Esercizio W8D4 (Benchmark 2) - Parte Obbligatoria

GUIDA OPERATIVA

L'esercitazione W8D4 (Benchmark 2), nella sua parte obbligatoria, prevede l'installazione di GameShell, un gioco pensato per familiarizzare con i comandi Linux. La procedura per l'installazione è la seguente:

- sudo apt update
- sudo apt install gettext man-db procps psmisc nano tree bsdmainutils x11-apps wget
- https://github.com/phyver/GameShell/releases/download/latest/gameshell.sh

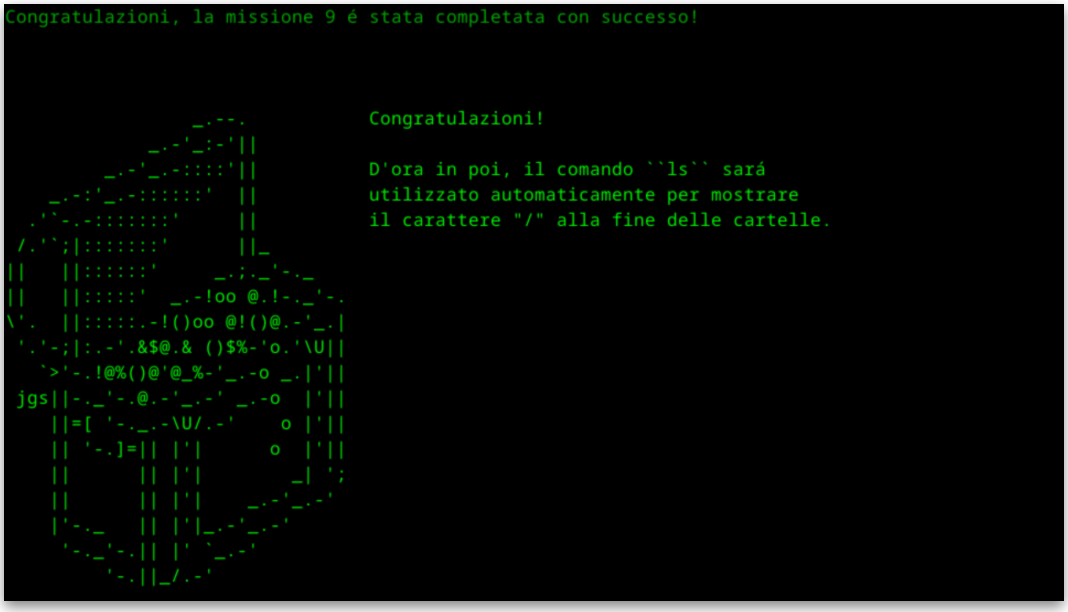
Una volta installato, il gioco può essere lanciato con il comando **bash gameshell.sh**

~~~

RISOLUZIONE

Le prime missioni si concentrano sull'utilizzo dei comandi per lo spostamento all'interno del file system di Linux. Tra i comandi più utilizzati troviamo **cd** per muoversi tra le directory, **ls** per visualizzare le liste dei file, **pwd** per controllare la directory corrente, **cd -** per tornare alla directory precedente e **cd ..** per salire di un livello nella gerarchia delle directory.

Il primo checkpoint è alla missione 3, che richiede di tornare con cd alla cartella di origine e poi di eseguire vari spostamenti in una sola volta. La sintassi corretta è

cd
cd Castello/Palazzo_principale/Sala_del_trono


~~~

Dalla quarta missione, il set di comandi disponibili si è ampliato, includendo ad esempio **mkdir** per creare nuove cartelle, **rm** per rimuovere file, **mv** per spostare file, **ls -a** per visualizzare una lista di tutti i file, inclusi quelli nascosti, e le wildcard ***** (che rappresenta qualsiasi sequenza di caratteri) e **?** (che rappresenta un singolo carattere).

Il checkpoint successivo è al livello 9, con la richiesta di rimuovere dalla cartella 'Cantina' tutti i file, inclusi quelli nascosti, che contengono la stringa 'ragno' nel nome. La sintassi utilizzata è:

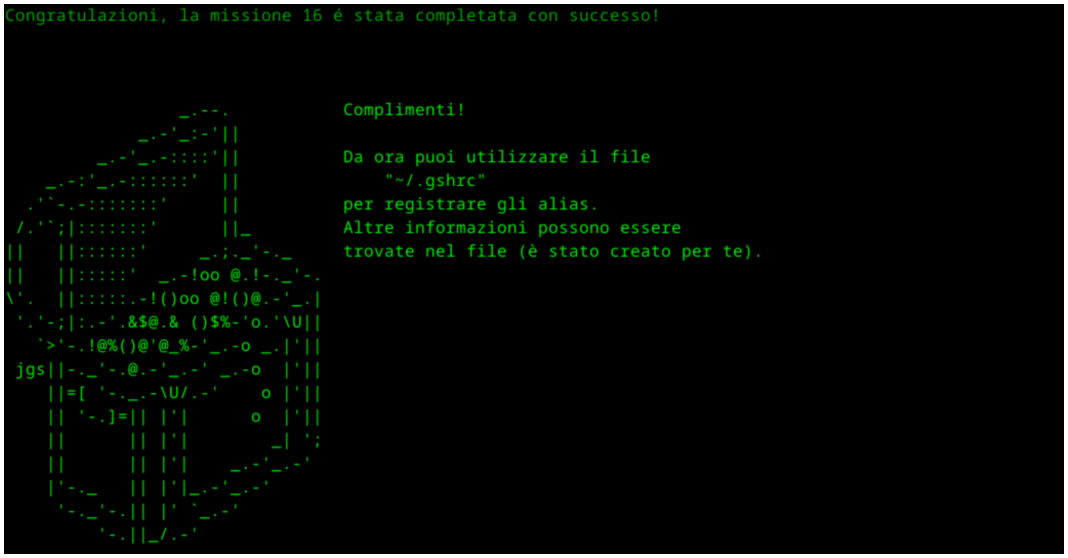
rm .*ragno*



A partire dalla missione 10, il set di comandi disponibili si espande ulteriormente, includendo **cp** per copiare file, **ls -l** per visualizzare una lista dei file in formato lungo (che include permessi, proprietario, dimensione, data di modifica, ecc.), **cal** per visualizzare un calendario (anche di uno specifico anno), **alias** per creare scorciatoie per comandi, e **nano** per aprire e modificare file con l'editor di testo nano.

Il checkpoint è alla missione 16, che prevede la creazione di un alias per accedere rapidamente al file TXT 'storico.txt' situato nella cartella 'Foresta/Capanna/Cassa'. La sintassi utilizzata è:

alias storico='nano /Foresta/Capanna/Cassa/storico.txt'



A partire dalla missione 18, l'attenzione si sposta dai comandi relativi al file system ai comandi relativi ai processi. Il primo comando di questo tipo è **xeyes**, un'applicazione grafica che mostra un paio di occhi che seguono il movimento del mouse. La missione prevede di attivarlo, fermarlo e riavviarlo in background. Dalla missione 21, il gioco introduce il labirinto, un intricato insieme di directory dai nomi complessi, progettato per incoraggiare l'uso del **completamento automatico** tramite il tasto TAB.

Il checkpoint per questa sezione è al livello 21, dove si richiede di localizzare una moneta di rame all'interno del labirinto e di trasferirla nella cartella 'Cassa', che funge da inventario del giocatore. La sintassi utilizzata per completare questa missione è:

cd /Giardino/Labirinto/9c627be05a0242580fc4faaa2fd5/ef64be6c5a658054f2e138/a2b63b98b3420f4cc
mv 00000_moneta_di_rame_00000 /Foresta/Capanna/Cassa/

Le missioni successive continuano con la ricerca di oggetti nel labirinto e il loro spostamento nell'inventario. A partire dalla missione 23, viene introdotto il comando **find** per la ricerca di file che soddisfano specifiche condizioni. Dalla missione 24, con l'introduzione del personaggio 'Servillus', si utilizzano i comandi per visualizzare il contenuto dei file: **cat** (per concatenare e visualizzare il contenuto), **head** (per visualizzare le prime righe) e **tail** (per visualizzare le ultime righe).

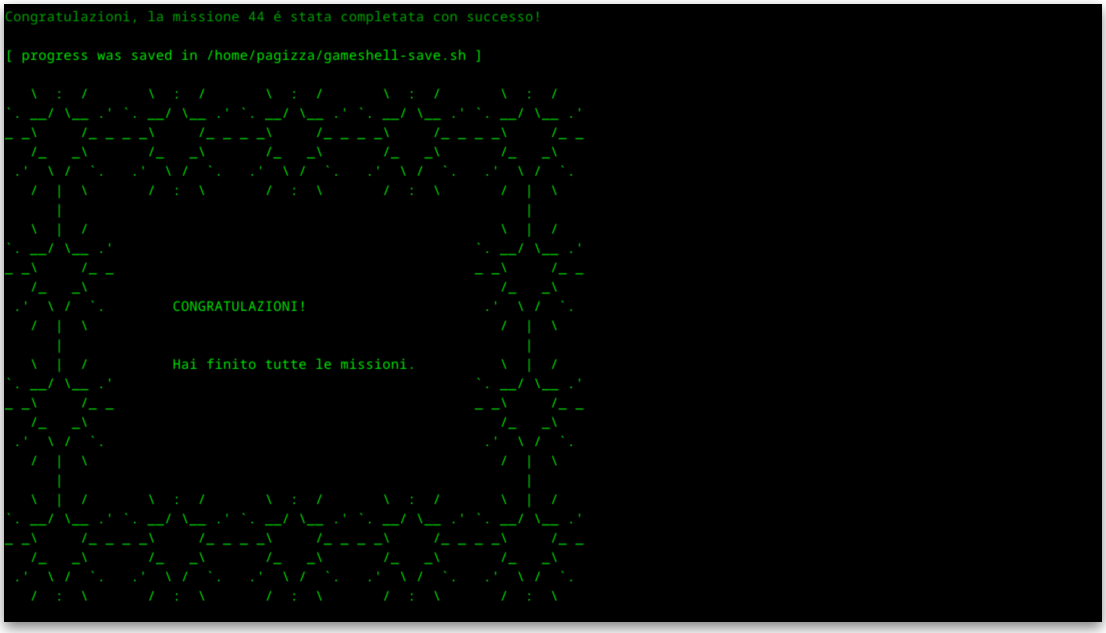
Dalla missione 29, il gioco introduce i **processi**. In particolare, compaiono processi 'malevoli' che stampano codice indesiderato sullo schermo. Questa sezione mostra l'uso del comando **ps** per elencare i processi in esecuzione e i vari comandi **kill** per terminarli.

A partire dalla missione 32, con l'introduzione del personaggio 'Merlino', le sfide si concentrano sui calcoli. Inizialmente, si risolvono operazioni di somma a mente. Successivamente, per le moltiplicazioni (a causa del limite di tempo), è necessario individuare il file corretto e fornirlo **come input al comando gsh check** usando il redirect di input (<). Dalla missione 36, che nello scenario si apre con la 'pazzia di Merlino', le attività riguardano l'esecuzione di file. Vengono introdotti i concetti di output standard (**stdout**) e output di errore standard (**stderr**). Dalla missione 37, l'attenzione si sposta sulla gestione dei **permessi**, in particolare sull'uso del comando chmod. La missione 39 coinvolge il 'furto' della Corona del Re.

Le missioni 40 e 41 richiedono l'impostazione dei parametri corretti per la ricerca e il 'furto' di un rubino e un diamante. Nella missione 42, si utilizza il comando **grep** con diverse opzioni per determinare l'ammontare del debito del Re verso il Mercante. La missione 44 consiste nel decifrare un messaggio cifrato con **il cifrario di Cesare**. Nonostante il gioco suggerisca spostamenti di 10 o 16 caratteri, la decrittazione corretta si ottiene con uno spostamento di 14. Questo rivela la parola chiave necessaria per aprire il forziere di Merlino e completare il gioco. Il comando utilizzato per la decrittazione è:

tr 'a-z' 'o-za-n' < messaggio_segreto

```
~/Castello/Palazzo_principale/Biblioteca/Ufficio_di_Merlino/Cassetto
[mission 44] $ tr 'a-z' 'o-za-n' < messaggio_segreto
queste sono le mie volontá:
avrai la mia cassa e tutto quello che contiene.
questa cassa é nella cantina e la parola per farla
apparire é: laiq
merlino lo stregone

~/Castello/Palazzo_principale/Biblioteca/Ufficio_di_Merlino/Cassetto
[mission 44] $ _
```



In conclusione, l'esperienza con GameShell si è rivelata un metodo efficace per familiarizzare con un'ampia gamma di comandi Linux. Le missioni del gioco hanno guidato l'utente attraverso l'utilizzo di comandi che spaziano dalla navigazione nel file system (come cd e ls) alla manipolazione di file e directory (mkdir, rm, mv), dall'interazione con i processi (ps, kill) alla gestione dei permessi (chmod).

Il completamento di queste attività fornisce una solida base per l'applicazione pratica di Linux in contesti reali.