

Guide pour l'audit et le nettoyage de serveurs NAS

Xavier Baume, 19.11.2025

Introduction à l'application

Les quelques scripts/fichiers présentés ci-dessous visent à automatiser la création d’empreintes numériques (ou hashes) sur des fichiers et dossiers d’archives, afin :

- d’identifier les doublons de fichiers,
- de générer un rapport CSV et une visualisation arborescente,
- d’automatiser la suppression de fichiers à partir d’un fichier JSON généré depuis la visualisation.

⚠ Le fonctionnement repose sur des fichiers macOS-friendly, exécutés via des fichiers .command (double-cliquables). Les autres systèmes d’exploitation (Windows, Linux) ne sont pas pris en charge.

Contenu du répertoire

Pour se rendre vers le contenu du répertoire, suivre le chemin absolu suivant sur le NAS:

/Volumes/UNIRIS/Scripts_archives/audit_nas/app_audit_nas

Fichier	Rôle	Détails
1_app_hashing.command	Fichier pour le calcul des empreintes sur un ou plusieurs dossiers racines.	Appelle le script hashes_scans.sh avec les bons paramètres.
2_app_delete.command	Fichier pour la suppression des fichiers après analyse.	Appelle delete_from_json.py et supprime les fichiers listés dans un fichier de rapport .json
hashes_scans.sh	Script principal de hachage et export CSV.	Gère le cur du traitement (hash, exclusion de fichiers, export CSV, empreintes de dossiers).
three_visu.py	Script Python de visualisation ou d’analyse des résultats d’analyse.	Permet une vue graphique de l’arborescence des dossiers/fichiers et des éventuels doublons.
requirements.txt	Fichier texte permettant l’installation de toutes les dépendances nécessaires à l’exécution du script Python.	Contient les versions de modules Python. ⚠ Contrôler l’actualité des versions !
delete_from_json.py	Fichier Python de suppression automatisée basé sur le fichier de rapport .json .	La suppression peut être simulée ou définitive selon l’option choisie.

⚠ Seul le script **.command** doit être utilisé par l’utilisateur ; les autres fichiers ne nécessitent aucune manipulation directe.

⚠ Veillez à ce que les trois fichiers mentionnés ci-dessus se trouvent ensemble dans le même dossier, et qu’aucun autre fichier ne s’y trouve.

Installation initiale

Lors de la première utilisation de **1_app_hashing.command**, il est nécessaire d’installer plusieurs "packages" afin de permettre l’exécution du script de hachage. Il est possible que cette opération dure plusieurs minutes.

- Outre les outils système présents sur l’OS, l’intégralité des packages seront automatiquement installés via **Hombrew**.
- Lors de l’installation, le Terminal vous demandera votre mot de passe utilisateur (**sudo**). Il faudra alors saisir le mot de passe que vous utilisez pour ouvrir votre session (il est normal que le mot de passe ne s’affiche pas lors de son entrée). Si certaines installations sont requises, pressez sur **ENTER** ou ré-entrez votre mot de passe.
 - ==> Checking for ‘sudo’ access (which may request your password)...
 - Password:
 - ==> The Xcode Command Line Tools will be installed.
 - Press RETURN/ENTER to continue or any other key to abort:
- Parfois, le module **gsed** n’est pas installé automatiquement. Pour vérifier son installation, ouvrez un nouveau terminal et saisissez : **brew install gsed**
Le terminal doit indiquer : **Warning: gnu-sed x.x is already installed and up-to-date.**

Une fois ces installations effectuées, cette étape n’est plus requise : vous pouvez passer à la section suivante.

Utilisation du fichier d'audit

1. Lancer le calcul des empreintes
 1. Double-cliquez sur **1_app_hashing.command**.
 - Une fenêtre du Terminal s’ouvre pour assurer que toutes les dépendances nécessaires sont installées.
 2. Sélectionnez le répertoire où s’enregistreront le CSV ainsi que la visualisation HTML. À noter qu’un sous-dossier sera créé dans ce même répertoire

3. Sélectionnez les dossiers à "hasher". Le script retiendra les chemins absolus sélectionnés, par exemple :

```
/Volumes/UNIRIS/_GEDUNIL_A_TRAITER
/Volumes/ULGA/COLLNUM
```

4. Une nouvelle fenêtre du Terminal s’ouvre pour calculer automatiquement (aucune action est nécessaire à cette étape) les empreintes en suivant les 4 étapes suivantes :
 - sélection des fichiers et dossiers à traiter, en excluant certains types ;
 - calcul du hash BLAKE3 pour les fichiers nouveaux ou modifiés ;
 - export CSV des métadonnées
 - calcul des empreintes de dossiers (à partir des fichiers qu’ils contiennent).⚠ Durant l’exécution, l’ordinateur ne doit pas être éteint.
5. Le résultat est enregistré dans un fichier **.csv** et dans un fichier **.html** dont le nom et l’emplacement sont précisés dans le Terminal. Le dossier créé doit contenir les deux éléments suivants:

```
├─ audit_hashes.csv
└─ tree_paths.html

1 directory, 2 files
```

2. Consulter les résultats

1. Via le fichier CSV, dont les colonnes sont indiquées par :
path,type,size_bytes,mtime,hash

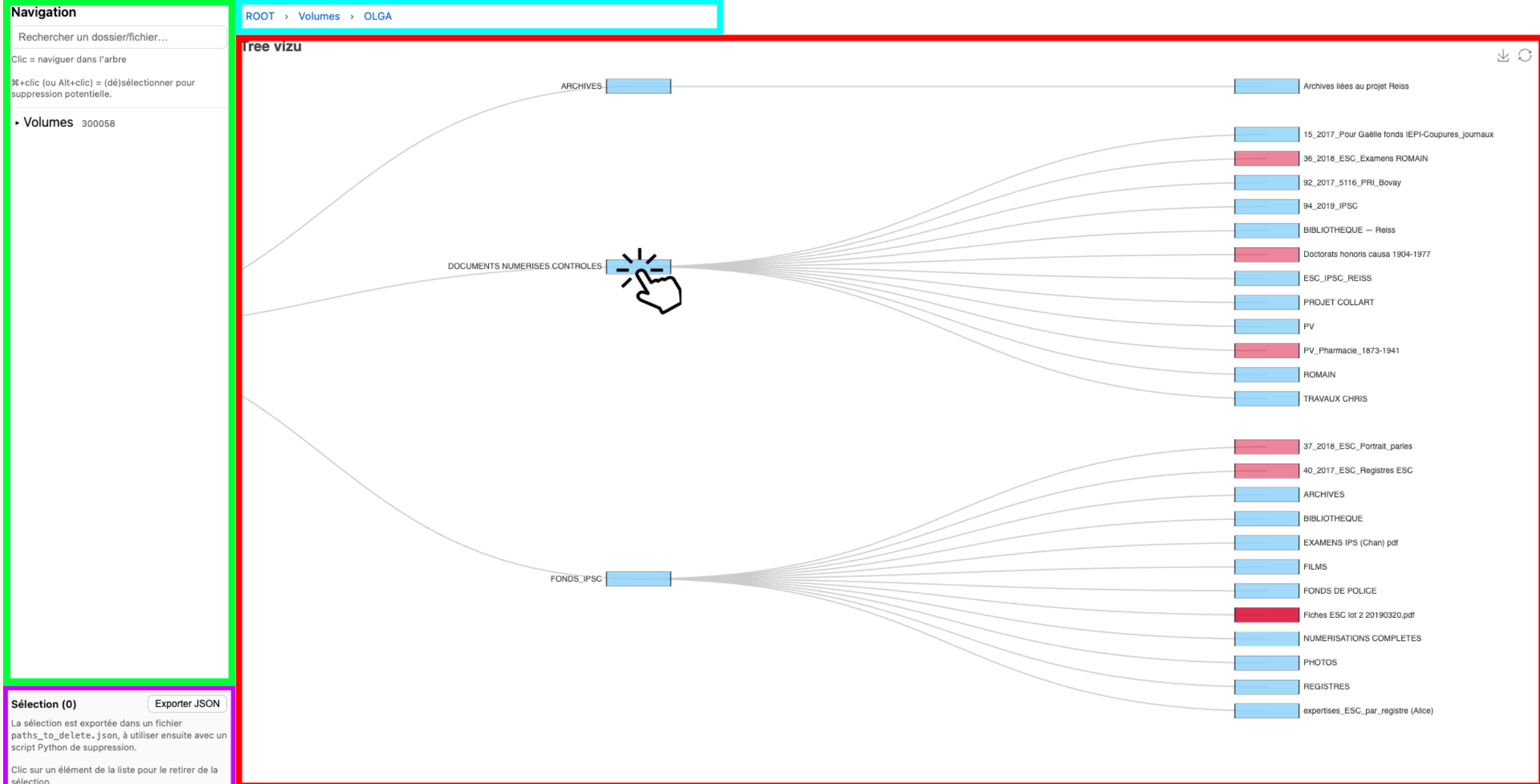
- **path** : chemin complet du fichier ou dossier ;
- **type** : file ou directory ;
- **size_bytes** : taille du fichier en octets ;
- **mtime** : date de dernière modification ;
- **hash** : empreinte numérique unique du fichier/dossier.

2. Via le fichier HTML sur le navigateur de votre choix.

La visualisation se présente sous la forme d’une arborescence horizontale (de gauche à droite)

Elle est répartie en quatre zones principales:

- **Zone 1**: Recherche plein texte
- **Zone 2**: Exploration de l’arborescence multi-niveaux
- **Zone 3**: Indication du chemin absolu
- **Zone 4**: Sélection des dossiers/fichiers à supprimer ultérieurement



La **Zone 2** répertorie quatre codes couleurs selon les niveaux dossier/fichier:

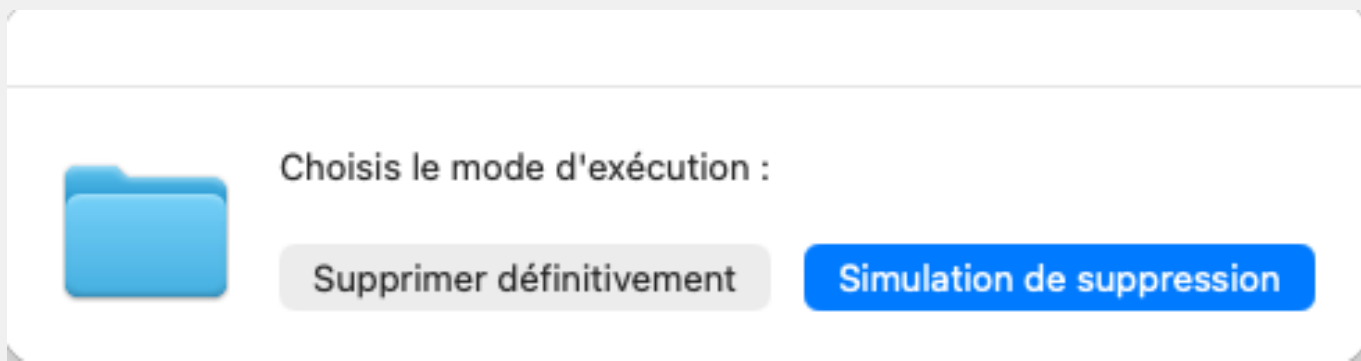
- Dossier : un dossier affiché en bleu clair indique que ses fichiers ou sous-dossiers ne contiennent aucun doublon ; un dossier affiché en rouge clair signale qu’au moins un fichier est un doublon.
- Fichier : un fichier affiché en bleu foncé indique qu’il s’agit d’un élément unique (aucun doublon) ; un fichier affiché en rouge foncé signale qu’il existe en double.

Utilisation du fichier de suppression de fichiers

Les fichiers ou dossiers à supprimer peuvent être sélectionnés directement dans le navigateur (en utilisant Cmd + clic). Ces derniers seront provisoirement listés dans la **Zone 4**. Un clic sur un fichier ou un dossier catalogué permet de le retirer de la liste.

Une fois les dossiers ou fichiers sélectionnés pour la suppression, cliquez sur "Exporter JSON" dans la **Zone 4**.

- Un fichier nommé **paths_to_delete.json** sera téléchargé dans votre navigateur.
- Déplacez ce fichier dans le répertoire préalablement créé, où se trouvent déjà les fichiers **audit_hashes.csv** et **tree_paths.html**
- Rendez-vous dans le répertoire présenté au point "Contenu du répertoire" et cliquez sur **2_app_delete.command**
- Choisissez le fichier **paths_to_delete.json** préalablement déplacé et sélectionnez l’option de simulation ou de suppression définitive
- Le fichier **delete_from_json.py** exécutera la suppression (simulée ou définitive) des chemins présents dans **paths_to_delete.json**

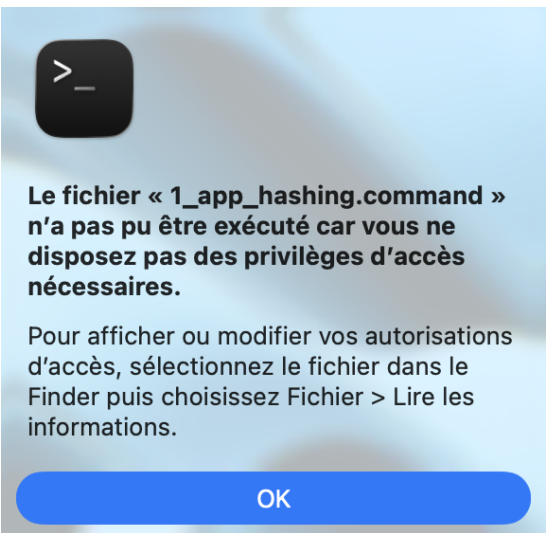


- Une fois l’option choisie, un fichier **.txt** est automatiquement ajouté au dossier et consigne le journal des opérations de simulation et des suppressions définitives.

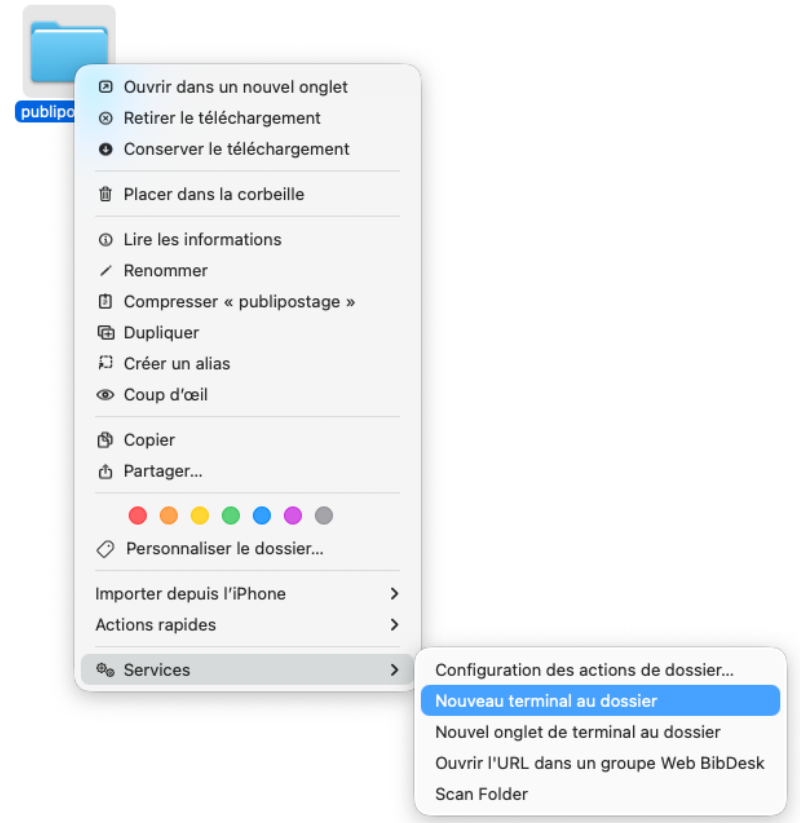
```
├─ audit_hashes.csv
├─ paths_to_delete_delete_log_20251110_162605.txt
├─ paths_to_delete.json
└─ tree_paths.html
```

Remarques complémentaires : problème d'exécution des fichiers

Au lancement du fichier **.command**, le message d’erreur suivant peut s’afficher :



Dans ce cas, ouvrez un terminal dans le dossier où se trouve le fichier **.command**.



Dans le terminal, saisissez la commande suivante : **chmod +x [Nom du fichier]**, puis **ENTER**.

```
xbaume@CENTRE41457 publipostage % chmod +x 1_app_hashing.command
xbaume@CENTRE41457 publipostage %
```

Une fois cette commande exécutée, relancez le fichier **.command** en double-cliquant sur l’icône.