MA – ICR

Mini-Projet

|  |
| --- |
| Shared encrypted file system |

|  |
| --- |
| Étudiant participant à ce travail :  **Benjamin Mouchet, CS**  undefined  Professeur :  **Alexandre Duc**  Restitution du rapport : **12.06.2024**  Semestre : **SP 2024**  École : **HES-SO, Lausanne** |

Table des matières

[Table des matières I](#_Toc167809284)

[1 - Introduction II](#_Toc167809285)

[2 - Architecture III](#_Toc167809286)

[3 - Implémentation IV](#_Toc167809287)

[4 - Améliorations possibles V](#_Toc167809288)

[5 - Conclusion VI](#_Toc167809289)

[6 - Sources VII](#_Toc167809290)

1. Introduction

Une personne utilisant un ordinateur au quotidien saurait dire qu’un gestionnaire de fichiers est un programme pouvant naviguer une structure arborescente composée de dossiers et de fichiers. Un utilisateur plus chevronné pourrait même ajouter que la racine de cet arbre est communément appelée *root* et que cette structure peut être partagée ou protégée pour un utilisateur localement, comme pour les droits d’accès à un fichier sur un système Linux. En revanche, qu’en est-il lorsque l’on souhaite stocker des données sur un serveur ? Comment donner des accès à un fichier ? Comment assurer la sécurité du service ? Toutes ces notions sont importantes à considérer, quel que soit le degré de sensibilité des données stockées, personne ne souhaite voir ses fichiers en lecture libre ou sur une infrastructure vulnérable sur le réseau.

L’objectif de ce projet est de répondre à ces différentes questions, en décrivant les concepts cryptographiques et l’architecture d’un service de stockage de fichiers chiffrés en ligne, aussi appelé cloud, ainsi que certains détails pertinents d’implémentation. Les notions de réseau ne seront pas abordées en détails dans ce rapport.

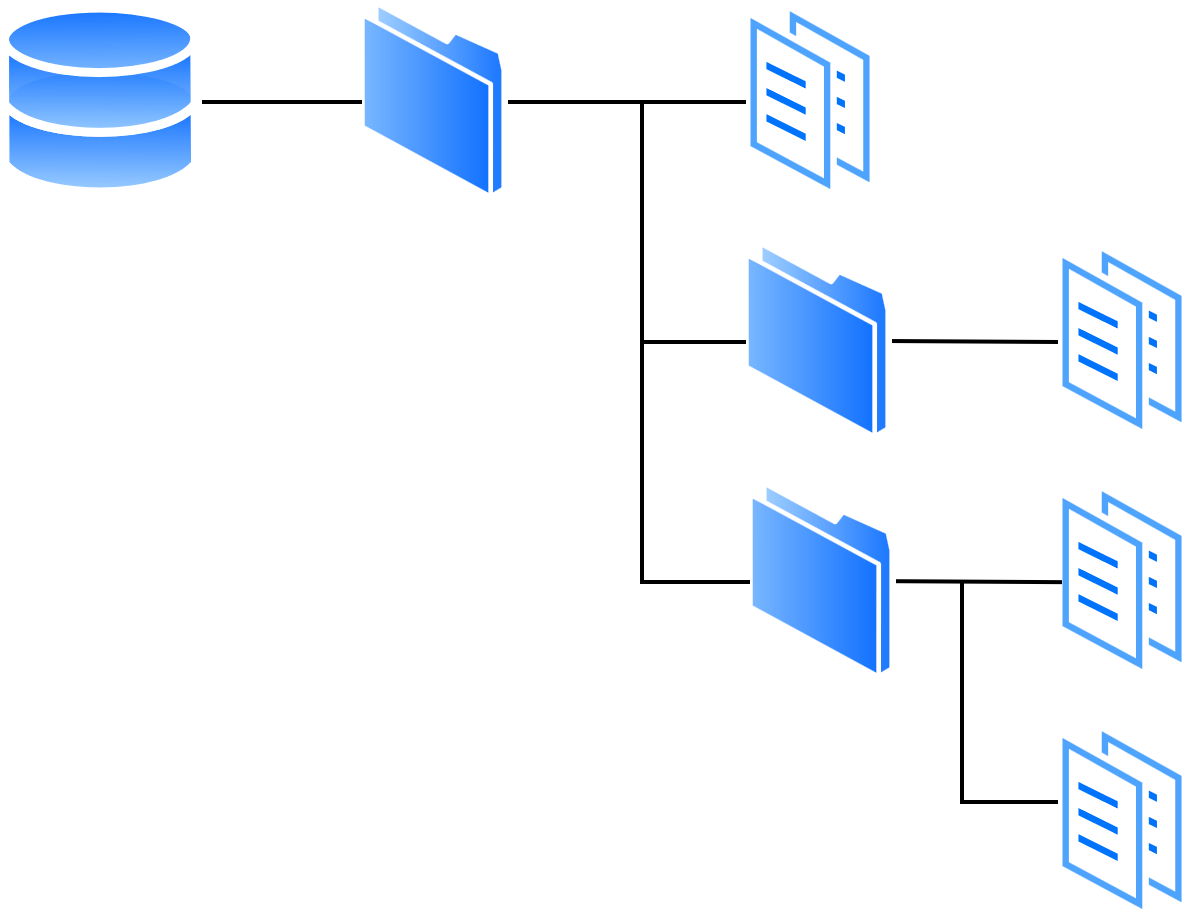
Ce projet repose grandement sur les notions vues durant le cours d’Industrial Cryptography ainsi que le White Paper publié en 2022 par Mega[1], service de partage de fichier dans le cloud garantissant de la *End to End Encryption* (abrégé E2EE).

Figure 1 : Représentation d’un système de fichiers

1. Architecture

Pour la modélisation du serveur, nous pouvons partir sur un schéma classique où le serveur contient une userbase et une database pour stocker les fichiers. Les utilisateurs ont chacun leur root folder (basé sur leur username, qui est unique). Comme nous nous préparons contre des adversaires actifs et que nous voulons garder le nom et le contenu des fichiers confidentiels, il faut stocker ces données chiffrées. De plus, les utilisateurs doivent pouvoir se login facilement, une seule fois et depuis n’importe quelle machine. Pour finir, les dossiers (et sous-dossiers) doivent pouvoir être partagés à d’autres utilisateurs. Dans les prochains chapitres nous allons détailler comment répondre à tous ces besoins

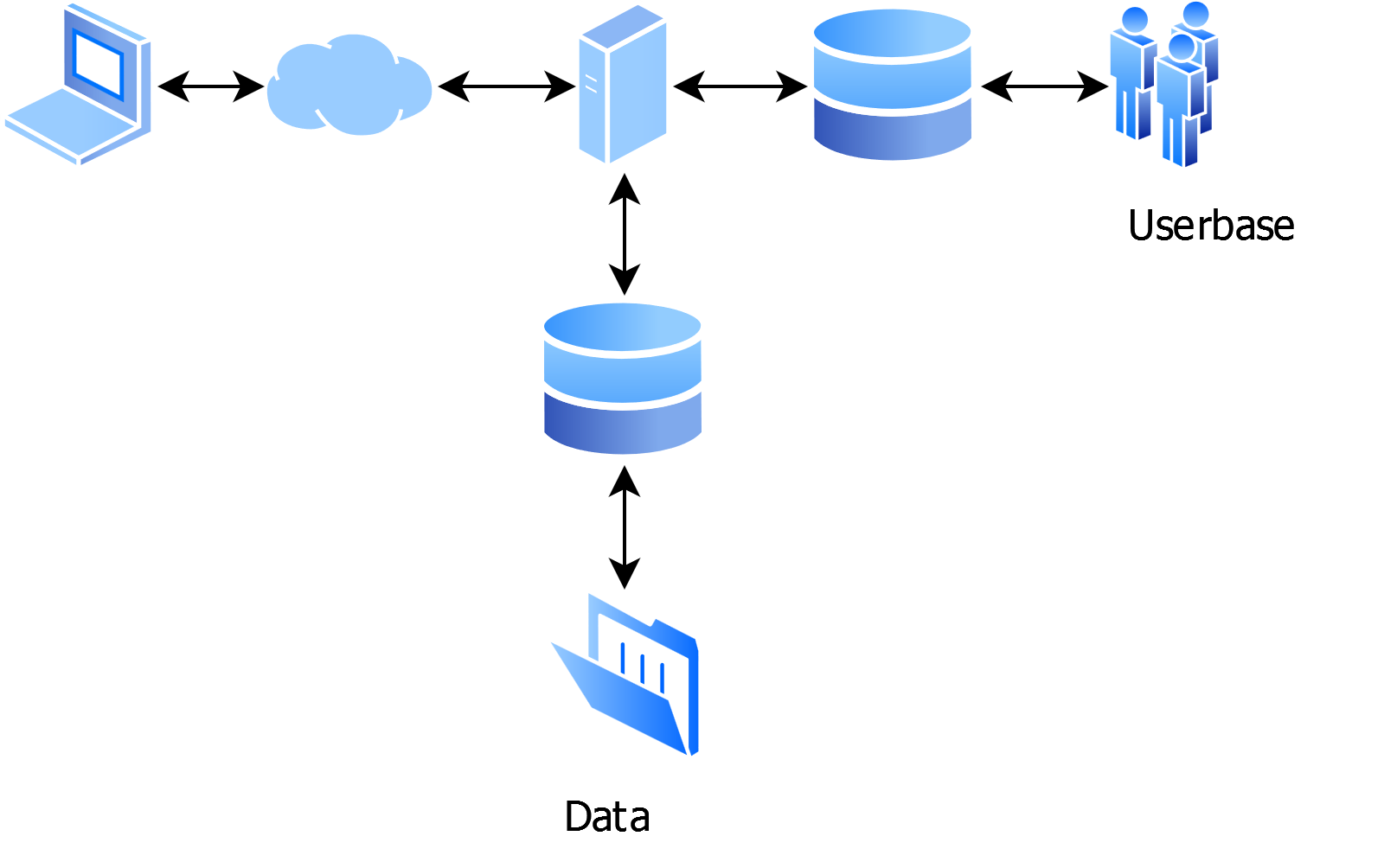


Figure 2 : Modélisation standard d’un serveur

* 1. Le stockage de données confidentielles

Une image contenant diagramme, texte, ligne, capture d’écran

Description générée automatiquementLe fichiers et noms de dossiers ne doivent pas fuiter, leur taille en revanche peut. Afin de garantir cette confidentialité, un chiffrement symétrique est mis en place. Chaque dossier possède une clé et chaque fichier est chiffré avec la clé du dossier parent. Ces clés ne sont pas directement stockées dans les dossiers mais sur une table dans le serveur afin de ne pas avoir besoin de déchiffrer en cascade toute la hiérarchie si l’on souhaite afficher un dossier intermédiaire. Un mapping est réalisé entre le nom du dossier, son nom chiffré (afin de pouvoir le retrouver), son uid et sa clé. Toutes ces données sont également chiffrées grâce à la master key de l’utilisateur.

Figure 3 : Chaque dossier et son contenur est chiffré par sa clé

* 1. Partager les dossiers

Comme les dossiers sont chiffrés de manière symétrique, être en possession de la clé du dossier permet de déverrouiller son contenu. Nous utilisons donc un processus de chiffrement asymétrique et de signature afin de partager à un autre utilisateur l’emplacement sur le serveur et la liste des clés nécessaires pour chaque dossier et sous-dossier. La signature permet de prévenir une attaque man in the middle et qu’un autre utilisateur donne accès à des dossiers potentiellement malicieux.

Une image contenant texte, capture d’écran, Police, diagramme

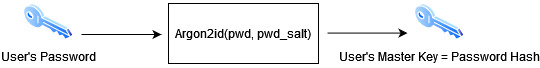
Description générée automatiquement

Figure 4 : Schéma de stockage des informations sur le serveur

* 1. Utilisation par un *User*

Un utilisateur doit commencer le processus par se register auprès du serveur.

1. Implémentation



1. Améliorations possibles
2. Conclusion
3. Sources
4. MEGA, 2022. MEGA Security White Paper. In: *Mega* [en ligne]. Disponible à l’adresse : <https://mega.nz/SecurityWhitepaper.pdf> [Consulté le 28.05.2024]