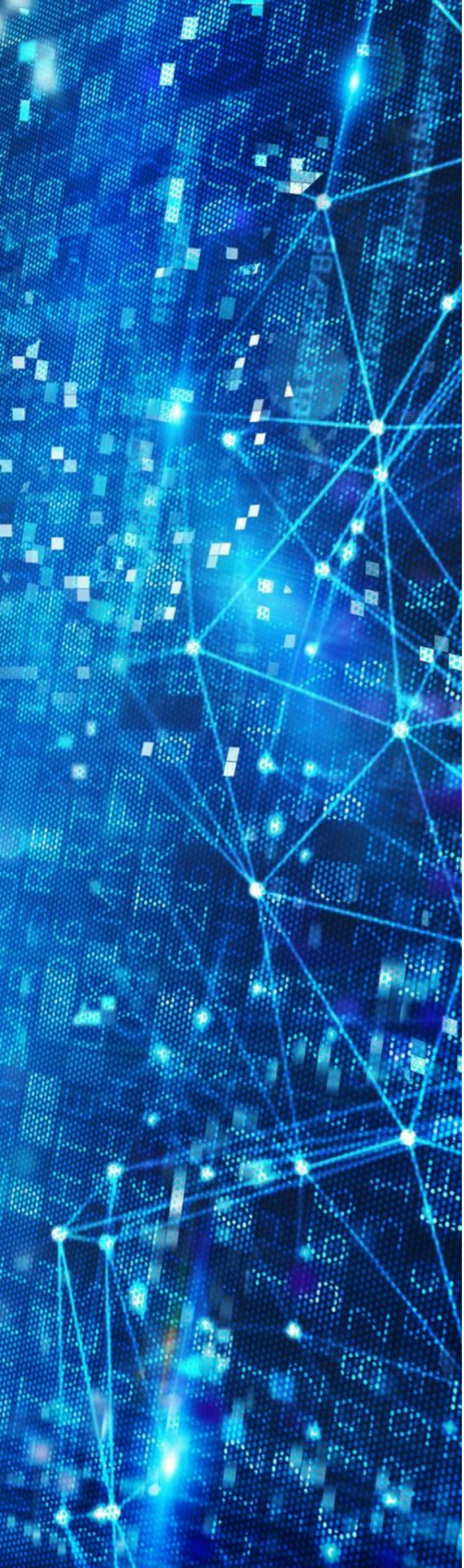


Années : 2022 - 2024

BTS SIO

Fiche Situation numéro 6 :

Mise en place d’un VPN sur un serveur Synology (OpenVPN)

Mots-clés :

Permissions, accès, droits, utilisateurs, groupes, contrôle d’accès, lecture, écriture, exécution, héritage, sécurité, dossiers, fichiers, autorisations, restrictions, privilèges, propriétaire, administrateur, gestion, partage, confidentialité, protection, authentification, identification, propagation, suppression, modification, restrictions d'accès, règles de sécurité, contrôle parental, permissions explicites, permissions implicites, gestion des droits, architecture de sécurité.

Plan de la situation

1. Introduction
2. Problématique rencontré
3. Processus de réalisation
4. Conclusion

1.Introduction

La gestion des permissions d'accès aux fichiers et dossiers est essentielle pour garantir la sécurité et l'organisation des ressources informatiques.

Dans ce cadre, un dossier spécifique a été créé, avec des permissions restreintes à un seul utilisateur, empêchant ainsi les autres d'y accéder. Cette configuration permet de contrôler les accès et d'assurer la confidentialité des données.

2.Problématique rencontrée

Lors de la mise en place des permissions, plusieurs défis ont été observés :

- Vérification des droits d'accès effectifs des utilisateurs.
- Gestion des permissions héritées pouvant interférer avec les règles définies.
- Configuration correcte des autorisations NTFS ou autres systèmes de fichiers.
- Tests de validation pour s'assurer que seuls les utilisateurs autorisés accèdent au dossier.

3. Processus de réalisation

3.1 Création du dossier : Un dossier nommé "Dossier_Sécurisé" a été créé à un emplacement défini

3.2 Création des utilisateurs : Plusieurs comptes utilisateurs ont été générés sur le système.

3.3 Attribution des permissions :

- Suppression des permissions héritées pour éviter des accès non désirés.
- Attribution des droits d'accès uniquement à l'utilisateur sélectionné (lecture, écriture, modification).
- Vérification et application des permissions via l'interface graphique ou les commandes en ligne (ex : icacls sous Windows, chmod/chown sous Linux).

3.4 Tests et validation :

- Connexion avec l'utilisateur autorisé pour vérifier l'accès.
- Connexion avec un utilisateur non autorisé pour tester la restriction d'accès.
- Ajustements éventuels pour affiner les permissions.

3.2 Installation et configuration :

- Processus de Réalisation

1.Création du dossier :
- Exécution de la commande : mkdir /home/utilisateur/Dossier_Sécurisé

2.Création des utilisateurs :
- Ajout de nouveaux utilisateurs : (sudo adduser utilisateur1, sudo adduser utilisateur2)

3. Attribution des permissions :

- Modification du propriétaire du dossier pour l'utilisateur spécifique :

sudo chown utilisateur1:utilisateur1 /home/utilisateur/Dossier_Sécurisé

- Suppression des permissions d'accès pour les autres utilisateurs :

sudo chmod 700 /home/utilisateur/Dossier_Sécurisé

7 : Lecture, écriture et exécution pour le propriétaire.
0 : Aucun accès pour le groupe et les autres utilisateurs.

- Vérification et test des permissions :

Se connecter avec utilisateur1 et tenter d'accéder au dossier :
su - utilisateur1
cd /home/utilisateur/Dossier_Sécurisé

- Se connecter avec utilisateur2 et vérifier l'interdiction d'accès :

su - utilisateur2
cd /home/utilisateur/Dossier_Sécurisé

Résultat attendu : "Permission denied"

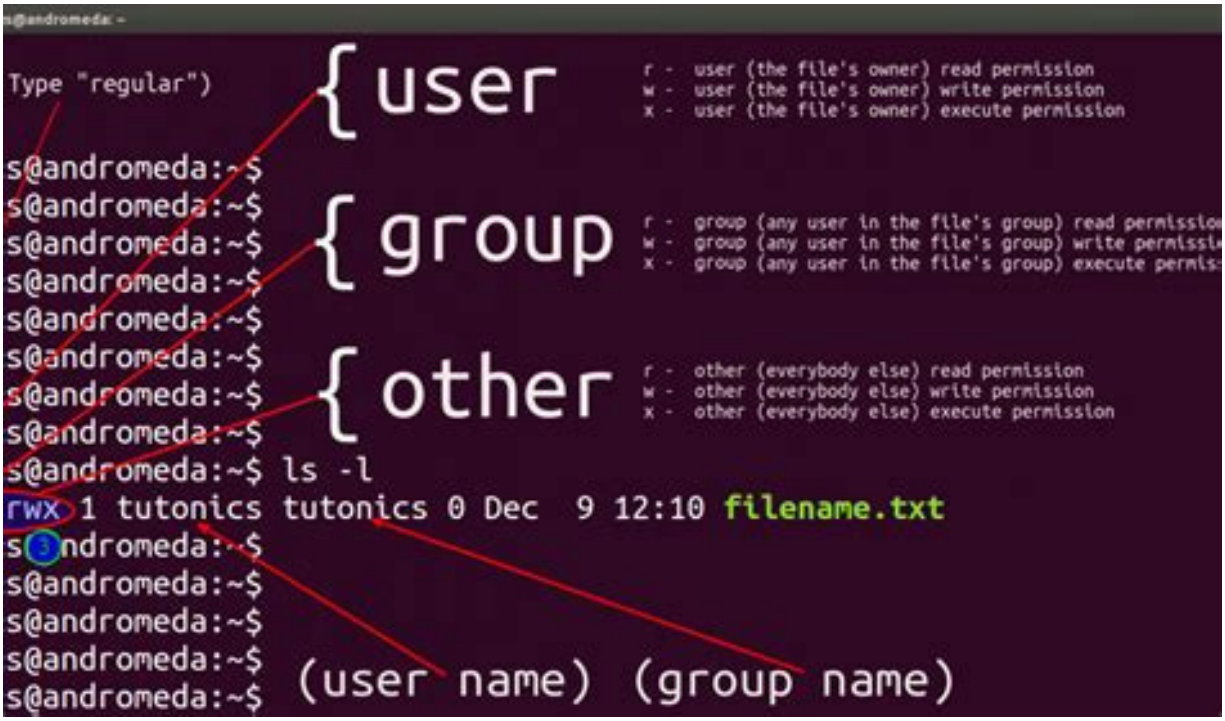
- Vérification des permissions du dossier avec :

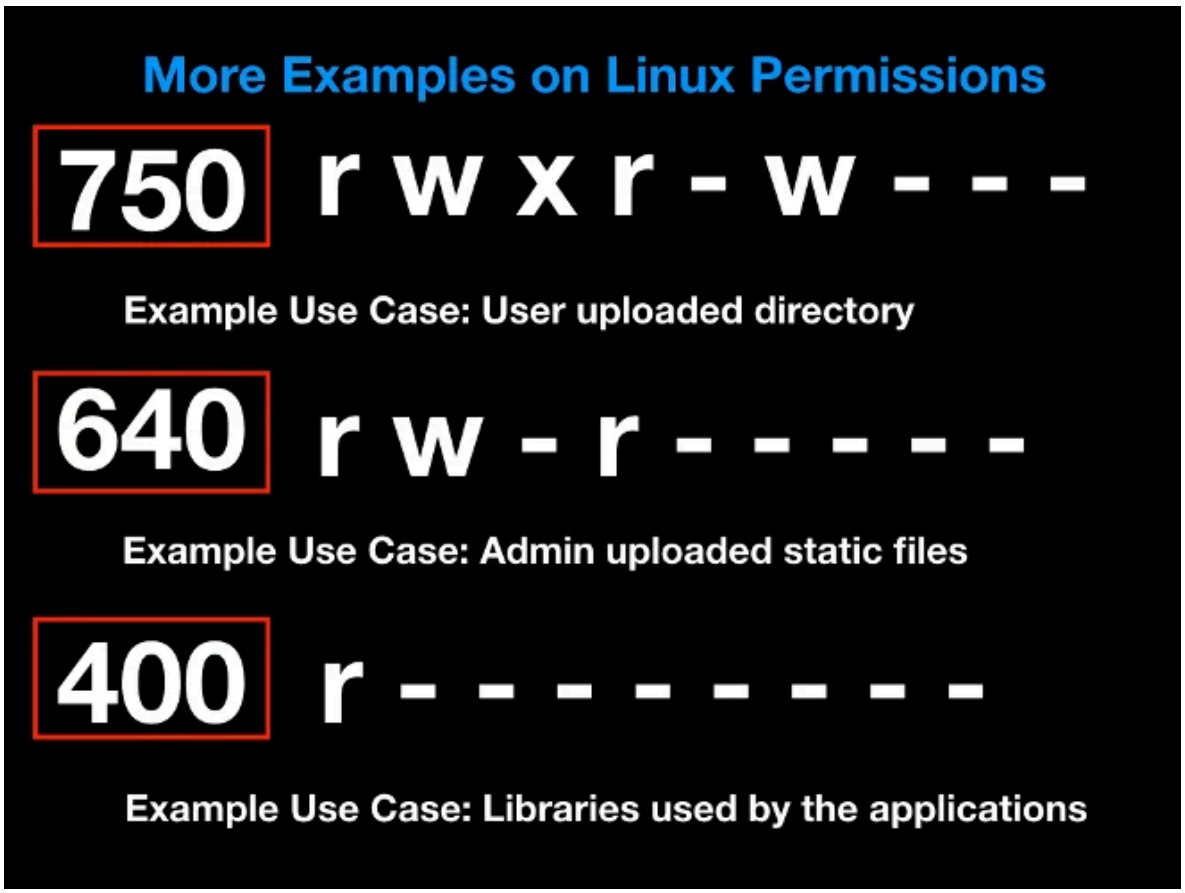
ls -ld /home/utilisateur/Dossier_Sécurisé

4. Conclusion

Cette mise en place permet de sécuriser efficacement les accès aux données en limitant les droits aux seuls utilisateurs nécessaires. Malgré quelques difficultés liées aux permissions par défaut et aux tests de validation, la configuration a été finalisée avec succès. Cette méthodologie garantit une gestion optimale des droits d'accès et peut être appliquée à d'autres dossiers nécessitant une restriction des permissions.

Annexe


Type "regular")
user
r - user (the file's owner) read permission
w - user (the file's owner) write permission
x - user (the file's owner) execute permission
group
r - group (any user in the file's group) read permissio
w - group (any user in the file's group) write permissi
x - group (any user in the file's group) execute permis
other
r - other (everybody else) read permission
w - other (everybody else) write permission
x - other (everybody else) execute permission
s@andromeda:~$ ls -l
rwx 1 tutonics tutonics 0 Dec 9 12:10 filename.txt
(user name) (group name)


More Examples on Linux Permissions
750 r w x r - w - - -
Example Use Case: User uploaded directory
640 r w - r - - - - -
Example Use Case: Admin uploaded static files
400 r - - - - - - - -
Example Use Case: Libraries used by the applications