

# Home Lab – Mini-Script linux



**Auteur :** Ismaël Baby

**Période :** Automne 2025

**Étudiant en Cybersécurité**

**Rédigé le 18 novembre 2024**

# Home Lab – Mini-Script linux

## Contents

1. Introduction.....	2
1.1 Contexte et Objectifs .....	3
1.2 Demande d'informations à l'utilisateur : .....	3
1.3 :Vérification des identifiants : .....	4
1.4 : Réponse en fonction de la vérification :.....	4
1.5 : Simulation d'une alerte de sécurité :.....	5
5. Gestion des alertes : .....	6
2.Conclusion : .....	7

# Home Lab – Mini-Script linux

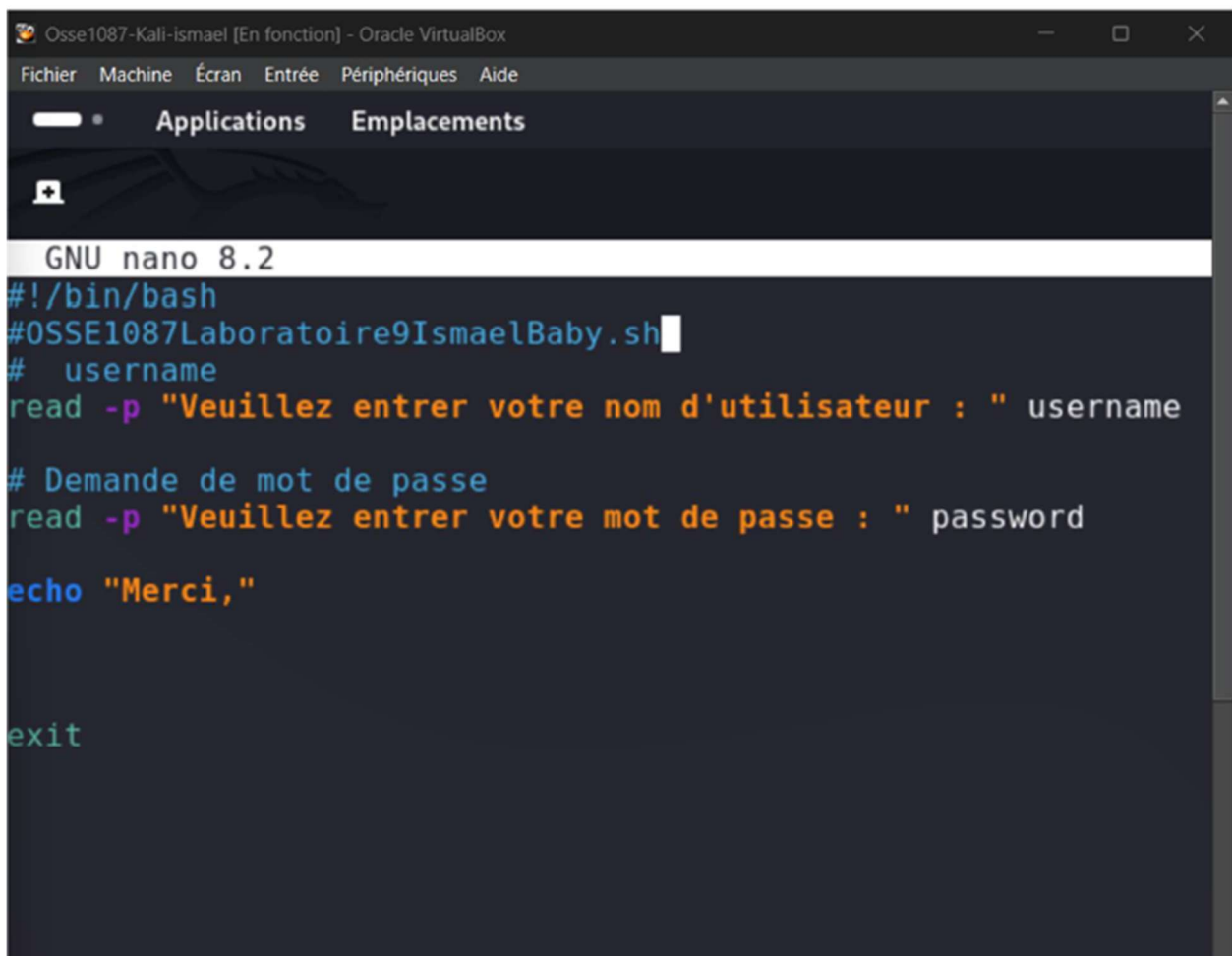
## 1. Introduction

### 1.1 Contexte et Objectifs

Ce rapport documente la réalisation d'une home lab, axé sur la cybersécurité. L'objectif principal était de concevoir et d'implémenter un script en Bash Shell capable de gérer un processus d'authentification utilisateur tout en intégrant des mécanismes de sécurité proactifs.

### 1.2 Demande d'informations à l'utilisateur :

Utilisez la commande `read` pour demander à l'utilisateur de saisir son nom d'utilisateur et son mot de passe.

A screenshot of a terminal window titled "Osse1087-Kali-ismael [En fonction] - Oracle VirtualBox". The window shows the GNU nano 8.2 editor editing a file named "OSSE1087Laboratoire9IsmaelBaby.sh". The script content is as follows:

```
#!/bin/bash
# OSSE1087Laboratoire9IsmaelBaby.sh
# username
read -p "Veuillez entrer votre nom d'utilisateur : " username

# Demande de mot de passe
read -p "Veuillez entrer votre mot de passe : " password

echo "Merci,"

exit
```

## Home Lab – Mini-Script linux

```
(cyse-ismael@kali) - [~]  
$ sudo ./OSSE1087Laboratoire9IsmaelBaby.sh  
Veuillez entrer votre nom d'utilisateur : ismael  
Veuillez entrer votre mot de passe : baby  
Merci,
```

### 1.3 :Vérification des identifiants :

Utilisez une structure conditionnelle if-else pour vérifier si les identifiants saisis correspondent aux identifiants autorisés. Utilisez des variables pour les identifiants corrects.

```
#!/bin/bash  
#OSSE1087Laboratoire9IsmaelBaby.sh  
bonusername="Ismael"  
mauvaispassword="Cyse1"  
  
read -p "Veuillez entrer votre nom d'utilisateur : " username  
  
# username  
read -p "Veuillez entrer votre mot de passe : " password  
echo  
  
# Demande de mot de passe  
if [ "$username" == "$bonusername" ] && [ "$password" == "$mauvaispassword" ]; then  
    echo "Accès autorisé."  
else  
    echo "Accès refusé."  
fi
```

```
(cyse-ismael@kali) - [~]  
$ sudo ./OSSE1087Laboratoire9IsmaelBaby.sh  
Veuillez entrer votre nom d'utilisateur : Ismael  
Veuillez entrer votre mot de passe :  
Accès autorisé.
```

### 1.4 : Réponse en fonction de la vérification :

- Si les identifiants sont corrects, affichez un message de succès.
- Sinon, affichez un message d'erreur. Utilisez echo pour afficher les messages.

## Home Lab – Mini-Script linux

```
(cyse-ismael@kali) - [~]  
$ sudo ./OSSE1087Laboratoire9IsmaelBaby.sh  
Veuillez entrer votre nom d'utilisateur : Ismael  
Veuillez entrer votre mot de passe :  
Accès autorisé.
```

### 1.5 : Simulation d'une alerte de sécurité :

- Si l'utilisateur échoue après 3 tentatives, simulez une alerte de sécurité.
- Utilisez une autre structure conditionnelle pour afficher un message d'alerte.

```
GNU nano 8.2 OSSE1087Laboratoire9IsmaelBaby.sh  
#!/bin/bash  
#OSSE1087Laboratoire9IsmaelBaby.sh  
#é  
  
bonusername="Ismael"  
mauvaispassword="Cysel"  
tentatives=0  
maxdetentatives=3  
  
# Demander le nom d'utilisateur  
read -p "Veuillez entrer votre nom d'utilisateur : " username  
  
# Demander le mot de passe  
while [ $tentatives -lt $maxdetentatives ]; do  
    read -p "Veuillez entrer votre mot de passe : " password  
    echo  
  
    if [ "$username" == "$bonusername" ] && [ "$password" == "$mauvaispassword" ]; then  
        echo "Accès autorisé."  
        break  
    else  
        tentatives=$((tentatives + 1))  
        echo "Accès refusé."  
    fi  
  
    # Vérification  
    if [ $tentatives -ge $maxdetentatives ]; then  
        echo "Alerte de sécurité : Acces refusé."  
        break  
    fi  
  
done
```

## Home Lab – Mini-Script linux

```
(cyse-ismael@kali) - [~]  
$ sudo ./OSSE1087Laboratoire9IsmaelBaby.sh  
Veuillez entrer votre nom d'utilisateur : isss  
Veuillez entrer votre mot de passe : sss  
  
Accès refusé.  
Veuillez entrer votre mot de passe : sss  
  
Accès refusé.  
Veuillez entrer votre mot de passe : ss  
  
Accès refusé.  
Alerte de sécurité : Acces refusé.  
  
(cyse-ismael@kali) - [~]  
$
```

### 5. Gestion des alertes :

- En fonction de la gravité de l'alerte (échec des tentatives), affichez des recommandations de sécurité à l'utilisateur. Utilisez un message d'alerte pour guider l'utilisateur.

```
(cyse-ismael@kali) - [~]  
$ sudo ./OSSE1087Laboratoire9IsmaelBaby.sh  
Veuillez entrer votre nom d'utilisateur : lk  
Veuillez entrer votre mot de passe : k  
  
Accès refusé.  
Alerte : Première tentative échouée. Vérifiez votre nom d'utilisateur et votre mot de passe.  
Veuillez entrer votre mot de passe : kk  
  
Accès refusé.  
Alerte : Deuxième tentative échouée. Vous avez encore une tentative avant de bloquer l'accès.  
Veuillez entrer votre mot de passe : jj  
  
Accès refusé.  
- Si vous avez oublié vos identifiants, contactez l'administrateur.
```

# Home Lab – Mini-Script linux

```
#!/bin/bash
# OSSE1087Laboratoire9IsmaelBaby.sh
#é

bonusername="Ismael"
mauvaispassword="Cysel"
tentatives=0
maxdetentatives=3

# username
read -p "Veuillez entrer votre nom d'utilisateur : " username

# Demander de mot de passe
while [ $tentatives -lt $maxdetentatives ]; do
    read -p "Veuillez entrer votre mot de passe : " password
    echo

    if [ "$username" == "$bonusername" ] && [ "$password" == "$mauvaispassword" ]; then
        echo "Accès autorisé."
        break
    else
        tentatives=$((tentatives + 1))
        echo "Accès refusé."
    fi

    if [ $tentatives -eq 1 ]; then
        echo "Alerte : Première tentative échouée. Vérifiez votre nom d'utilisateur et votre"
    elif [ $tentatives -eq 2 ]; then
        echo "Alerte : Deuxième tentative échouée. Vous avez encore une tentative avant de bl"
    fi

    # Alerte de sécurité après 3 tentatives échouées
    if [ $tentatives -ge $maxdetentatives ]; then
        echo "- Si vous avez oublié vos identifiants, contactez l'administrateur."
    fi
done
```

## 2.Conclusion :

Ce lab permet de créer un script bash pour gérer les tentatives de connexion d'un utilisateur. Le script vérifie le nom d'utilisateur et le mot de passe, limite les tentatives à trois, et affiche des alertes en cas d'échec. Après trois tentatives échouées, il donne des recommandations de sécurité pour guider l'utilisateur et renforcer la sécurité de ses identifiants.