MyLast

Folosirea unui sistem de stări pentru reconstruirea sesiunilor de logare

Gaitur Ruslan

2025

Cuprins

1 Prezentare Generală **3**

2 Caracteristici Principale **3**

3 Instalare și Cerințe **3**

3.1 Dependințe 3

3.2 Permisiuni 3

4 Utilizare **4**

4.1 Sintaxă 4

4.2 Moduri de Apel `<mod>` 4

4.3 Opțiuni de Filtrare `[flag_filtrare]` 4

4.4 Parametri de Filtrare `[valoare_filtrare]` 4

4.5 Interfața Vizuală 5

4.5.1 Arhitectura Interfeței 5

4.5.2 Diagrama de Afișare 5

5 Arhitectura Proiectului **6**

5.1 Arhitectura Funcțiilor de Bază 6

5.1.1 Gestionarea Jurnalelor 6

5.1.2 `mylast` 6

5.1.3 `mylastb` 6

5.2 Arhitectura Funcțiilor Ajutătoare 7

5.2.1 `get_duration` 7

1 Prezentare Generală

MyLast este un script Bash conceput pentru a analiza jurnalele de autentificare ale sistemului situate în `/var/log/`. Acesta reproduce funcționalitatea comenzilor Linux standard `last` și `lastb`.

Scriptul extrage date din fișierele `auth.log.x` (inclusiv cele comprimate cu `gzip`) pentru a reconstrui istoricul sesiunilor utilizatorilor fizici și a identifica încercările de autentificare eșuate.

2 Caracteristici Principale

- **Procesare Jurnale:** Citește și concatenează `auth.log`, `auth.log.1`, etc.
- **Reconstruirea Sesiunii:** Reconstruiește exact firul de autentificare, deconectare al utilizatorului și oprirea (*shutdown*) sistemului.
- **Reconstruirea Eșecurilor de Logare:** Determină sesiunile în care au avut loc eșecuri de autentificare.
- **Filtrare:** Permite filtrarea rezultatelor după constrângeri de timp specifice (De la, Până la, Prezent), dar și după numărul de afișări.

3 Instalare și Cerințe

3.1 Dependințe

Asigurați-vă că următoarele sunt instalate:

- `bash` (Mediul Shell)
- `gzip` / `zcat` (Citirea fișierelor comprimate)
- `awk`, `cut`, `grep` (Procesare text)

3.2 Permisiuni

Deoarece scriptul citește din `/var/log/auth.log`, necesită de obicei privilegii de **root**.

```
sudo ./mylast.sh [...]
```

4 Utilizare

Scriptul acceptă argumente în linia de comandă pentru a determina modul de operare și logica de filtrare.

4.1 Sintaxă

```
./mylast.sh <mod> [flag_filtrare] [valoare_filtrare]
```

4.2 Moduri de Apel `<mod>`

Determină modul de analiză:

Flag	Descriere
`-l`	**Ultimele Logări:** Analizează sesiunile reușite.
`-lb`	**Logări Eșuate:** Analizează încercările eșuate.

Tabel 1: Moduri Principale de Execuție

4.3 Opțiuni de Filtrare `[flag_filtrare]`

Acești parametri sunt valabili doar pentru modul `-l` de rulare al scriptului.

Flag	Format
`-n`	Afișează ultimele N intrări.
`-s`	Arată sesiunile **de la** data specificată.
`-t`	Arată sesiunile **până la** data specificată.
`-p`	Arată sesiunile **active** la timpul specificat.

Tabel 2: Opțiuni de Filtrare

4.4 Parametri de Filtrare `[valoare_filtrare]`

Reprezintă valoarea ce ține de filtrare. Fiecare flag de filtru are un format unic pentru propria valoare.

Flag	Format	Descriere
`-n`	**INT**	Număr întreg
`-s`	**YYYY-MM-DD-HH-mm**	Dată și timp
`-t`	**YYYY-MM-DD-HH-mm**	Dată și timp
`-p`	**YYYY-MM-DD-HH-mm**	Dată și timp

Tabel 3: Format Acceptat Pentru Parametrii Filtrării

4.5 Interfața Vizuală

Deși **MyLast** este o aplicație bazată pe linia de comandă, interfața sa vizuală este definită de formatarea strictă a datelor returnate la *Standard Output*. Interacțiunea are loc exclusiv în cadrul terminalului.

4.5.1 Arhitectura Interfeței

Scriptul utilizează un *layout tabular bazat pe text*. Structura vizuală este proiectată pentru lizibilitate maximă, asemenea comenzilor `last, lastb`.

Fiecare linie afișată reprezintă o entitate distinctă și este împărțită în coloane vizuale delimitate prin spațiere dinamică.

4.5.2 Diagrama de Afișare

```
|-------------------------------------------------------------|
|Utilizator|Sesiune|    Data    |Ore de Început și Sfârșit|Durată|
|----------|-------|----------|-------------------------|------|
| student1 |   c2  |2025-12-01|      12:30-14:49         |(2:19)|
| student2 |   c3  |2025-12-01|      12:12-14:30         |(2:18)|
|-------------------------------------------------------------|
```

5 Arhitectura Proiectului

5.1 Arhitectura Funcțiilor de Bază

5.1.1 Gestionarea Jurnalelor

Scriptul definește o variabilă `SYSLOG` (`/var/log`) ce stochează locația jurnalelor de sistem. Fișierele sunt concatenate într-un singur flux procesat de funcțiile de bază:

```
LOG1="$SYSLOG/auth.log"
LOG2="$SYSLOG/auth.log.1"
LOG3="$SYSLOG/auth.log.2.gz"
# ... (pana la .4.gz)
```

5.1.2 `mylast`

Funcția `mylast` implementează o mașină de stări pentru a urmări sesiunile utilizatorilor.

1. **Detectare Autentificare:** Scanează după "new session".
2. **Deconectare/Eliminare:** Scanează după "removed session" sau "logged out".
3. **Oprire Sistem:** Dacă este detectat "system is powering down", toate sesiunile active sunt închise.

Această mașină de stări este construită pe baza a două variabile:

Queue (Coada): Salvează utilizatorii autentificați care nu au fost încă deconectați.
Session (Sesiunea): Salvează istoricul complet al utilizatorilor deconectați.

Scriptul parcurge jurnalele linie cu linie. Un utilizator nou este adăugat în *Queue*. La detectarea deconectării, acesta este mutat din *Queue* în *Session*.

5.1.3 `mylastb`

Funcția `mylastb` procesează încercările eșuate. Funcționalitatea este asigurată de filtrarea după șirul: "password check failed".

5.2 Arhitectura Funcțiilor Ajutătoare

5.2.1 `get_duration`

Calculează diferența dintre timpul de început (T_s) și timpul de sfârșit (T_e). Aceasta este esențială pentru formatarea duratei sesiunii în formatul `(H:MM)`.