Rapport d’audit technique

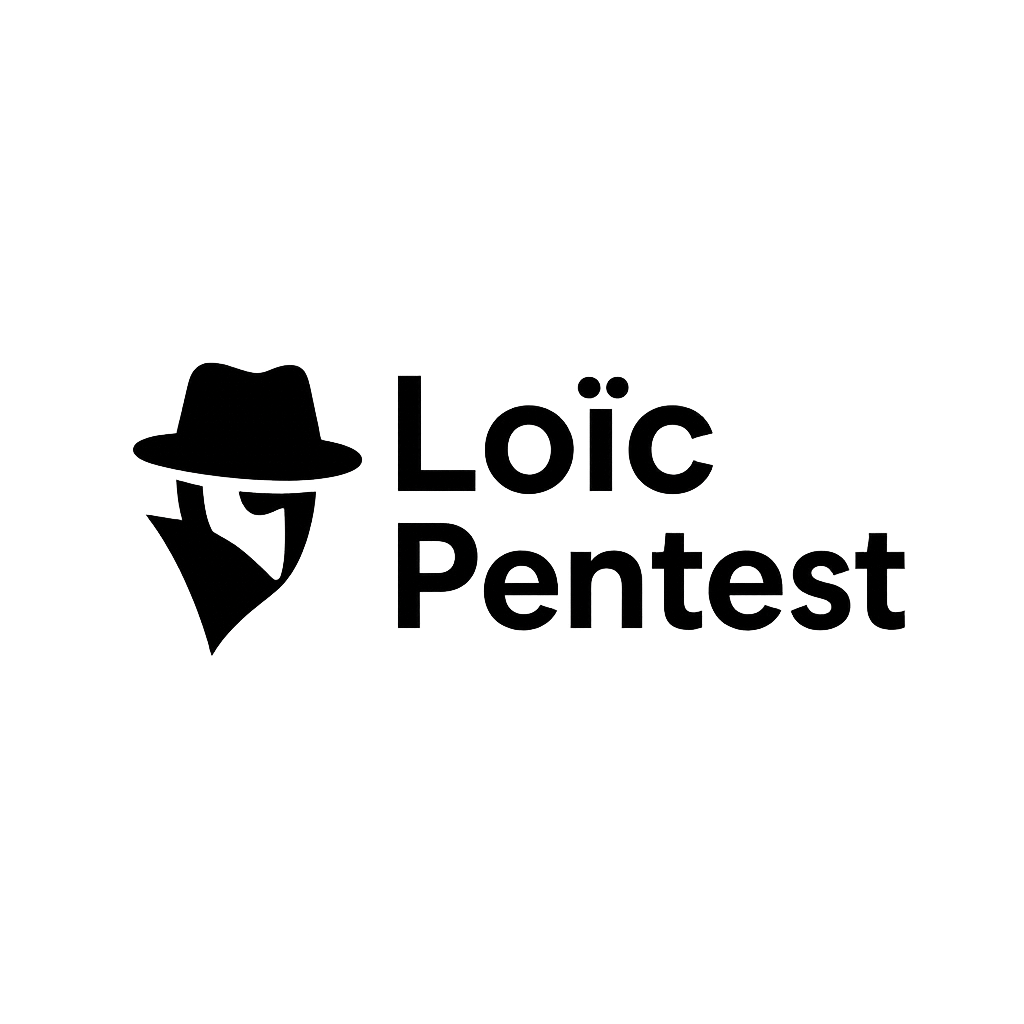
Test d’intrusion

**par Loïc BIGOT**

Début du test **: 04/09/2025**

Fin du test : **05/09/2025**

Version: **1.0**



Confidentialité

Ce document est strictement confidentiel et contient des informations sensibles relatives à la sécurité informatique de l'organisation. Toute diffusion, totale ou partielle, sans autorisation écrite préalable est interdite.

Historique des Versions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** |  | **Date** | **Auteur** | **Modifications** |
| **0.1** |  | 27/08/2025 | Loïc BIGOT | Création |
| **1.0** |  | 05/09/2025 | Loïc BIGOT | Livraison |

Liste de diffusion

|  |  |  |  |
| --- | --- | --- | --- |
| **Destinataire** |  | **Société** | **Objet** |
| **Mustapha M’hidi** |  | M2i Formation | Commanditaire de l’audit |

Table des matières

[I - Synthèse générale 4](#__RefHeading___Toc730_1215021673)

[1. Contexte et Objectifs 5](#__RefHeading___Toc732_1215021673)

[1.1 Résumé 5](#__RefHeading___Toc734_1215021673)

[1.2 Périmètre technique 5](#__RefHeading___Toc736_1215021673)

[1.3 Organisation de l’audit 5](#__RefHeading___Toc738_1215021673)

[1.4 Conclusion de l'expertise 5](#__RefHeading___Toc740_1215021673)

[1.5 Synthèses des vulnérabilités présentes 6](#__RefHeading___Toc742_1215021673)

[4. Scope & Périmètre 7](#__RefHeading___Toc744_1215021673)

[5. Méthodologie 7](#__RefHeading___Toc746_1215021673)

[6. Résultats des Tests 7](#__RefHeading___Toc748_1215021673)

[7. Recommandations 7](#__RefHeading___Toc750_1215021673)

[8. Conclusion 7](#__RefHeading___Toc752_1215021673)

[9. Annexes 7](#__RefHeading___Toc754_1215021673)

[II – Vulnérabilités présentes 8](#__RefHeading___Toc756_1215021673)

[1 Injection de commandes SQL 9](#__RefHeading___Toc758_1215021673)

[Tableau de classification des vulnérabilités Annexes A 10](#__RefHeading___Toc760_1215021673)

# I - Synthèse générale

## 1. Contexte et Objectifs

### 1.1 Résumé

La société <CHANGER> nous a mandaté pour effectuer un test d’intrusion externe sur son périmètre.  
L’objectif de cet audit était de vérifier la conformité de la solution étudiée par rapport à l’état de l’art en matière de sécurité et aux méthodes d’attaques connues.

Nos tests ont été réalisés selon une méthodologie dite « boîte noire », sans authentification préalable.

Ce rapport recense les vulnérabilités identifiées et propose une évaluation de leur risque.  
Cette évaluation est effectuée par l’auditeur en s’appuyant sur les standards techniques en vigueur et son expérience professionnelle. Elle prend en compte, autant que possible, la perspective d’un auditeur externe ainsi que l’environnement et le contexte métier de <CHANGER>.

Chaque vulnérabilité est accompagnée de recommandations techniques pour sa remédiation. Une synthèse globale du risque présente l’ensemble des découvertes.

Nous recommandons, après la mise en œuvre des correctifs, de procéder à un audit de validation afin de vérifier l’efficacité de ces mesures.

### 1.2 Périmètre technique

Le test d'intrusion a été réalisé sur l'environnement de production suivant :

<IP>

### 1.3 Organisation de l’audit

Les tests techniques ont été réalisés du 04/09/2025 au 05/09/2025 dans les locaux de <CHANGER>.

L'analyse de ces tests, la rédaction et la validation du livrable se sont poursuivies jusqu'à la livraison de celui-ci.

Les personnes ayant intervenu dans le processus d'audit sont :

|  |  |  |
| --- | --- | --- |
| **Nom et prénom** | **Email** | **Qualification** |
| Moustapha M’hidi | 27/08/2025 | Commanditaire de l’audit |
| Loïc BIGOT | 05/09/2025 | Auditeur |

### 1.4 Conclusion de l'expertise

#### 1.4a Conclusion de l'expertise

Le risque global associé au système audité est jugé **critique**. En effet, la probabilité d’occurrence d’une attaque est estimée **élevée** en raison des éléments suivants :

|  |  |  |
| --- | --- | --- |
| **Facteur** | **Classification** | **Justification** |
| Interaction utilisateur | Nulle | L’exploitation ne nécessite aucune action de l’utilisateur. |
| Privilèges requis | Faible |  |
| Complexité | Faible | Aucune condition particulière ni compétence avancée n’est requise. |
| Possibilité d’attaque | Réseau | L’attaque peut être menée via internet |

|  |  |  |
| --- | --- | --- |
| **Facteur** | **Classification** | **Justification** |
| Portée | Inchangée | La portée de l’impact demeure identique. |
| Confidentialité | Élevée | L’attaquant peut extraire la base de données et accéder aux mots de passe en clair. |
| Intégrité | Élevée | L’exploitation peut permettre la modification non autorisée de données. |
| Disponibilité | Élevée | L’exploitation peut entraîner une altération ou une interruption du service. |

### 1.5 Synthèses des vulnérabilités présentes

|  |  |  |  |
| --- | --- | --- | --- |
| **Numéro** | **Vulnérabilité** | **Environnement attaqué** | **Risque** |
| N1 | Injection SQL | <IP> | 8.6 |
| N2 |  |  | 5.3 |
| N3 |  |  | 3.1 |
| N4 |  |  |  |

# II – Vulnérabilités présentes

## 1. Injection de commandes SQL

Risque : Élevé

Probabilité : Élevé

Impact : Élevé

Nul besoin d’authentification, nul besoin de privilèges.

Calcul du score CVSS : [3-1#CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:L/A:L](https://www.first.org/cvss/calculator/3-1" \l "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:L/A:L)

### 1.1 Résumé

Le site est vulnérable à une injection de commandes SQL, qui permet de récupérer le contenu de la base de données et ainsi d'accéder à des données confidentielles.

### 1.2 Actifs concernés

172.20.10.10:80/tcp

### 1.3 Références

[[FR] PHP - Requêtes préparées et procédures stockées](https://www.owasp.org/index.php/Blind_SQL_Injection)

[[FR] Wikibooks - Table Ascii](https://fr.wikibooks.org/wiki/Les_ASCII_de_0_à_127/La_table_ASCII)

[[EN] OWASP - Injection SQ](https://www.owasp.org/index.php/SQL_Injection)L

[[EN] OWASP - Injection SQL en aveugle](https://www.owasp.org/index.php/Blind_SQL_Injection)

### 1.4 Description

Les vulnérabilités de type injection SQL concernent les applications interagissant avec une base de données.

Lors de la construction dynamique d’une requête à partir d'éléments fournis par l'utilisateur du service (authentification, recherche dans une page Web, consultation ou modification de contenu, etc.), les variables contiennent des valeurs fournies par l'utilisateur ou son client (navigateur Web). Si ces valeurs ne sont pas vérifiées par le service, leur interaction avec la syntaxe du langage de l'application peut provoquer des comportements inattendus et des vulnérabilités.

L'injection SQL, en particulier, se produit lorsqu'une requête à destination de la base de données est construite par l'application, à partir de paramètres fournis par l'utilisateur. Un utilisateur malveillant peut tenter de manipuler ses paramètres, afin de remplacer la requête SQL légitime par une requête sous son contrôle et à son profit.

Par exemple, il peut être en mesure de :

récupérer le contenu de la base de données, dont des informations sensibles (mots de passe, coordonnées, informations bancaires, etc.) ;

lire et écrire des fichiers sur la machine du serveur SQL (sous certaines conditions) ;

exécuter des commandes à distance sur la machine du serveur SQL (sous certaines conditions).

### 1.5 Recommandations

Employer des requêtes préparées et prendre soin de neutraliser toute donnée fournie enentrée sur les applications.

Les injections SQL doivent tout d’abord être bloquées à la source par l’emploi de requêtes préparées. Cette pratique de développement délègue au cadriciel applicatif le typage et la neutralisation des variables insérées dynamiquement dans les requêtes SQL. En PHP, vous pouvez par exemple utiliser la bibliothèque PDO. De plus, au niveau de l’application, aucune entrée utilisateur ne doit être considérée comme de confiance. Il faut donc filtrer, de préférence sur la base d'une liste blanche, les valeurs ou types de valeurs acceptables.

### 1.6 Exploitation

SCREEN

# III – Annexes

## Tableau de classification des vulnérabilités Annexes A

|  |  |  |  |
| --- | --- | --- | --- |
| **Niveau** | **Plage CVSS v3.x** | **Description** | **Exemple de réponse** |
| **Critique** | 9.0 – 10.0 | Vulnérabilité permettant une compromission totale à distance, exécution de code arbitraire sans interaction. | Déploiement urgent d’un correctif / désactivation du service |
| **Élevé** | 7.0 – 8.9 | Impact élevé : fuite de données sensibles, élévation de privilèges locale ou pivot réseau. | Application d’un patch dans les 24 h / renforcement des ACL |
| **Moyen** | 4.0 – 6.9 | Vulnérabilité non triviale : DoS partiel, information disclosure, contournement mineur de contrôle d’accès. | Planification du correctif sous 1 semaine / surveillance accrue |
| **Faible** | 0.1 – 3.9 | Impact limité : information non sensible, vulnérabilité nécessitant une forte interaction ou contexte particulier. | Correction lors de la prochaine release / revue de configuration |
| **N/A** | — | Vulnérabilité non applicable ou hors périmètre (ex. tests manquants). | Retrait de l’analyse ou clarification de périmètre |

## Résultat des ports réseaux

Fin du Document