



From Software

Ensemble des incidents

Alves Fernandes
INTERWEB

From Software

Sommaire

Table des matières

Sommaire	1
- I- Introduction	2
II- Les erreurs de codes.	2
1. Lien avec la base de donnée	2
2. Erreur de hachage.....	3
3. Erreur de connexion.....	4

From Software

I- Introduction

- Dans ce document, nous listerons :
 - o Les erreurs survenues dans le code.

II- Les erreurs de codes.

1. Lien avec la base de donnée

```
Exception in thread "AWT-EventQueue-0" java.lang.ExceptionInInitializerError
    at controleur.ClientLourd.creerVueGenerale(ClientLourd.java:28)
    at vue.VueConnexion.traitemement(VueConnexion.java:100)
    at vue.VueConnexion.actionPerformed(VueConnexion.java:75)
    at java.desktop/javax.swing.AbstractButton.fireActionPerformed(AbstractButton.java:1972)
    at java.desktop/javax.swing.AbstractButton$Handler.actionPerformed(AbstractButton.java:2313)
    at java.desktop/javax.swing.DefaultButtonModel.fireActionPerformed(DefaultButtonModel.java:405)
    at java.desktop/javax.swing.DefaultButtonModel.setPressed(DefaultButtonModel.java:262)
    at java.desktop/javax.swing.plaf.basic.BasicButtonListener.mouseReleased(BasicButtonListener.java:279)
    at java.desktop/java.awt.Component.processMouseEvent(Component.java:6626)
    at java.desktop/javax.swing.JComponent.processMouseEvent(JComponent.java:3389)
    at java.desktop/java.awt.Component.processEvent(Component.java:6391)
    at java.desktop/java.awt.Container.processEvent(Container.java:2266)
    at java.desktop/java.awt.Component.dispatchEventImpl(Component.java:5001)
    at java.desktop/java.awt.Container.dispatchEventImpl(Container.java:2324)
    at java.desktop/java.awt.Component.dispatchEvent(Component.java:4833)
    at java.desktop/java.awt.LightweightDispatcher.retargetMouseEvent(Container.java:4948)
    at java.desktop/java.awt.LightweightDispatcher.processMouseEvent(Container.java:4575)
    at java.desktop/java.awt.LightweightDispatcher.dispatchEvent(Container.java:4516)
    at java.desktop/java.awt.Container.dispatchEventImpl(Container.java:2310)
    at java.desktop/java.awt.Window.dispatchEventImpl(Window.java:2780)
    at java.desktop/java.awt.Component.dispatchEvent(Component.java:4833)
    at java.desktop/java.awt.EventQueue.dispatchEventImpl(EventQueue.java:775)
    at java.desktop/java.awt.EventQueue$4.run(EventQueue.java:720)
    at java.desktop/java.awt.EventQueue$4.run(EventQueue.java:714)
    at java.base/java.security.AccessController.doPrivileged(AccessController.java:399)
    at java.base/java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain$JavaSecurityAccessImpl.java:100)
    at java.desktop/java.awt.EventQueue$5.run(EventQueue.java:747)
    at java.desktop/java.awt.EventQueue$5.run(EventQueue.java:745)
    at java.base/java.security.AccessController.doPrivileged(AccessController.java:399)
    at java.base/java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain$JavaSecurityAccessImpl.java:100)
    at java.desktop/java.awt.EventQueue.dispatchEvent(EventQueue.java:744)
    at java.desktop/java.awt.EventDispatchThread.pumpOneEventForFilters(EventDispatchThread.java:203)
    at java.desktop/java.awt.EventDispatchThread.pumpEventsForFilter(EventDispatchThread.java:124)
    at java.desktop/java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:113)
```

- Ce problème est survenu après c'être connecter avec son nom et mot de passe cette erreur s'affichait.
- La raison est que Eclipse ne faisait pas le lien entre java et MySQL pour la base de données.
- Nous avons installé un packages supplémentaire pour ce faire, « mysql-connector-j-9.2.0 » dans le dossier de notre projet. Permettant de faire marcher notre application.

From Software

2. Erreur de hachage

Dans mon application, lors de la connexion d'un utilisateur, le mot de passe saisi était haché avec la méthode `Modele.hashSHA1`. Cependant, j'ai découvert que le mot de passe était déjà haché une première fois avant d'être inséré dans la base de données. Lors de la vérification, je hachais à nouveau ce mot de passe déjà haché, ce qui entraînait une incohérence. En conséquence, même avec les bons identifiants, la connexion échouait systématiquement.

Le problème venait de la méthode `traitement` dans la classe `VueConnexion`. Voici l'erreur :

```
String mdpHash = Modele.hashSHA1(mdp);
String doubleHash = Modele.hashSHA1(mdpHash); // Double hachage par erreur
Manager unManager = Controleur.selectWhereManager(email, doubleHash);
```

Le mot de passe était haché deux fois (une fois lors de l'insertion et une autre fois ici), alors que la base de données ne contenait qu'un hachage simple.

Pour résoudre ce problème, j'ai supprimé le double hachage et m'assuré que le mot de passe était haché **une seule fois** avant d'être comparé à celui de la base de données. Voici la version corrigée :

```
public void traitement() {
    String email = this.txtEmail.getText();
    String mdp = new String(this.txtMdp.getPassword());

    // Hachage du mot de passe (une seule fois)
    String mdpHash = Modele.hashSHA1(mdp);

    // Vérification avec le mot de passe stocké dans la base de données
    Manager unManager = Controleur.selectWhereManager(email, mdpHash);
    if (unManager == null) {
        JOptionPane.showMessageDialog(this, "Veuillez vérifier vos identifiants.",
            "Erreur de Connexion", JOptionPane.ERROR_MESSAGE);
    } else {
        JOptionPane.showMessageDialog(this, "Bienvenue " + unManager.getNom());
        ClientLourd.setAdminConnecte(unManager);
        ClientLourd.rendreVisibleVueConnexion(false);
        ClientLourd.creerVueGenerale(true);
    }
}
```

Résultat

Après cette correction, la connexion fonctionne correctement. Le mot de passe saisi est haché une seule fois et correspond au hachage stocké dans la base de données. Cela a résolu le problème d'authentification et a permis aux utilisateurs de se connecter sans problème.

From Software

3. Erreur de connexion

Un jour, en travaillant sur la page de connexion de mon application, j'ai rencontré un problème frustrant : **la connexion échouait systématiquement**, même lorsque je saisisais les bons identifiants. Après quelques tests, j'ai réalisé que l'erreur venait d'un **problème de hachage du mot de passe**.

L'incident

Dans la méthode ``traitement``, le mot de passe saisi par l'utilisateur était bien récupéré, mais il n'était pas correctement haché avant d'être comparé à celui stocké dans la base de données. Cela signifiait que la requête SQL ne trouvait jamais de correspondance, car la base de données contenait des mots de passe hachés, tandis que je comparais directement le mot de passe en clair.

La cause

En relisant mon code, j'ai remarqué que j'avais oublié d'appeler la méthode ``Modele.hashSHA1`` pour hacher le mot de passe avant de l'envoyer à la méthode ``Controleur.selectWhereManager``. Voici la partie problématique de mon code :

```
String email = this.txtEmail.getText();
String mdp = new String(this.txtMdp.getPassword());

// Erreur : le mot de passe n'est pas haché avant la comparaison
Manager unManager = Controleur.selectWhereManager(email, mdp);
```

From Software

Solution :

Pour résoudre ce problème, j'ai ajouté l'appel à la méthode `Modele.hashSHA1` pour m'assurer que le mot de passe était haché avant d'être comparé. Voici la version corrigée de la méthode traitement :

```
public void traitement() {
    String email = this.txtEmail.getText().trim();
    String mdp = new String(this.txtMdp.getPassword()).trim();

    // Hachage du mot de passe
    String mdpHash = Modele.hashSHA1(mdp);

    // Vérification de la connexion du manager par email et mot de passe haché
    Manager unManager = Controleur.selectWhereManager(email, mdpHash);
    if (unManager == null) {
        JOptionPane.showMessageDialog(this, "Veuillez vérifier vos identifiants.",
            "Erreur de Connexion", JOptionPane.ERROR_MESSAGE);
    } else {
        JOptionPane.showMessageDialog(this, "Bienvenue sur l'application FromSoftware
Recrutement Administrator\n"
            + " Identifiants : \n"
            + "Nom : " + unManager.getNom() + "\n"
            + "Prénom : " + unManager.getPrenom() + "\n",
            "Connexion en admin", JOptionPane.INFORMATION_MESSAGE);

        // Sauvegarde du manager connecté
        ClientLourd.setAdminConnecte(unManager);

        // Ouverture du logiciel
        ClientLourd.rendreVisibleVueConnexion(false);
        ClientLourd.creerVueGenerale(true);
    }
}
```

Résultat

Après cette correction, la connexion a fonctionné parfaitement. Le mot de passe saisi par l'utilisateur était correctement haché, et la comparaison avec la base de données se faisait sans problème. Cela m'a rappelé l'importance de toujours vérifier que les données sensibles, comme les mots de passe, sont traitées de manière cohérente entre le client et la base de données.

Depuis, je fais toujours attention à inclure des tests pour vérifier que les mots de passe sont correctement hachés et comparés.