

# TP évalué: scripting Shell

Hugo Blanc - Université Lyon 1  
LP ESSIR 2024-2025

November 04, 2024

Le TP est à rendre obligatoirement avant le **dimanche 15 décembre 2024 à 23h59** (Europe/  
Paris).

Attention au respect des consignes de rendu (voir Chapitre 4).

## Table des matières

1. Énoncé .....	4
2. Fichier de logs .....	5
3. Usage et sortie du script .....	6
3.1. Agrégation des logs .....	6
3.2. Analyse temporelle des logs .....	6
4. Rendu .....	7
5. Critères de notation .....	8
6. Annexes .....	9
6.1. Code d'affichage .....	9
7. Licence .....	10

# 1. ÉNONCÉ

Le but du TP est d'écrire un script shell permettant de parser un fichier de logs et d'analyser son contenu.

Le script devra être capable de:

- **parser et agréger les données des logs:** il doit compter le nombre d'occurrence de chaque niveau de log (trace, debug, info, warn, error, et fatal) et identifier les messages les plus et moins communs.
- **proposer une analyse temporelle:** indiquer quelles sont les heures et jours où le système présente le plus d'activité et identifier les heures avec le plus grand nombre d'erreurs.

## **i** Note

Une « erreur » est un message de log dont le log level est « error » ou « fatal ». Tous les autres sont des messages d'information.

Si aucune commande n'est passée au script, il devra afficher une page d'aide expliquant comment l'utiliser.

## 2. FICHIER DE LOGS

Le fichier de log est à télécharger ici: <https://syscall.cafe/t/><sup>o</sup>

Les lignes de log suivent le format suivant:

```
[log level] YYYY-MM-DDZh:mm:ss <msg=text>
```

Par exemple:

```
[trace] 2024-11-04T15:33:06.226302197+01:00 msg="backup completed"
```

## 3. USAGE ET SORTIE DU SCRIPT

### 3.1. AGRÉGATION DES LOGS

- Commande:

```
$ ./log_analyzer.sh aggregate logfile.log
```

- Sortie:

```
== Aggregating file "logfile.log" ==  
  
Log level counts:  
- trace: 144  
- debug: 90  
- info: 77  
- warn: 32  
- error: 22  
- fatal: 0  
  
Most common message: "user logged in" (count: 54)  
Least common message: "backup failed" (count: 1)  
  
== End of report ==
```

### 3.2. ANALYSE TEMPORELLE DES LOGS

- Commande:

```
$ ./log_analyzer.sh temporal_analysis logfile.log
```

- Sortie:

```
== "logfile.log" temporal analysis ==  
  
Most active day: monday  
Most active hour: 18h  
Most error-prone hour: 9h  
  
== End of report ==
```

Où `error-prone` est calculé pour tous les logs dont le niveau de log est soit « error », soit « fatal ».

#### Avertissement

Il faudra **impérativement** respecter le format de sortie des scripts, car vos réponses seront testées automatiquement. Afin de vous faciliter la tâche, les fonctions d'affichage sont données en annexe.

## 4. RENDU

- Une archive **au format zip** est attendue, contenant votre script et tout autre fichier additionnel qui fera l'objet d'une notation (documentation au format Markdown, dépôt Git, ...).
- Le nom de l'archive devra **impérativement** respecter la convention de nommage suivante: `NOM_PRENOM_SHELL24.zip`.
- Cette archive devra m'être envoyée par mail. Le sujet du mail devra suivre la convention de nommage suivante: `[ESSIR] Rendu Shell 2024 NOM PRENOM`.
- La date de rendu maximale est précisée en début de document.

### Avertissement

- Veillez à respecter la casse des conventions de nommage.
- Tout rendu ne respectant pas **tous** ces points **ne sera pas évalué**.
- En cas de retard, un malus de -1 point par heure de retard sera appliqué à la note finale (heure de réception du mail faisant foi).

## 5. CRITÈRES DE NOTATION

La notation se fera selon le barème suivant:

CRITÈRE	DÉTAILS	BARÈME
Pas d'erreurs lors du lancement du script	Le script se lance sans erreurs avec les commandes <code>./log_analyzer.sh</code> et <code>bash log_analyzer.sh</code> .	2
Respect des consignes	Les fonctions d'analyse demandées sont implémentées et le format de sortie est respecté.	5
Programmation idiomatique	Le script est programmé de manière idiomatique.	4
Commentaires et lisibilité	Le script est lisible et commenté.	4
Documentation	Le script est rendu avec un fichier <code>README.md</code> contenant la documentation du projet. Ce fichier est écrit en Markdown.	2
L'étudiant.e a fait preuve d'initiatives	Couleurs, nouvelles fonctions ou commandes, formats de sortie (par exemple CSV)... Mais attention à ne pas trop en faire: KISS <sup>1</sup>	3
Bonus: dépôt Git	Le TP est rendu avec un dépôt Git correctement utilisé.	2
	<b>TOTAL</b>	<b>22</b>

### ✓ Bonus

Deux points bonus peuvent être ajoutés si le TP est rendu avec un dépôt Git correctement utilisé (commits atomiques, *conventional commits messages*, ...). En cas de note maximale, les points seront ajoutés à une autre évaluation.

<sup>1</sup>[https://fr.wikipedia.org/wiki/Principe\\_KISS](https://fr.wikipedia.org/wiki/Principe_KISS)<sup>o</sup>

## 6. ANNEXES

### 6.1. CODE D’AFFICHAGE

Veillez à scrupuleusement respecter l’affichage.

- Agrégation:

```
echo "--= Aggregating file $filename$ $timeframe ==="
echo ""
echo "Log level counts:"
echo " - trace: $trace_count"
echo " - debug: $debug_count"
echo " - info: $info_count"
echo " - warn: $warn_count"
echo " - error: $error_count"
echo " - fatal: $fatal_count"
echo ""
echo "Most common message: \"$most_common_msg\" (count:
$most_common_msg_count)"
echo "Least common message: \"$least_common_msg\" (count:
$least_common_msg_count)"
echo ""
echo "--= End of report ==="
```

- Analyse temporelle:

```
echo "--= $filename temporal analysis $timeframe ==="
echo ""
echo "Most active day: $most_active_day"
echo "Most active hour: $most_active_hour"
echo "Most error-prone hour: $most_errors_hour"
echo ""
echo "--= End of report ==="
```

## 7. LICENCE

© Hugo Blanc, 2024

Ce document peut être distribué librement, selon les termes de la version 4.0 de la licence Creative Commons Attribution-ShareAlike: <http://creativecommons.org/licenses/by-sa/4.0/>.

Vous êtes libres de :

- reproduire, distribuer et communiquer ce document au public;
- modifier ce document.

Selon les conditions suivantes :

- **Paternité.** Vous devez citer le nom de l'auteur original.
- **Partage des Conditions Initiales à l'Identique.** Si vous modifiez, transformez ou adaptez cette création, vous n'avez le droit de distribuer la création qui en résulte que sous un contrat identique à celui-ci.
- A chaque réutilisation ou distribution, vous devez faire apparaître clairement aux autres les conditions contractuelles de mise à disposition de cette création.
- Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits.