Proiect LazyWGET

Mitrache Antonie Daniel

1. Descrierea problemei

Proiectul LazyWGET presupune dezvoltarea unui script shell care emuleaza comportamentul comenzii ‘wget -r’ dar intr-un mod ‘lenes’ (lazy evaluation). Astfel, descarcarea recursiva se realizeaza treptat, fiind controlata de apeluri explicite ale utilizatorului.

Script-ul este initial apelat prin comanda ‘lwget’, iar promisiunile sunt procesate la fiecare nivel de comanda ‘lget’. In plus, dupa fiecare nivel utilizatorul poate verifica rezultate folosind comenzi precum ‘ls’ sau ‘tree’.

2. Rulare + flag – uri

Scriptul se ruleaza initial cu ./lwget.sh <optiuni 1> <URL>, iar apelurile ulterioare se fac cu ./lget.sh <optiuni 2>

Flag-urile pentru primul apel (optiuni 1) sunt:

- **--dir <director>** : Va salva continutul scriptului in directorul specificat. Daca nu este nimic specificat se va crea un director default “LWGET_CONTENTS”.
- **--absolute** : Va salva doar link-urile absolute (le va ignora pe cele relative)

Flag-urile pentru al doilea apel (optiuni 2) sunt:

- **--dir <director>** : Va opera in directorul specificat, fiind astfel posibila descarcarea mai multor site-uri in acelasi timp, fiecare la un alt nivel de recursivitate. Daca nu este nimic specificat va opera in “LWGET_CONTENTS”.
- **--absolute** : Va salva doar link-urile absolute (le va ignora pe cele relative)
- **--show-err** : Va crea un fisier aditional care va salva link-urile pe care comanda “wget” nu a putut sa le descarce.

3. Modul general de functionare a script-ului:

Comanda “lwget”:

- Testeaza daca a primit un numar corect de parametrii
- Activeaza flag-urile primite
- Daca link-ul introdus nu contine protocolul il completeaza automat (pentru a putea completa link-uri relative)

- Creeaza doua fisiere: “new_links” si “processed_link” cu urmatoarele roluri:
    - “new_links” salveaza link-urile gasite in pagina descarcata
    - “processed_links” salveaza link-urile deja descarcate
- Descarca folosind “wget” continutul site-ului dat ca parametru
- Folosind “grep” si expresii regulate gaseste toate link-urile din continutul site-ului si le salveaza in “new_links”, completand link-urile relative unde e cazul

Comanda “lget”:

- Activeaza flag-urile primite
- Verifica daca a primit un director valid in care sa opereze si daca are fisierul cu link-urile care trebuie procesate
- Citeste rand cu rand din “new_links” si efectueaza urmatoarele operatii:
    - Descarca continutul cu “wget”
    - Verifica daca s-a descarcat cu succes, iar daca nu pune link-ul in fisierul “error_links” (daca flag-ul este activat)
    - Daca s-a descarcat cu succes, verifica daca fisierul, este gol, iar daca este gol il sterge
    - Cauta in continutul descarcat in acelasi mod ca “lwget” link-urile noi si le adauga intr-un fisier temporar
- Urmeaza popularea fisierului “new_links”:
    - Sterge continutul fisierului
    - Parcurge link-urile adaugate in fisierul temporar si le cauta in “processed_links”. Daca nu le gaseste, le adauga in “new_links”, daca le gaseste le ignora (continutul a fost deja descarcat)
    - Sterge fisierul temporar

Pe scurt:

lwget : Creare director -> Descarcare initiala -> Populare “new_links”

lget: Citire “new_links” -> Descarcare continut -> Actualizare “new_links”+ “processed_links”

4. Structura directorului creat in urma apelurilor:

```
nume_director_introdus/
├── downloads/
│   ├── pagina1.html
│   ├── pagina2.html
├── new_links
├── processed_links
└── error_links (va fi creat doar la activarea flag-ului specific)
```

5. Tehnologii si intrumente utilizate

- **Bash Scripting** – Pentru automatizare si gestionarea fisierelor
- **Wget** – Pentru descarcarea paginilor web
- **Grep** – Pentru extragerea linkurilor din fisierele HTML
- **Expresii Regulate** – Pentru gasirea si potrivirea linkurilor

6. Dificultati intampinate si solutii

- Identificarea si completarea linkurilor relative
  - **Problema**: Unele linkuri (cele relative) trebuiau completate pe baza URL-ului original pentru a putea fi descarcate ulterior
  - **Solutie**: Pentru fiecare pagina am salvat URL-ul de baza intr-o variabila pe care am concatenat-o la inceputul fiecarui link relativ
- Management-ul eficient al linkurilor descarcate
  - **Problema**: Necesitatea de a evita linkurile deja descarcate
  - **Solutie**: Am utilizat fisierul 'processed_links' pentru a stoca linkurile procesate, el fiind actualizat dupa fiecare nivel de descarcari
- Gestionarea diferitelor tipuri de protocoale (http/https)
  - **Problema**: Utilizatorul poate furniza ca input un URL fara protocol, iar protocolul este necesar in completarea linkurilor relative
  - **Solutie**: Am adaugat o verificare pentru a detecta linkurile fara protocol si script-ul le completeaza automat prin folosirea expresiilor regulate in combinatie cu redirectionarea error outputului (stderr) al comenzii 'wget' la output.

7. Rezultate experimentale

Am testat script-ul pe mai multe site-uri pentru a evalua functionalitate si eficienta:

- https://www.example.com – Structura simpla, multe link-uri relative
- https://www.emag.ro – Site complex, cu link-uri protejate

- https://www.wikipedia.org – Structura bogata, link-uri extensive

Rezultate:

- Descarcarea si extragerea linkurilor functioneaza conform asteptarilor atat pentru pagini simple, cat si cele complexe
- Linkurile relative au fost completate corect in 100% din cazuri
- Timpul de procesare creste exponential la fiecare nivel de descarcare (poate dura ~2 minute pentru 100 de linkuri)

Limitari observate:

- Descarcarile pot fi blocate de anumite site-uri care nu permit asta
- Script-ul nu gestioneaza corect anumite pagini care folosesc JavaScript pentru generarea linkurilor