check
Congratulations, mission 32 has been successfully completed!

While you are waiting, a snail goes by ...

|                                     |
--+-----+-----+-----+-----+
| Use the command                    |
| $ gsh help                        |
| to get the list of "gsh" commands. |
--+-----+-----+-----+-----+
|                                     |
```



Arriviamo davanti al trono del re, ma non avremo i permessi inizialmente. Li andiamo ad abilitare con il comando **chmod** e andando ad assegnare all'user (u) i permessi di execute (x).

```
~/Castle/Main_building/Throne_room
[mission 35] $ chmod u+w Kings_quarter/

~/Castle/Main_building/Throne_room
[mission 35] $ chmod u+x Kings_quarter/

~/Castle/Main_building/Throne_room
[mission 35] $ cd Kings_quarter/

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 35] $ ls

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 35] $ ls -a
./  ../

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 35] $ gsh check

Congratulations, mission 35 has been successfully completed!

| eseguibili |
--+-----+--
| Use the command |
| $ gsh help |
| to get the list of "gsh" commands. |
--+-----+--
```

Successivamente faremo lo stesso procedimento di abilitazione dei permessi con **chmod** per far sì di accedere alle stanze del re e leggere il messaggio.

```
~/Castle/Main_building/Throne_room/Kings_quarter
[mission 36] $ chmod u+rw .secret_note

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 36] $ cat .secret_note
3835156893

~/Castle/Main_building/Throne_room/Kings_quarter
[mission 36] $ gsh check
What's the combination to open the King's safe? 3835156893

Congratulations, mission 36 has been successfully completed!
```

Qui visualizziamo con **cat** la nostra coroncina con il numero segreto che ci permetterà di sbloccare la missione successiva che ci permetterà di spostare la corona nella Chest.

```
~/Castle/Main_building/Throne_room
[mission 37] $ chmod u+rx Safe/

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

~/Castle/Main_building/Throne_room/Safe
[mission 37] $ chmod u+rx crown

~/Castle/Main_building/Throne_room/Safe
[mission 37] $ cat crown
(^.-.+.-)
(^\\/^\\/^)
\\@*@*@/
{_157_}
```

```
[mission 37] $ cd Safe/

~/Castle/Main_building/Throne_room/Safe
[mission 37] $ mv crown /home/kali/gameshell/World/Forest/Hut/Chest

~/Castle/Main_building/Throne_room/Safe
[mission 37] $
```

Qui arriviamo ad un punto in cui avremo molti file e directory e a noi ci serve il file così scritto Ruby. Ritorniamo ad utilizzare il **find**, ma questa volta gli associamo il **-type f** per specificare che la ricerca deve essere solo sui file, **|** per concatenare più comandi assieme e infine il **grep** che andrà a cercare la parola Ruby con la condizione **-i** che permetterà di non fare differenza fra caratteri maiuscoli o minuscoli.

```
~/Garden/Maze
[mission 38] $ find . -type f | grep -i "Ruby"

~/Garden/Maze
[mission 38] $ find . -type f | xargs grep -i "Ruby"
./bfba447634a0a/63135c3de5a45b42ef56453/7a2069a8bb014f7080f1a320/38668 ruby 1b50de5d6285517ec1909b6aacce04122001ea4e

~/Garden/Maze
[mission 38] $ cd ./bfba447634a0a/63135c3de5a45b42ef56453/7a2069a8bb014f7080f1a320/38668
bash: cd: ./bfba447634a0a/63135c3de5a45b42ef56453/7a2069a8bb014f7080f1a320/38668: Not a directory

~/Garden/Maze
[mission 38] $ mv ./bfba447634a0a/63135c3de5a45b42ef56453/7a2069a8bb014f7080f1a320/38668 /home/kali/gameshell/World/Forest/Hut/Chest/

~/Garden/Maze
[mission 38] $ gsh check

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

Qui a differenza di prima abbiamo anche aggiunto il comando **xargs** che ci serve per passare i dati forniti dal primo comando **find** in input al **grep** andando a definire ancora meglio la ricerca del 'diamond' nelle decine di directory.

In aggiunta al **grep** abbiamo inserito **-l** che va a specificare solo i nomi dei file corrispondenti.

```
~/Garden/Maze
[mission 39] $ find . -type f | xargs grep -i -l "diamond"
./659837d27b0d30499992299/856c2103a260d496f5dd993e/7903cd1a58eed543a5cc2abb

~/Garden/Maze
[mission 39] $ mv ./659837d27b0d30499992299/856c2103a260d496f5dd993e/7903cd1a58eed543a5cc2abb /home/kali/gameshell/World/
.bashrc  Castle/  Forest/  Garden/  .gshrc  .lessht  .local/  Mountain/ Stall/

~/Garden/Maze
[mission 39] $ mv ./659837d27b0d30499992299/856c2103a260d496f5dd993e/7903cd1a58eed543a5cc2abb /home/kali/gameshell/World/Forest/Hut/Chest/

~/Garden/Maze
[mission 39] $ gsh check

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

Anche qui andiamo a fare una ricerca specifica inserendo però questa volta al **grep** il **-v** che mi permetterà di selezionare tutte righe che non corrispondono a 'paid'.

```
(0)
~/Stall
[mission 40] $ ls | grep -v boring_object | xargs -I {} grep -i king {} | grep -v -i "paid"
the King bought a leather ball for 4 coppers.
the King bought a cow for 2 coppers.
Catherine from next town bought a walking stick for 2 coppers.
the King bought an apple for 5 coppers.
the King bought a chackal for 6 coppers.
the King bought an apple for 5 coppers.
Thibaud Messonnier bought a walking stick for 4 coppers.
the King bought a blanket for 3 coppers.
(1)
~/Stall
[mission 40] $ gsh check
How much does the king owe? 25

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

Nella missione finale dobbiamo decifrare un messaggio cifrato sfruttando le conversioni dei caratteri.

Utilizziamo il comando **tr** a cui assegniamo come primo l'insieme di caratteri da tradurre e come secondo insieme di caratteri corrispondenti e grazie al comando **<** tutto questo avverrà nel file secret message, per poi andarlo a stampare.

```
~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 42] $ tr 'abcdefghijklmnopqrstuvwxyz' 'nopqrstuvwxyzabcdefghijklm' < secret_message
here is my will:
ypu will get ny chest, aod everythiog it cpoaios.
this chest is io the cellar, aod the wprd tp nake
it re-aqqear is: kbld
nerlio the eochaoter

~/Castle/Main_building/Library/Merlin_s_office/Drawer
[mission 42] $ cd

~/
[mission 42] $ cd Castle/Cellar/

~/Castle/Cellar
[mission 42] $ ls -la
./      10123_snowflake  .11385_bat_1  14431_bat_1  .19805_bat_2  21905_bat_3  .31217_bat_4  34729_sr
../     10722_bat_5       11835_snowflake 19526_bat_2  .20552_bat_3  26172_bat_4  .31938_bat_5  40526_sr

~/Castle/Cellar
[mission 42] $ gsh check
What's the key that will make Merlin's chest to appear?
kbld

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

Hon

sequ

cia

CONGRATULATIONS!

You have finished all the missions.