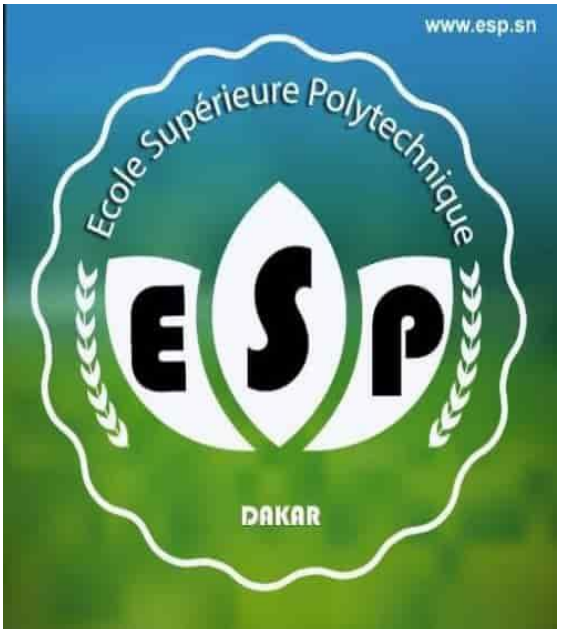
Université Cheikh Anta Diop de Dakar

Ecole Supérieure Polytechnique




Département Génie Informatique

Ingénierie Cryptographique

**Membres du groupes**

Haby DIOP

Mariama Ka

Marième AIDARA

Papa Cheikh GNINGUE

Taamba Bedel Brice THIOMBIANO

Année anniversaire : 2022/2023

# Plan

I. Matching des concepts avec une partie du cours

II. Traiter la catégorie par rapport à elle-même (CVE et CWE associées)

III. Création de notre propre scénario

IV. Outils pour mettre en place les scenarios et les bonnes pratiques

V. Implémentation d’un scénario

VI. Les mesures préventives

## I. Matching des concepts avec une partie du cours

Des Attaque par force brute : lorsque l'attaquant a la possibilité de tenter toutes les clés une par une pour le décryptage : (chapitre1 _ page 52)

D'utiliser des mots de passe en texte brut, chiffrés ou faiblement hachés : dus à la mauvaise utilisation de primitives cryptographiques tels que le chiffrement et des fonctions de hachages, dont il est question dans le chapitre1 du cours à la page 23.

D'autoriser l'authentification par défaut et non autorisée : due à l'utilisation des mots de passe par défaut, faibles ou bien connus, tels que "Password1" ou "admin / admin" ou « passer ». Cela va entrainer la violation des services de sécurités à savoir la confidentialité et l'authentification (chapitre1_page17).

## II. Traiter la catégorie par rapport à elle-même (CVE et CWE associées)

Cette vulnérabilité présente plusieurs échecs d'authentification dues à deux causes :

1. L'utilisateur n'est pas autorisé à lire le modèle de connexion par mot de passe à usage unique.
2. L'ordinateur de l'utilisateur ne peut pas accéder au contrôleur de domaine en raison de problèmes réseau.

Quelques CWEs et CVEs associées à cette vulnérabilité :

- **CWE-259 Utilisation d'un mot de passe codé en dur**

Le logiciel contient un mot de passe codé en dur, qu'il utilise pour sa propre authentification entrante ou pour la communication sortante vers des composants externes. Un mot de passe codé en dur entraîne généralement un échec d'authentification important qui peut être difficile à détecter pour l'administrateur système. Une fois détecté, il peut être difficile à corriger, de sorte que l'administrateur peut être contraint de désactiver complètement le produit. Il existe deux variantes principales :

Entrant : le logiciel contient un mécanisme d'authentification qui recherche un mot de passe codé en dur.

Sortant : le logiciel se connecte à un autre système ou composant, et il contient un mot de passe codé en dur pour se connecter à ce composant.

| | |
|---|---|
| **CVE-2022-29964** | Le système de contrôle distribué (DCS) a des mots de passe codés en dur pour l'accès au shell local |

| | |
|---|---|
| **CVE-2021-37555** | Le service Telnet pour le chargeur IoT pour chiens et chats a un mot de passe codé en dur |

- **CWE-287 Authentification incorrecte**

Lorsqu'un acteur prétend avoir une identité donnée, le logiciel ne prouve pas ou prouve insuffisamment que la revendication est correcte.

| | |
|---|---|
| **CVE-2022-36436** | Le proxy d'authentification basé sur Python n'applique pas l'authentification par mot de passe lors de la poignée de main initiale, ce qui permet au client de contourner l'authentification en spécifiant un type d'authentification "Aucun". |
| **CVE-2022-33139** | Le système SCADA utilise uniquement l'authentification côté client, permettant aux adversaires de se faire passer pour d'autres utilisateurs. |
| **CVE-2020-13927** | Le paramètre par défaut dans le produit de gestion des flux de travail autorise toutes les demandes d'API sans authentification, comme exploité dans la nature par CISA KEV. |
| **CVE-2009-2382** | Le script d'administration permet le contournement de l'authentification en définissant une valeur de cookie sur "LOGGEDIN". |
| **CVE-2009-3232** | le script de mise à jour de l'authentification ne gère pas correctement lorsque l'administrateur ne sélectionne aucun module d'authentification, ce qui permet le contournement de l'authentification. |

- **CWE-304 Étape critique manquante dans l'authentification**

Le logiciel met en œuvre une technique d'authentification, mais il saute une étape qui fragilise la technique. Les techniques d'authentification doivent suivre les algorithmes qui les définissent exactement, sinon l'authentification peut être contournée ou plus facilement soumise à des attaques par force brute.

| | |
|---|---|
| **CVE-2004-2163** | Secret partagé non vérifié dans un paquet de réponse RADIUS, permettant le contournement de l'authentification en usurpant les réponses du serveur. |

- **CWE-306 Authentification manquante pour la fonction critique**

Le produit n'effectue aucune authentification pour les fonctionnalités nécessitant une identité d'utilisateur vérifiable ou consommant une quantité importante de ressources. Au fur et à mesure que les données sont migrées vers le cloud, si l'accès ne nécessite pas d'authentification, il peut être plus facile pour les attaquants d'accéder aux données depuis n'importe où sur Internet.

| | |
|---|---|
| **CVE-2022-29951** | Le protocole basé sur TCP dans Programmable Logic Controller (PLC) n'a pas d'authentification. |
| **CVE-2022-29952** | Le micro-logiciel de Condition Monitor utilise un protocole qui ne nécessite pas d'authentification. |
| **CVE-2022-30276** | Le protocole basé sur SCADA pour le pontage du trafic WAN et LAN n'a pas d'authentification. |
| **CVE-2022-30313** | Safety Instrumented System utilise des protocoles TCP propriétaires sans authentification. |
| **CVE-2022-30317** | Le système de contrôle distribué (DCS) utilise un protocole sans authentification. |
| **CVE-2020-13927** | Le paramètre par défaut dans le produit de gestion des flux de travail autorise toutes les demandes d'API sans authentification, comme exploité dans la nature par CISA KEV. |
| **CVE-2002-1810** | MFV. Accédez au serveur TFTP sans authentification et obtenez un fichier de configuration contenant des informations sensibles en texte brut. |
| **CVE-2008-6827** | Le logiciel agent exécuté avec des privilèges n'authentifie pas les demandes entrantes sur un canal non protégé, ce qui permet une attaque "Shatter". |

- **CWE-307 Restriction incorrecte des tentatives d'authentification excessives**

Le produit ne met pas en œuvre suffisamment de mesures pour empêcher plusieurs tentatives d'authentification infructueuses dans un court laps de temps, ce qui le rend plus vulnérable aux attaques par force brute.

| | |
|---|---|
| **CVE-2019-0039** | l'API REST pour un système d'exploitation réseau a une limite élevée pour le nombre de connexions, ce qui permet de deviner le mot de passe par force brute |
| **CVE-1999-1152** | Le produit ne se déconnecte pas ou n'expire pas après plusieurs échecs de connexion. |
| **CVE-1999-1324** | Les comptes d'utilisateurs ne sont pas désactivés lorsqu'ils dépassent un seuil ; peut-être un problème résultant. |

- **CWE-620** Changement de mot de passe non vérifier

Lors de la définition d'un nouveau mot de passe pour un utilisateur, le produit n'exige pas la connaissance du mot de passe d'origine ni l'utilisation d'une autre forme d'authentification.

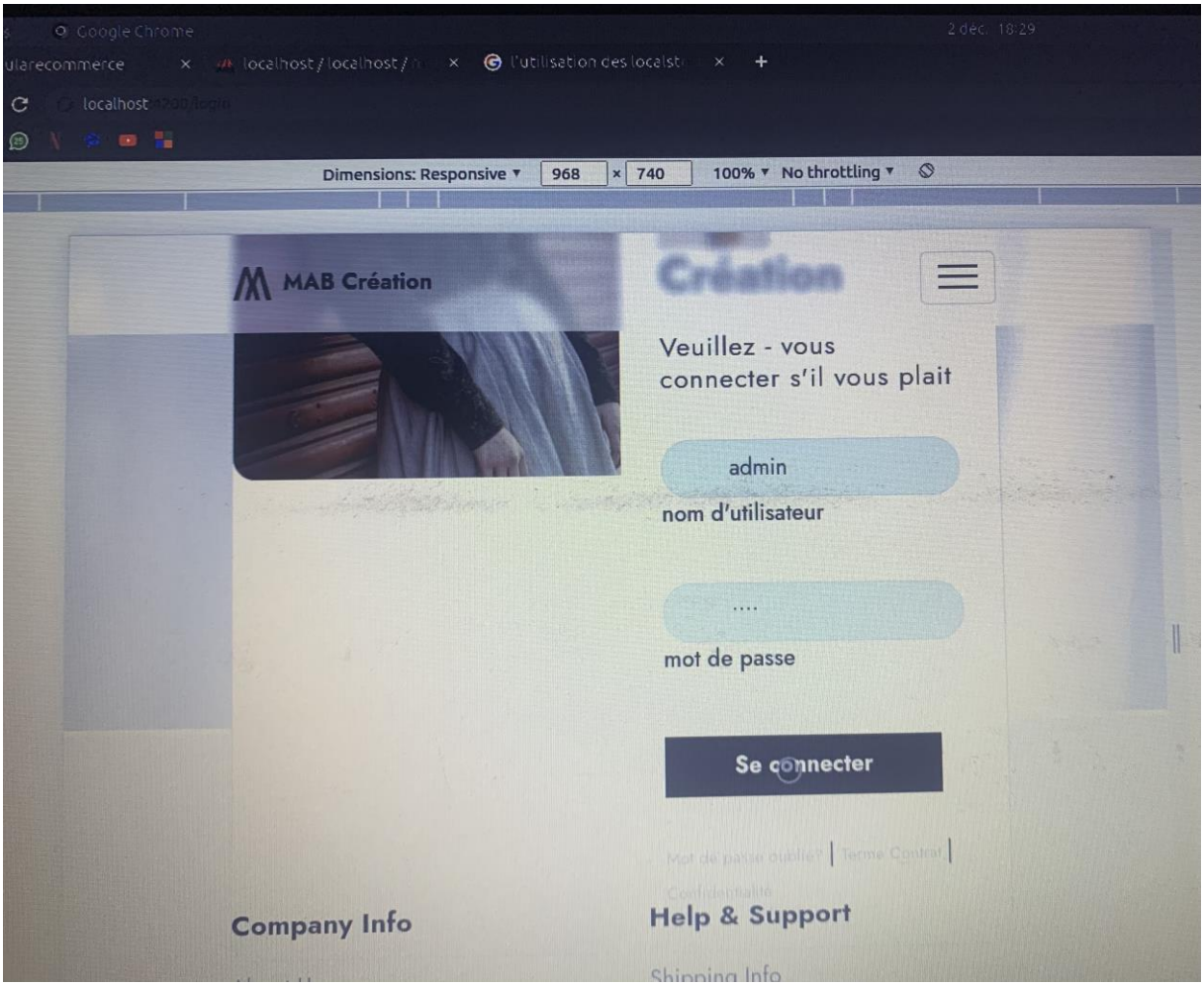
| | |
|---|---|
| **CVE-2007-0681** | L'application Web permet aux attaquants distants de modifier les mots de passe d'utilisateurs arbitraires sans fournir le mot de passe d'origine, et éventuellement d'effectuer d'autres actions non autorisées. |
| **CVE-2000-0944** | L'utilitaire de modification du mot de passe de l'application Web ne vérifie pas le mot de passe d'origine. |

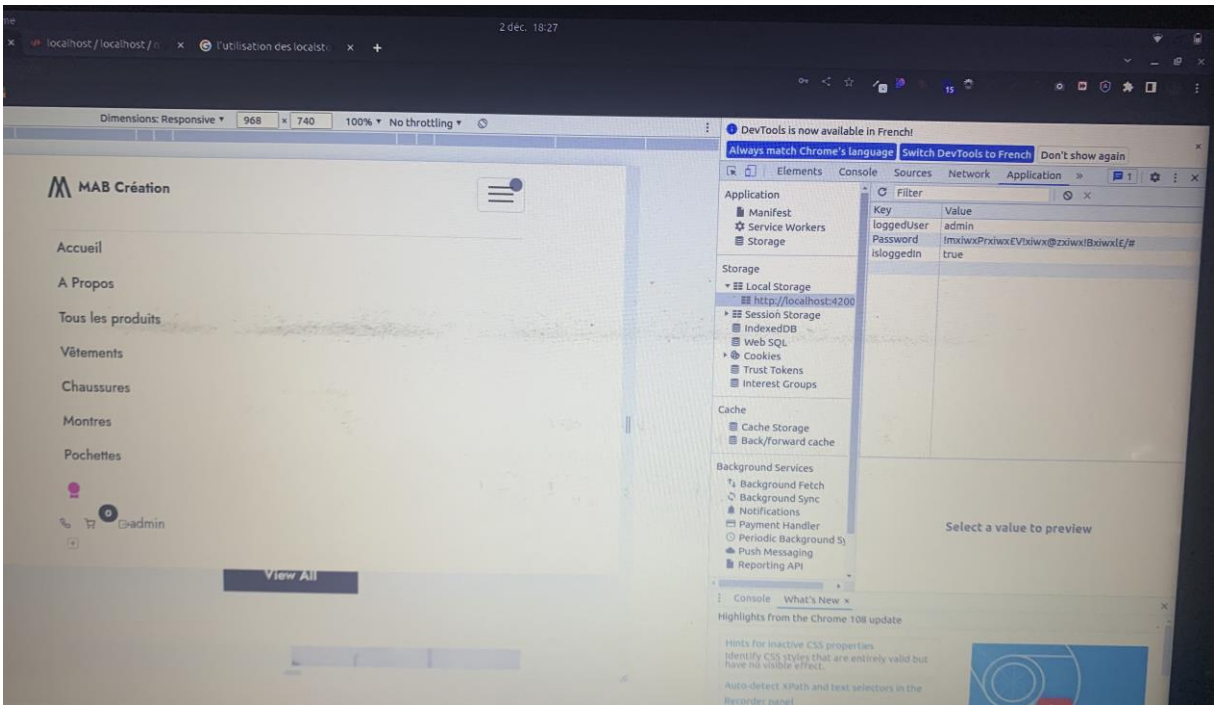
## III. Création de notre propre scénario

**Scénario :** l'utilisation des localStorages

Le localStorage est une méthode de stockage de données sur le navigateur créé pour pallier aux deux problèmes majeurs des cookies : le faible espace de stockage ainsi que le manque de réelles API pour y accéder en JavaScript. En effet lors de l'authentification, le localStorage enregistre le login et le mot de passe entré au niveau du navigateur. un attaquant porrait les réutilisait et avoir les même accès en retrouvant les informations dans : outils de développement –>Applications – >local Storage- http://Localhost:port d'écoute

**Illustration** :



## IV. Outils pour mettre en place les scenarios et les bonnes pratiques

## V. Implémentation d'un scénario

## VI. Les mesures préventives

- Lorsque cela est possible, implémentez l'authentification multifacteur pour éviter les attaques automatisées, le bourrage des informations d'identification, la force brute et la réutilisation des informations d'identification volées ;
- Ne pas livrer ou déployer avec des informations d'identification par défaut, en particulier pour les utilisateurs avec privilèges ;
- Intégrer des tests de mots de passes faibles, à la création ou au changement. Comparer ce mot de passe avec la liste des 10000 mots de passe les plus faibles ;
- Respecter la longueur, la complexité et la rotation des mots de passe par rapport aux directives du National Institute of Standards and Technology (NIST) 800-63 B à la section 5.1.1 ou autres directives modernes ;
- Assurez-vous que l'inscription, la récupération des informations d'identification et les chemins d'accès aux API sont durcis contre les attaques d'énumération de compte en utilisant le même message pour tous les résultats ;
- Limitez ou retardez de plus en plus les tentatives de connexion échouées, mais veillez à ne pas créer un scénario de déni de service. Enregistrer tous les échecs et alerter les administrateurs lors du bourrage des informations d'identification, de brute force ou d'autres attaques détectées
- Utilisez un gestionnaire de session intégré et sécurisé côté serveur qui génère un nouvel identifiant de session aléatoire avec une entropie élevée après la connexion. Les identifiants de session ne doivent pas se trouver dans l'URL, ils doivent être stockés de manière sécurisée et être invalidés après la déconnexion, une inactivité et une certaine durée.