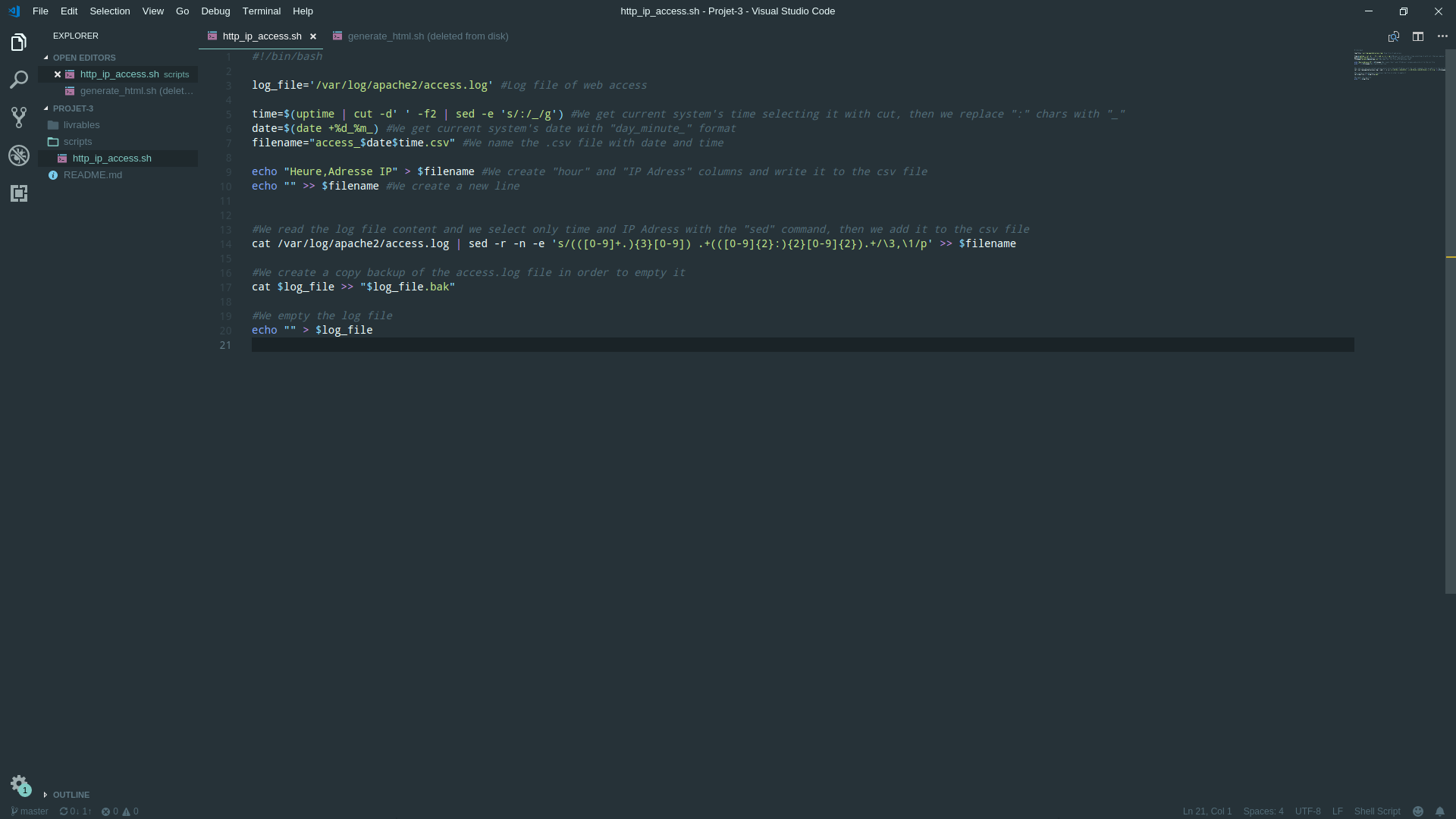
# Script http\_ip\_access.sh



Ce script a pour but de récupérer les entrées du fichier log d’Apache 2, notamment les adresses IP de connexion et les heures de connexion.

On utilise en tout 4 variables:

* Log\_file: Chemin d’accès au fichier de log
* Time: temps actuel de la machine
* Date: date actuelle de la machine
* Filename: nommage du fichier .csv

Le temps est récupéré avec la commande uptime associé avec cut et sed. Cut permet de récupérer le champ correspondant au temps et sed permet de remplacer le caractère “:” par des “\_”

La date est récupérée avec la commande date et est formaté grâce aux arguments +%d\_%m\_

Filename est nommé avec le préfixe “access\_” suivi de la date et du temps, avec l’extension .csv

On commence par créer les colonnes Heure et Adresse IP en écrivant la sortie de la commande echo dans la variable filename.

On crée ensuite une ligne vide de la même manière que la commande précédente, à l’exception que cette fois on l’ajoute à la fin du fichier

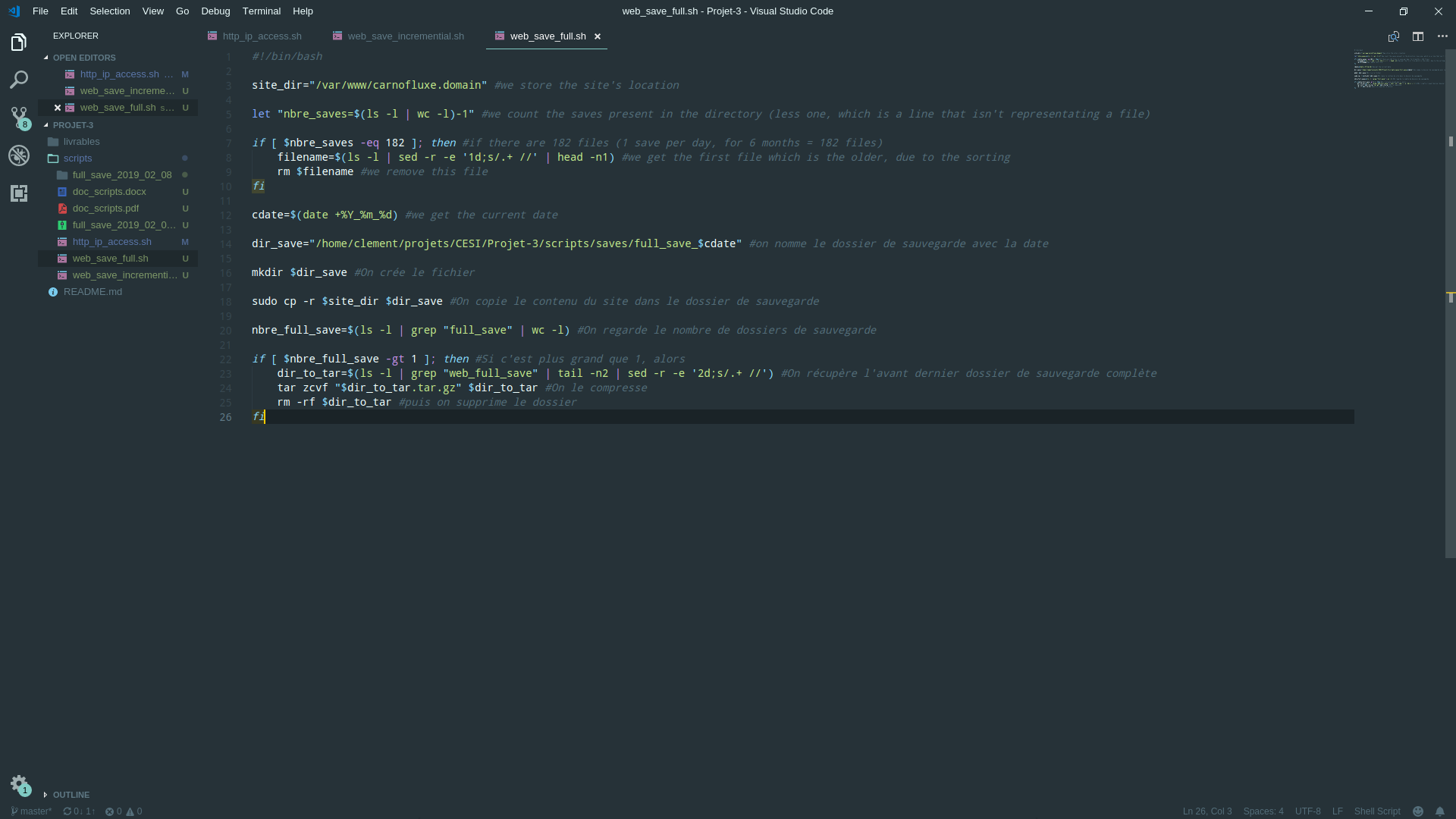
Ensuite, nous récupérons toutes les entrées du fichier .log puis utilisons sed pour sélectionner les valeurs voulues et mettre en forme la sortie.

On récupère d’abord les adresses IP situées au début de chaque ligne et les mettons en mémoire grâce aux parenthèses, puis nous mettons un point avec un + afin d’indiquer que nous prenons tous les caractères suivant, puis nous écrivons le regex de l’heure, entre parenthèses pour une mise en mémoire, et enfin nous sélectionnons le reste pour le supprimer.

Après le slash (/) nous affichons le 3ème champ en mémoire, correspondant donc à l’heure, nous le séparons d’une virgule pour la séparation CSV puis nous écrivons le premier champ en mémoire, c’est à dire l’adresse IP.

Enfin, nous créons un backup du fichier log puis nous le vidons.

# Script web\_save\_full.sh



Ce script a pour but de créer une sauvegarde complète du site web en prenant le soin de vérifier et supprimer le fichier datant de plus de 6 mois.

On utilise également ici 7 variables:

* Site\_dir: chemin jusqu’au dossier contenant le site web
* Nbre\_saves: on récupère le nombre de fichiers de sauvegarde
* Filename: on récupère l’archive la plus ancienne
* Cdate: on récupère la date actuelle du système
* Dir\_save: On nomme le dossier de sauvegarde complète à créer
* Nbre\_full\_save: on récupère le nombre de fichiers de sauvegarde
* Dir\_to\_tar: on nomme l’avant dossier à archiver

On utilise let afin d’exécuter une opération arithmétique (ici, une soustraction) entre le résultat de la commande ls -l | wc -l et le chiffre 1, qui va servir à enlever la première ligne du résultat qui ne symbolise pas un fichier.

Ensuite, nous vérifions si le nombre de fichiers est égal à 182, qui symbolise le nombre de jours qu’il y a dans 6 mois.

Si c’est e cas, on récupère le nom du fichier à supprimer qui se trouve être le premier affiché, grâce au tri naturel du shell.

Pour cela, nous listons les fichiers présents, nous supprimons la première ligne et sélectionnons uniquement le nom absolu du fichier avec sed, puis nous cachons les autres avec head. On le supprime alors.

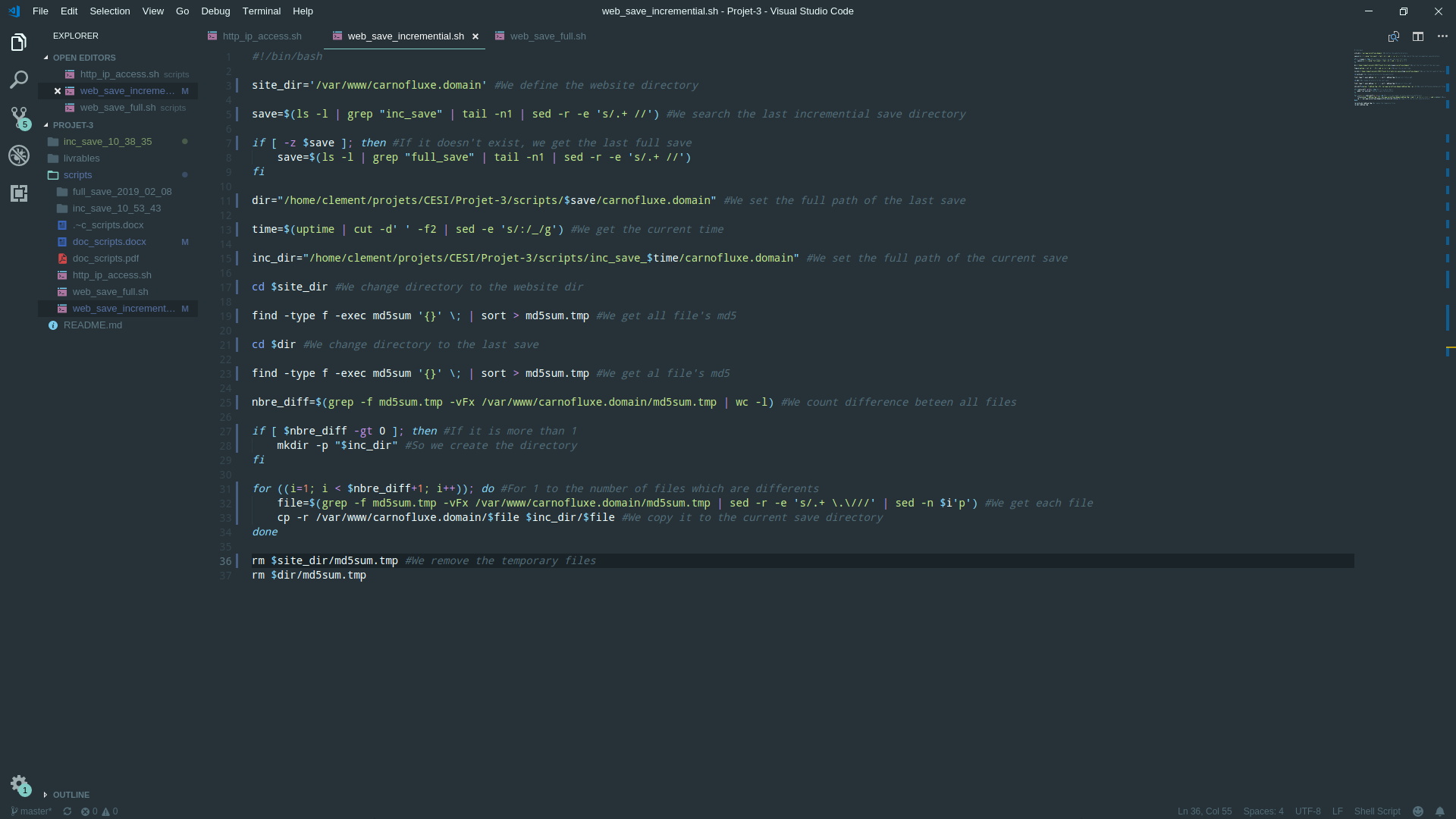
Ensuite, nous récupérons la date actuelle de la machine, puis créons le dossier de sauvegarde, qui sera utile pour les sauvegardes incrémentielles

On le créer et copions le contenu du site à l’intérieur, puis nous récupérons le nombre de dossier de sauvegarde

Enfin, nous archivons et compressons le fichier avec tar.

Enfin, nous supprimons le répertoire qui n’est plus utile

# Script web\_save\_incremential.sh



Ce script a pour but de créer une sauvegarde incrémentielle du site web en prenant comme référence la dernière sauvegarde incrémentielle ou la dernière sauvegarde complète selon sa disponibilité.

On utilise ici 7 variables:

* Site\_dir: chemin vers le dossier du site web
* Save: nom du dossier de la dernière sauvegarde
* Dir: chemin complet vers la dernière sauvegarde du site
* Time: temps du système actuel
* Inc\_dir: chemin complet de la future sauvegarde
* Nbre\_diff: nombre de fichiers différents entre le site actuel et la dernière sauvegarde
* File: Fichier à sauvegarder

Au début de ce script, nous cherchons la dernière sauvegarde incrémentielle. Si elle existe, nous la prenons comme référence. Dans le cas contraire, nous prenons la dernière sauvegarde complète comme référence.

Ensuite, nous définissons le chemin complet vers cette sauvegarde.

Nous nommons ensuite le dossier de la future sauvegarde avec l’heure actuelle.

Nous cherchons ensuite tous les fichiers du site web et récupérons leur hash md5, les trions et les mettons dans un fichier temporaire. De même pour les fichiers de la dernière sauvegarde

Nous cherchons ensuite le nombre de fichiers différents entre ces 2 répertoires, puis, si la différence est supérieure à 0, nous créons le dossier de la sauvegarde incrémentielle.

Ensuite, pour chaque fichier, nous le copions depuis le site web et le mettons dans le dossier de sauvegarde actuelle.

Enfin, nous supprimons les fichiers temporaire.