

Relazione - finger

Il programma presentato ricrea il comando "finger" di Linux.

Utilizza file di sistema come /etc/passwd, /var/run/utmp, e /var/log/wtmp per ottenere e visualizzare i dati degli utenti, fornendo varie opzioni di filtraggio e formattazione attraverso la riga di comando.

In base alla presenza o assenza di nomi utente e alle opzioni specificate, il programma chiama funzioni specifiche per elaborare e visualizzare le informazioni:

caso 1. Se non vengono forniti nomi utente, vengono chiamate `call_finger_noparams` o `finger_noparams_opt_l` per visualizzare le informazioni di tutti gli utenti attivi

caso 2. Se vengono forniti nomi utente, la funzione `call_finger_with_params` viene chiamata per ogni nome, per cercare e visualizzare le informazioni specifiche di quell'utente.

I nomi utente forniti come parametri vengono memorizzati in un array dinamico.

caso 1:

cerco solo utenti attivi, quindi scorro UTMP per trovarli e cerco le loro altre info in PASSWD

-l : utenti attivi in formato long

-s (default) : utenti attivi in formato short, se -l = true, vince -l

-m : non ha effetto

-p : se -l = true, non mostro .Plan, altrimenti non ha effetto

caso 2:

cerco gli utenti specificati, quindi scorro PASSWD per trovarli e prendo le loro altre info in UTMP

Se un utente non viene trovato in UTMP allora verrà ricercato in WTMP

-l (default) : utenti specificati in formato long

-s : utenti specificati in formato short

-m : cerco il nome degli utenti solo tra gli username

-p : non mostro .Plan, se -s = true e -l = false non ha effetto

In **check_wtmp** io cerco l'ultimo accesso dell'utente ed ottengo tutte le informazioni necessarie alla stampa. Per quanto riguarda l'opzione -s c'è una differenza tra la stampa con un'interfaccia grafica e la stampa senza un'interfaccia grafica. Ho quindi ottenuto il valore intero associato alla funzione `graphical_user_interface` relativo all'utente specificato per poter stampare nel modo corretto l'opzione specifica.

--

Gestione delle Opzioni

Il programma accetta diverse opzioni da riga di comando che influenzano il comportamento e l'output:

-l: Visualizza informazioni dettagliate sugli utenti.

-s: Visualizza informazioni in formato breve.

-m: Cerca solo per nome di login.

-p: Non stampa il file .plan dell'utente.

Queste opzioni vengono elaborate all'inizio del programma tramite un ciclo che utilizza la funzione `getopt`

--

Verifica dei Nomi Stampati

La funzione `check_printed_names` controlla se un nome utente è già stato stampato e, in caso contrario, lo aggiunge a un elenco per evitare duplicazioni nell'output.

Mentre verifico se un nome è in `passwd`, se non lo trovo vuol dire che non esiste quindi lo aggiungo alla lista `not_found_names` che contiene tutti i nomi degli utenti non trovati e verrà stampata alla fine attraverso la funzione `print_not_existing_user` richiamata nel `main`.

Spiegazione di context

Context ci aiuta a capire che formato di stampa dobbiamo ottenere. Se `context` è 1 la stampa è long, se `context` è 0 la stampa sarà in formato short. Importante per poter differenziare i due output

La funzione **`print_gecos_fields`** ha lo scopo di stampare i campi GECOS degli utenti presenti nel sistema.

Questi dati sono memorizzati nel file `/etc/passwd`.

La funzione apre il file `/etc/passwd` in modalità di lettura. Ogni linea del file `/etc/passwd` è una stringa che contiene vari campi separati da due punti (:).

I campi includono il nome utente, la password, l'UID (User ID), il GID (Group ID), il campo GECOS, la directory home e la shell.

Per ogni linea letta, la funzione identifica il quinto campo (campo GECOS), che è separato dai campi circostanti da due punti.

Il campo GECOS stesso può contenere diverse informazioni separate da virgole, come il nome completo, l'ufficio, il numero di telefono dell'ufficio, il numero di telefono di casa e altre informazioni di contatto.

La funzione divide il campo GECOS in questi sottocampi utilizzando la virgola come delimitatore.

Infine, la funzione stampa ciascuno di questi sottocampi in modo ordinato attraverso un `for` diversificando la stampa "long" e la stampa "short"