Guide de Sécurité des Systèmes d'Information

-

Projet Voltron

Auteur : T. de Lachaux
Dernière version : 02.07.2021

I Introduction

L’objet de ce rapport est de présenter les principes élémentaires de sécurité à respecter dans le cadre du projet Voltron. Il est divisé en deux sections :

- La première section s’adresse à l’ensemble des acteurs susceptibles d’utiliser la solution développée par l’équipe Voltron.
- La seconde section s’adresse aux développeurs et administrateurs du projet Voltron.

I Section utilisateurs

1 Mails

- **Ne pas utiliser son adresse mail professionnelle à des fins personnelles.**
- **Ne pas ouvrir les courriers, pièces jointes et liens** dès lors que vous ne connaissez pas l'expéditeur du courrier.

2 Mots de passe

- **Changer régulièrement de mot de passe.**
- **Les mots de passe doivent être suffisamment complexes**, c'est à dire composés de lettres minuscules, de lettres majuscules, de chiffres et de caractères spéciaux. Un bon mot de passe doit être facilement mémorisable. Des stratégies telles que l'utilisation des premières lettres de chaque mot d'une phrase peuvent être utilisées.
- **Ne pas écrire ni communiquer son mot de passe** à qui que ce soit.

I Section développeurs/administrateurs

1. Gestion des droits et des utilisateurs

- **Mettre en place un système d'authentification**. Cela permet d'identifier et de tracer sur le réseau les utilisateurs individuellement ou au sein d'un groupe.

- **Conserver les logs des activités des utilisateurs sur le réseau.** Cela peut être crucial pour comprendre l'origine d'une panne ou d'une attaque. Cette conservation des logs peut être limitée dans le temps et doit respecter les lois en vigueur concernant la protection des données et de la vie privée.

- **Implémenter un ensemble de règles limitant l'accès aux ressources physiques et logicielles.** Cela peut être fait en attribuant des rôles aux utilisateurs (Role_User, Role_Moderator, Role_Admin etc…). La gestion des rôles et des droits est un privilège que seul l'administrateur système possède. Pour une implémentation correcte, il convient de n'accorder aux utilisateurs que les droits qui leurs sont nécessaires. Ni plus, ni moins.

- **Ne pas utiliser le rôle administrateur pour un usage ne nécessitant pas les privilèges administrateur.**

2. Stockage des mots de passe et autres données sensibles

- **Les mots de passe ne doivent en aucun cas être stockés en clair** sur quelque machine que ce soit ou sur un répertoire Gitlab ou Github. Par exemple, les mots de passe utilisés pour se connecter à une base de données devront :
    - Être stockés dans les “secrets” de la solution de versionning pour l’environnement de production.
    - Être stockés en local pour l’environnement de Dev.

- Tous les mots de passe en base de données doivent être hashés.

- **Pour chaque type de données collectées, il est recommandé d’attribuer un niveau de criticité et des règles propres à chaque niveau de criticité.** Si le niveau de criticité l’exige, les données seront cryptées avant tout transite ou stockage.

3. Accès aux machines distantes en SSH

- **La gestion des droits d'accès aux machines distantes revient à l'administrateur système**. Son rôle est de créer une session "développeur" et d'ajouter les clés SSH des différents développeurs travaillant sur le projet, ceci afin que les autres membres de l'équipe puissent s'y connecter.

- Une fois que l'administrateur a finalisé la configuration d'une machine distante, **il doit désactiver la possibilité de se connecter par mot de passe sur la machine distante.**

- **Les clés privées utilisateurs sont strictement confidentielles** et ne doivent être en aucun cas divulguées ou utilisées sur des sites peu recommandables.

- Si la clé privée d'un utilisateur vient à changer ou si un utilisateur ne fait plus partie de l'équipe, elle devra être rapidement remplacée ou supprimée sur la machine distante.

- **Il convient d'utiliser des logiciels tels que Fail2Ban afin de limiter le nombre de tentatives de connexion** à une machine distante et ainsi éviter les intrusions par "force brute".

- **Il est impératif de paramétrer un pare-feu** pour fermer les ports des machines distantes dès lors qu'ils ne sont pas nécessaires au bon fonctionnement de cette dernière.

4. Récupération des données

- Afin de garantir la sécurité et l'intégrité des données stockées en base de données, **il est fortement conseillé de mettre en place une sauvegarde automatisée** quotidienne (en local, sur un répertoire Git, sur un disque dur externe ou sur une machine distante).

- **Il est conseillé de cloner l'état de votre système** toutes les semaines ou tous les mois. Cela revêt une importance capitale dans le cas où une mise à jour ne serait pas stable ou si votre système est détruit ou endommagé à la suite d'une intrusion malveillante. Ce clone de la machine est en général géré de manière automatique par l'hébergeur.

5. Mises à jour et maintien du système

- **Le système ainsi que tous ses services doivent être en permanence tenus à jour** afin de limiter les failles de sécurité. Comme mentionné précédemment, il est conseillé de faire une sauvegarde du système avant d'effectuer les mises à jour.

6. Protocoles de communication

- **Le protocole HTTPS est obligatoire pour l'échange de données sécurisées sur internet**. Plus généralement, l'utilisation de protocoles cryptés est fortement recommandée même sur des réseaux locaux.

- **L'utilisation de JWT tokens est recommandée pour l'authentification** des utilisateurs sur la solution web.