M362 : Pentest : Part 1 - Part 2

mardi 13 février 2024 10:01

Bloc 36 - M362
VII**. Pentest (Test d'Intrusion)**
A. Introduction au Pentesting
B. Types de Tests d'Intrusion (Boîte noire, Boîte blanche, Boîte grise)
C. Outils et Techniques de Pentesting
D. Analyse et Rapport des Résultats

Objectifs du Module
- Comprendre les principes et les méthodologies des tests d'intrusion.
- Apprendre à planifier, exécuter et analyser un pentest.

A. **Introduction au Pentest

Définition des tests d'intrusion :

Un test d'intrusion, communément appelé **pentest (Penetration Testing)**, est une méthode proactive et autorisée pour évaluer la sécurité d'un système informatique ou réseau en simulant une attaque d'un acteur malveillant. L'objectif est d'identifier et d'exploiter les vulnérabilités existantes dans les systèmes, applications, infrastructures réseau, et autres points d'exposition potentiels.

Les pentests peuvent être réalisés sur une variété de cibles, incluant des applications web, des réseaux internes, des services cloud, des appareils mobiles, et même des systèmes physiques de sécurité. Ils sont exécutés dans le but de découvrir des failles de sécurité avant que les attaquants réels ne le fassent, permettant ainsi aux organisations de renforcer leurs défenses.

Objectifs des Tests d'Intrusion

Les tests d'intrusion servent plusieurs objectifs clés dans une stratégie de sécurité globale :

- Identification des Vulnérabilités : Détecter les faiblesses dans les systèmes, applications, et configurations réseau qui pourraient être exploitées par des attaquants.
- Évaluation de l'Impact des Vulnérabilités : Comprendre l'impact potentiel d'une exploitation réussie sur les ressources et les opérations de l'organisation.
- Validation des Contrôles de Sécurité : Tester l'efficacité des mécanismes de défense en place, y compris les dispositifs de prévention d'intrusion, les firewalls, et les politiques de sécurité.
- Conformité aux Normes et Réglementations : Aider les organisations à se conformer aux exigences de sécurité spécifiées par les réglementations et les normes de l'industrie, telles que le PCI DSS, le GDPR, et l'ISO 27001.
- Formation et Sensibilisation à la Sécurité : Fournir une expérience pratique aux équipes de sécurité et sensibiliser le personnel à l'importance de la cybersécurité et aux techniques d'attaque courantes.
- Planification des corrections et des améliorations : Offrir des recommandations basées sur les résultats du pentest pour appliquer les corrections et améliorer la posture de sécurité globale de l'organisation.

B. **Types de tests d'Intrusion : (Boîte noire, Boîte blanche, Boîte grise)

La définition de ces types de tests d'intrusion a déjà été vu dans les cours précédents. On a plutôt classé cela en fonction de la relation avec la société pour laquelle on va décider de faire un test d'intrusion.
Il faut bien faire la différence avec un Audit, lors d'un test d'intrusion (Pentest), le pentester adopte la position de l'attaquant. La méthode devrait être de type Boîte noire. Mais une société qui mandate un Pentest voudra vraisemblablement aussi faire monter en compétence son équipe informatique ou du moins échanger avec le(s) pentesters afin d'adopter les bonnes pratiques (Boîte Grise).

- Sources:
  - [Test d'intrusion - Wikipédia](https://fr.wikipedia.org/wiki/Test_d%27intrusion)

1. **Planification d'un Pentest**

- Définition du périmètre et des objectifs.

La définition du Périmètre :
Le périmètre d'un pentest délimite clairement les systèmes, réseaux, applications, et autres actifs numériques qui seront inclus dans le test. La définition précise du périmètre est cruciale pour plusieurs raisons :

- Focalisation des Efforts : Permet aux pentesters de concentrer leurs efforts sur les zones spécifiques d'intérêt ou de préoccupation, optimisant ainsi l'utilisation des ressources.
- Évitement des Surprises : Assure que toutes les parties prenantes ont une compréhension claire de ce qui sera testé, évitant ainsi les malentendus et les conflits potentiels.
- Conformité Légale et Contractuelle : Garantit que le pentest est réalisé dans le cadre légal approprié, respectant les accords contractuels et les lois sur la protection des données.

La définition du périmètre peut inclure :

- Adresses IP/Plages d'IP : Spécification des adresses IP ou des plages d'IP des systèmes à tester.
- Applications Web : Identification des applications web spécifiques, y compris les URL et les environnements (production, pré-production).
- Réseaux Internes/Externes : Décision sur les réseaux internes ou externes à inclure dans le test.
- Composants Physiques : Inclusion de dispositifs physiques, tels que les serveurs, les commutateurs, et les routeurs.

Définition des Objectifs :
Les objectifs du pentest définissent ce que l'organisation espère accomplir à travers le test. Ils doivent être alignés avec les objectifs de sécurité globaux de l'organisation et peuvent varier considérablement d'un test à l'autre. Les objectifs typiques incluent :

- Identification des Vulnérabilités : Découvrir les failles de sécurité dans les systèmes, applications, et réseaux.
- Évaluation de l'Impact des Exploitations : Comprendre les conséquences potentielles d'une exploitation réussie des vulnérabilités identifiées.
- Test des Mécanismes de Détection et de Réponse : Évaluer l'efficacité des systèmes de détection d'intrusion et des procédures de réponse aux incidents.
- Validation des Contrôles de Sécurité : Vérifier l'efficacité des contrôles de sécurité en place, tels que les pare-feu, les systèmes de prévention d'intrusion, et les politiques d'accès.
- Conformité aux Normes de Sécurité : Assurer que les systèmes et les processus sont conformes aux normes de sécurité et aux réglementations industrielles.

2. ** Préparation et organisation du test.

- Préparation et Organisation du Test

1. **Établissement des Règles d'Engagement**

Les **règles d'engagement (RoE)** définissent le cadre dans lequel le pentest doit être réalisé. Elles incluent des informations telles que :

- Les heures et les jours où les tests peuvent être effectués : Pour minimiser l'impact sur les opérations commerciales.
- Les techniques et méthodes autorisées : Définir ce qui est permis et ce qui ne l'est pas, comme l'exploitation des vulnérabilités ou le social engineering.
- Les points de contact : Identifier les personnes à contacter en cas de problèmes ou d'incidents pendant le test.
- Les procédures d'escalade : Établir comment gérer la découverte de vulnérabilités critiques ou d'autres problèmes importants.

2. **Collecte d'Informations Préliminaires**

Avant de commencer le test, collecter autant d'informations que possible sur la cible en utilisant des méthodes de **reconnaissance ouverte**. Cela peut inclure :

- Cartographie du réseau : Identifier la structure et les composants clés du réseau.
- Recherche de domaines et sous-domaines : Utiliser des outils comme "DNS enumeration" pour découvrir les domaines associés à l'organisation.
- Collecte d'informations sur les employés : Pour les tests impliquant du social engineering, recueillir des informations sur les employés via les réseaux sociaux ou d'autres sources publiques.

3. **Configuration de l'Environnement de Test**

Préparer l'environnement de test implique :

- Configuration des Outils et des Plateformes : S'assurer que tous les outils nécessaires, comme "**Kali Linux**", sont configurés et prêts à l'emploi.
- Établissement d'un Environnement Sécurisé : Configurer un environnement isolé pour le pentest afin de prévenir tout impact non intentionnel sur les réseaux ou systèmes réels.
- Accès aux Systèmes : Si nécessaire, obtenir des accès légitimes à certains systèmes ou applications pour les tests de boîte blanche.

4. **Communication et Coordination**

La communication est essentielle tout au long du processus de pentest :

- Briefing Initial : Réunir toutes les parties prenantes pour un briefing sur les objectifs, le périmètre, et les règles d'engagement du pentest.
- Mise en Place de Canaux de Communication : Établir des lignes de communication claires pour les mises à jour régulières, les alertes, et les rapports d'incident.

5. **Planification Détaillée**

Développer un plan d'action détaillé qui inclut :

- Chronologie : Un calendrier des activités de pentest, y compris les phases de reconnaissance, d'exploitation, et de post-exploitation.
- Répartition des Tâches : Assigner des rôles et des responsabilités spécifiques aux membres de l'équipe de pentest.

- Sources:
- [Test d'intrusion - Pentest | Cybersécurité - Devensys](https://www.devensys.com/cybersecurite/pentest-test-intrusion)

3. ** Phases d'un pentest : reconnaissance, scanning, exploitation, post-exploitation.

1. **Reconnaissance :** Collecte d'informations sur la cible sans interaction directe.

   Objectif : comprendre l'environnement de la cible, identifier les points d'entrée potentiels. Identifier les points faibles potentiels et les informations clés telles que les noms de domaine, les adresses IP, les informations sur les employés, etc.

   Activités :
   - Collecte d'informations passives : Recherche d'informations disponibles publiquement sans interaction directe avec la cible. Cela inclut l'analyse des réseaux sociaux, des bases de données **WHOIS**, et des archives web.
   - Collecte d'informations actives : Interaction directe avec la cible pour obtenir des données, comme l'envoi de **pings** ou de **requêtes DNS**.

2. **Scanning :** Utilisation d'outils automatisés pour identifier les systèmes, services, et vulnérabilités ouverts.

   Objectif : cartographier l'infrastructure de la cible et trouver des vulnérabilités exploitables.

   Activités :
   - Scanning de ports : Utilisation d'outils comme **Nmap** pour identifier les ports ouverts et les services associés.
   - Scanning de vulnérabilités : Emploi de scanners de vulnérabilités pour détecter les failles de sécurité connues dans les systèmes et applications.

3. **Exploitation :** Tentative d'exploiter les vulnérabilités identifiées pour accéder au système ou au réseau.

   Objectif : évaluer l'impact potentiel d'une attaque réussie.

   Activités :
   - Exploitation des vulnérabilités : Mise en œuvre d'attaques pour exploiter les failles trouvées, pouvant inclure l'exécution de code à distance, l'élévation de privilèges, ou l'injection SQL.

- Implantation de backdoors : Installation de points d'accès cachés (backdoors) pour garantir un accès futur au système.

4. **Post-Exploitation :** Exploration des systèmes compromis pour collecter des données sensibles, établir une présence persistante, ou naviguer latéralement à travers le réseau.

   Objectif : déterminer la valeur des systèmes compromis et identifier d'autres cibles potentielles.

   Activités :
   - Collecte de données sensibles : Accès à des bases de données, fichiers de configuration, emails, et documents internes.
   - Mouvement latéral : Exploration d'autres systèmes dans le réseau pour évaluer leur vulnérabilité et potentiellement les compromettre.

5. **Rapport :** Documentation des découvertes, des méthodes d'exploitation, et des recommandations pour les corrections.

   Objectif : fournir une analyse détaillée des vulnérabilités, des impacts, et des mesures correctives.

   Activités :
   - Rédaction du rapport : Présentation des vulnérabilités découvertes, des preuves d'exploitation, et des recommandations pour les corrections.
   - Présentation aux parties prenantes : Discussion des résultats avec les responsables de la sécurité de l'organisation cible pour planifier les mesures correctives.

4. **Méthodologies de Pentest**

- Approches méthodologiques (OSSTMM, PTES, etc.).

Open Source Security Testing Methodology Manual (OSSTMM) :

Description : L'OSSTMM est une méthodologie complète pour effectuer des tests de sécurité et d'audit. Elle couvre une large gamme d'aspects de la sécurité, y compris la sécurité physique, la sécurité des communications, et la sécurité des systèmes d'information.

Application : Utilisée pour obtenir une vue d'ensemble détaillée de la sécurité d'une organisation, l'OSSTMM est particulièrement utile pour les audits de sécurité approfondis.

Ressources et documents utiles : Le manuel officiel de l'OSSTMM : Disponible sur le site de l'Isecom (Institute for Security and Open Methodologies) à **ISECOM - OSSTMM**. Ce document est la source principale pour comprendre la méthodologie, ses applications, et comment mener un test de sécurité selon l'OSSTMM.

Penetration Testing Execution Standard (PTES) :

Description : Le PTES fournit un cadre standardisé pour mener des tests d'intrusion. Il détaille les phases d'un pentest, de la pré-engagement à la post-exploitation, en mettant l'accent sur la méthodologie et la documentation.

Application : Idéal pour structurer des pentests de manière cohérente, le PTES aide à assurer que tous les aspects critiques de la sécurité sont évalués.

Ressources et documents utiles : Le site officiel du PTES : PTES - Penetration Testing Execution Standard. Il offre un guide détaillé sur chaque phase du pentest selon le PTES, des recommandations pour les pentesters, et des conseils pour la rédaction de rapports de pentest.

Information Systems Security Assessment Framework (ISSAF) :

Description : L'ISSAF est une méthodologie orientée vers l'évaluation de la sécurité des systèmes d'information. Elle couvre la préparation, l'évaluation, et la phase de rapport, en se concentrant sur les techniques d'évaluation détaillées.

Application : Utilisée pour des évaluations de sécurité approfondies, l'ISSAF est adaptée aux environnements complexes nécessitant une analyse détaillée.

Ressources et documents utiles :

- Faire des recherches en ligne.

C. **Outils et Techniques de Pentest**

- Outils de pentesting (Kali Linux, Metasploit, etc.).

Kali Linux

- Présentation : Kali Linux est une distribution Linux basée sur Debian, conçue spécifiquement pour le pentesting et la sécurité informatique. Elle est développée et maintenue par Offensive Security, une entreprise leader dans le domaine de la formation en sécurité informatique. Kali Linux intègre plus de 100 outils pré-installés couvrant un large éventail de techniques de pentesting, y compris le scanning de réseaux, l'analyse de vulnérabilités, l'exploitation, et la post-exploitation.
- Points Clés :
  - Complète : Une suite complète d'outils pour toutes les phases du pentest.
  - Personnalisable : Possibilité de personnaliser les outils et l'environnement de travail selon les besoins spécifiques du pentest.
  - Communauté active : Large communauté d'utilisateurs et de développeurs offrant un soutien et des **mises à jour régulières**.

- TP 1 : Prise d'empreinte
  - Voir Fiche Pratique : Installation et configuration de Kali Linux
  - Voir Fiche pratique : Prise d'empreinte/Whois
  - En premier, utiliser les outils Wireshark pour l'analyse de paquets, Nmap pour le scanning de ports,
  - En second, utiliser les outils listés dans la partie Focus ci-dessous (SQLMap pour la base de données MySql)

- TP 2 : Prise d'empreinte/Scanning
  - Voir Fiche pratique : Prise d'empreinte/Scanning
  - Voir Fiche pratique : Scanners de vulnérabilité
  - **Focus sur architecture web Apache/MySQL :**
    i. Reconnaissance
      1) theHarvester : Utilisé pour recueillir des emails, des noms, des sous-domaines, des adresses IP, et des noms de domaine associés à l'organisation cible.
      2) Nmap : Bien qu'utilisé principalement pour le scanning, Nmap peut également être employé pour détecter les serveurs web actifs et les technologies utilisées.

    ii. Scanning
      1) Nmap : Pour scanner les ports ouverts et identifier les services exécutés sur la cible, y compris les versions d'Apache et MySQL.(Utiliser des scripts: http-enum ; mysql-vuln-cve2017-3599)
      2) Nikto : Un scanner web qui teste les serveurs web pour des milliers de vulnérabilités potentielles et des fichiers dangereux.

    iii. Exploitation
      1) Metasploit Framework : Pour exploiter les vulnérabilités découvertes dans Apache, MySQL, ou d'autres composants du système.
      2) SQLMap : Automatise la détection et l'exploitation des vulnérabilités d'injection SQL dans les applications web utilisant MySQL.

    iv. Post-Exploitation
      1) Meterpreter (via Metasploit) : Pour une interaction avancée avec le système compromis, collecte d'informations supplémentaires, et exploration latérale. (Utiliser Meterpreter pour explorer le système, élever les privilèges, ou installer des backdoors sur le serveur Apache.)
      2) sqlmap : Pour extraire des données de la base de données MySQL après l'exploitation.(extraire des données sensibles, telles que les informations d'identification des utilisateurs)

    v. Rapport
      1) Faraday : Plateforme de gestion de pentest pour organiser les données recueillies, les vulnérabilités identifiées, et faciliter la rédaction de rapports.
      2) Documenter les vulnérabilités spécifiques à Apache et MySQL trouvées, les méthodes d'exploitation utilisées, et les données potentiellement compromises.

3) Fournir des recommandations spécifiques pour sécuriser les configurations d'Apache et MySQL, ainsi que pour remédier aux vulnérabilités exploitées.

Metasploit (intégré à Kali)

- Présentation : Metasploit est un cadre (framework) de test d'intrusion open-source qui **permet le développement et l'exécution d'exploits** contre une machine distante. Il sert à tester la sécurité des systèmes en simulant des attaques sur des vulnérabilités connues. Metasploit est largement utilisé pour l'évaluation de la sécurité des systèmes informatiques et est intégré dans Kali Linux.
- Points Clés :
  - Polyvalent : Supporte les tests contre des systèmes Windows, Linux, et d'autres plateformes.
  - Base de données de vulnérabilités : Comprend une vaste base de données d'exploits connus et de payloads.
  - Interface utilisateur flexible : Offre une interface en ligne de commande ainsi qu'une interface graphique (via Armitage).

- TP 3 : Exploitation/Post Exploitation
  - Voir Fiche pratique : Exploitation
  - Recherche d'exploits : Utiliser Metasploit pour rechercher des vulnérabilités spécifiques dans la base de données d'exploits.
  - Configuration et lancement d'un exploit : Guide étape par étape pour configurer un exploit et l'utiliser pour tester la sécurité d'un système cible.
  - Interprétation des résultats : Analyse des résultats de l'exploit pour évaluer l'impact potentiel de la vulnérabilité.
  - **Focus sur architecture web Apache/MySQL :**
    i. Reconnaissance (Facultative avec Metasploit)
       1) Metasploit n'est pas le plus approprié pour cette phase. Plutôt se servir des outils du TP 1 et TP 2.
    ii. Scanning et Analyse de Vulnérabilités
       1) Modules auxiliaires conçus pour scanner des vulnérabilités spécifiques :
          a) **http_version :** Utiliser ce module pour identifier la version du serveur web Apache et rechercher des vulnérabilités connues associées à cette version.
          b) **mysql_version :** Détecter la version du serveur MySQL pour identifier les vulnérabilités potentielles.
    iii. Exploitation
       1) Fournit des exploits prêts à l'emploi pour une grande variété de vulnérabilités.
          a) **Exploits Apache :** Rechercher dans la base de données de Metasploit les exploits ciblant des vulnérabilités spécifiques dans le serveur web Apache. Cela peut inclure des dépassements de tampon, des injections de commandes, ou des vulnérabilités de script cross-site.
          b) **Exploits MySQL :** Utiliser des exploits pour MySQL pour obtenir un accès non autorisé à la base de données. Cela peut inclure l'exploitation de vulnérabilités d'injection SQL ou d'autres failles de sécurité dans MySQL.
    iv. Post-Exploitation
       1) Offre une variété de payloads et de modules de post-exploitation pour extraire des données, élever les privilèges, ou maintenir l'accès.
          a) **Meterpreter :** Un payload avancé qui fournit un contrôle interactif sur le système cible. Il peut être utilisé pour explorer le système, accéder à la base de données MySQL, extraire des fichiers de configuration Apache, ou installer des backdoors.
          b) **mysql_sql :** Ce module de post-exploitation permet d'exécuter des commandes SQL sur une base de données MySQL compromise, facilitant l'extraction de données sensibles.
    v. Rapport
       1) Permet d'exporter les résultats des exploits et des sessions de post-exploitation, qui peuvent ensuite être intégrés dans des rapports de pentest.
          a) **db_export :** Exporter les données recueillies pendant le pentest depuis la base de données de Metasploit vers des formats de fichier utilisables dans des rapports de pentest.

Autres outils (* non inclus dans Kali) :

- Burp Suite : le scanner de vulnérabilités professionnel bien connu des hackers.
- Nessus *, OpenVas *, OSWAP Zed Attack Proxy (ZAP)
- Maltego : un logiciel qui permet de collecter des informations sur une personne ou une organisation, bien utile pour faire de l'ingénierie sociale ;
- aircrack-ng : un outil de crack de réseaux WiFi

Suggestions d'outils Supplémentaires (* non inclus dans Kali) :

- Wireshark : Un analyseur de protocole réseau qui permet de capturer et d'interpréter le trafic réseau en temps réel. Indispensable pour l'analyse des problèmes de réseau et la détection des anomalies.
- John the Ripper : Un puissant cracker de mots de passe, utile pour tester la robustesse des mots de passe dans les systèmes et applications.
- Ghidra : Un outil d'analyse de logiciels développé par la NSA. Il est particulièrement utile pour le reverse engineering de logiciels malveillants et de systèmes.
- **OWASP Dependency-Check * :** Un outil qui identifie les dépendances de projet avec des vulnérabilités connues, très utile pour le développement sécurisé d'applications.
- GitLeaks * : Outil de détection de fuites d'informations sensibles dans les dépôts Git, idéal pour prévenir les fuites de données accidentelles.

- Techniques d'exploitation et d'escalade de privilèges.

Techniques d'Exploitation

L'exploitation consiste à utiliser des vulnérabilités dans les logiciels, les applications ou les systèmes pour exécuter du code, obtenir un accès non autorisé ou recueillir des informations sensibles. Voici quelques techniques courantes :

- Injection de code : L'attaquant injecte du code malveillant (par exemple, SQL, XSS, Command Injection) dans une application pour exécuter des commandes non autorisées.
- Débordement de tampon : Exploite les erreurs de gestion de la mémoire dans les applications pour exécuter du code arbitraire.
- Cross-Site Scripting (XSS) : Injecte des scripts malveillants dans des pages web vues par d'autres utilisateurs pour voler des informations ou effectuer des actions en leur nom.
- Cross-Site Request Forgery (CSRF) : Induit en erreur un utilisateur pour qu'il exécute des actions non désirées sur une application web où il est authentifié.
- Deserialization Unsafe : Exploite des vulnérabilités dans le processus de désérialisation pour exécuter du code arbitraire.

Techniques d'Escalade de Privilèges

L'escalade de privilèges se produit lorsque l'attaquant obtient un niveau d'accès plus élevé que celui initialement accordé, souvent en exploitant des vulnérabilités dans le système ou les configurations.

- Exploitation des configurations erronées : Utilisation de mauvaises configurations système ou de permissions trop larges pour obtenir un accès élevé.
- Abus de services vulnérables : Exploitation de vulnérabilités dans les services exécutés pour élever les privilèges.
- Injection de DLL : Injection de bibliothèques dynamiques dans des processus pour exécuter du code avec les privilèges du processus ciblé.
- Pass-the-hash / Pass-the-ticket : Utilisation des hachages d'authentification ou des tickets Kerberos volés pour s'authentifier en tant qu'utilisateur sans connaître le mot de passe.

Outils pour l'Exploitation et l'Escalade de Privilèges

- Metasploit : Un framework d'exploitation qui fournit des modules pour tester l'exploitabilité des vulnérabilités et pour l'escalade de privilèges.
- BeEF (Browser Exploitation Framework) : Utilisé pour exploiter les vulnérabilités XSS et autres vulnérabilités côté client.
- PowerSploit : Une collection de scripts PowerShell pour la post-exploitation, y compris l'escalade de privilèges sur des systèmes Windows.
- Mimikatz : Outil célèbre pour extraire des mots de passe, des hachages et d'autres secrets du système Windows, facilitant l'escalade de privilèges et le mouvement latéral.
- LinEnum / PEASS - Privilege Escalation Awesome Scripts SUITE (LinPEAS, WinPEAS) : Scripts pour identifier les vulnérabilités et les configurations erronées qui peuvent être exploitées pour l'escalade de privilèges sur Linux et Windows.

- Sources :
  - [Test d'intrusion : approche, méthodologie, types de tests et prix - Vaadata](https://www.vaadata.com/blog/fr/test-dintrusion-approche-methodologie-types-de-tests-et-prix/)

D. **Analyse et Rapport des Résultats**

- Interprétation des résultats.

L'interprétation des résultats est une étape critique qui suit l'exécution des tests d'intrusion. Elle implique d'analyser les données recueillies pour classer et identifier les vulnérabilités, comprendre leur impact potentiel en terme de sécurité, et évaluer le risque global pour l'organisation.

- Analyse des Vulnérabilités : Examiner chaque vulnérabilité détectée pour comprendre sa nature, sa sévérité, et les conditions d'exploitabilité.
- Évaluation de l'Impact : Déterminer l'impact potentiel de chaque vulnérabilité sur les actifs de l'organisation, en tenant compte de la probabilité d'exploitation et des conséquences d'une attaque réussie.
- Priorisation des Risques : Classer les vulnérabilités identifiées en fonction de leur sévérité et de leur impact potentiel pour aider l'organisation à allouer ses ressources de manière efficace.

Matrice d'Analyse des Vulnérabilités :

- Une matrice des vulnérabilités synthétise les vulnérabilités découvertes après un pentest.
- Informations à remplir : Identifiant, Description, Sévérité, Impact, Exploitabilité, Recommandations, Statut

Matrice des vulnérabilités :

ID Vulnérabilité	Description	Sévérité	Impact	Exploitabilité	Recommandations	Statut de Correction
VULN001	Exemple de vulnérabilité dans Apache	Critique	Élevé	Facile	Mettre à jour vers la dernière version	En cours
VULN002	Exemple de vulnérabilité dans MySQL	Moyenne	Modéré	Moyen	Appliquer le patch XYZ	Planifié
VULN003	Exemple de vulnérabilité dans l'Application en PHP : injection SQL sur le formulaire de login	Critique	Très Élevé	Facile	Consolider les développement : filtrage htmlentities, etc… Requête préparée	Corrigé
VULN004	Exemple de vulnérabilité dans l'Application en PHP : XSS sur la table des articles	Critique	Élevé	Moyen	Consolider les développement : filtrage htmlentities, etc…	En cours

Matrice des risques :

La priorisation des vulnérabilités dans le cadre d'un pentest peut-être dérivée d'une évaluation des risques, qui prend en compte à la fois la sévérité de la vulnérabilité et son impact potentiel sur l'organisation. Cette approche permet de s'assurer que les efforts seront concentrés là où ils sont le plus nécessaires, optimisant ainsi l'utilisation des ressources de sécurité.

- Une matrice des risques synthétise les risques potentiels d'une vulnérabilité en fonction de sa sévérité et de son impact.
- Elle facilite la prise de décision concernant les mesures correctives à appliquer en priorité.
- Comment la Priorité est-elle Obtenue ?
    - La priorité est généralement obtenue en croisant deux paramètres principaux dans la matrice des risques :
    - - **Sévérité (ou Probabilité)** : Indique à quel point il est facile d'exploiter la vulnérabilité. Elle peut être évaluée en fonction de facteurs tels que la complexité de l'exploitation, les compétences requises pour l'attaquant, et la disponibilité des outils d'exploitation.
    - - **Impact** : Représente les conséquences potentielles de l'exploitation de la vulnérabilité sur l'organisation. L'impact peut être mesuré en termes de perte financière, de dommages à la réputation, d'interruption des opérations, ou de compromission des données sensibles.
    - La combinaison de ces deux dimensions permet de classer en catégories de risque : Très élevé, Elevé, Moyen, Faible, Très faible).
- Matrice des risques : Priorité = Sévérité*Impact

Sévérité/Impact	Moyen	Élevé	Très Élevé

Critique	Moyen	Élevé	Très Élevé
Moyenne	Faible	Moyen	Élevé
Faible	Très Faible	Faible	Moyen

- **Très Élevé, Élevé** : Priorité immédiate => nécessité d'agir rapidement.
- **Moyen** : À corriger après les vulnérabilités de priorité élevée.
- **Faible, Très Faible** : À surveiller ou corriger selon les ressources disponibles.

- Recommandations et mesures correctives.

Les recommandations fournissent des orientations sur la manière de remédier aux vulnérabilités identifiées et de renforcer la posture de sécurité de l'organisation.

- Mesures Correctives Spécifiques : Pour chaque vulnérabilité, proposer des mesures correctives spécifiques, en indiquant des solutions possibles, des mises à jour de sécurité, des configurations à modifier, ou des pratiques à adopter.
- Plan d'Action : Élaborer un plan d'action priorisé pour les corrections, en tenant compte des ressources disponibles et de l'impact sur les opérations de l'organisation.
- Suivi et Réévaluation : Recommander un suivi et une réévaluation réguliers pour s'assurer que les mesures correctives sont mises en œuvre et pour détecter de nouvelles vulnérabilités potentielles.

Matrice d'un plan d'actions et des responsabilités

- Une matrice synthétise les actions à entreprendre suite aux vulnérabilités découvertes, elle définit les priorités, les équipes ou les personnes responsables de l'exécution de l'action, ainsi que son délai de correction attendu et son statut d'avancement.
- Informations à remplir : Identifiant, Recommandation, Priorité, Responsable, Délai, Statut

Matrice du plan d'action (Matrice RACI*Matrice des Risques) :

ID Vulnérabilité	Recommandation	Priorité	Responsable	Délai	Statut
VULN001	Mettre à jour Apache vers la dernière version	Elevé	Équipe web	30 jours	En cours
VULN002	Appliquer le patch de sécurité XYZ pour MySQL	Faible	Administrateur DB	60 jours	Planifié
VULN003	Consolider les développement : filtrage htmlentities, etc… Requête préparée	Très Élevé	Équipe développement	15 jours	Corrigé
VULN004	Consolider les développement : filtrage htmlentities, etc…	Elevé	Équipe développement	30 jours	En cours

- Rédaction de rapports de pentest.

Le rapport de pentest documente les découvertes, l'analyse et les recommandations issues du test d'intrusion. Il doit être clair, précis, et compréhensible et adapter aux différents publics (Direction générale, DSI, équipes informatique.

- Structure du Rapport : Un rapport typique inclut un résumé exécutif, la portée du test, la méthodologie utilisée, les vulnérabilités découvertes avec des preuves, l'analyse de l'impact, et les recommandations.
- Clarté et Précision : Utiliser un langage adaptée aux interlocuteurs. Fournir des preuves concrètes, comme des captures d'écran ou des logs, pour étayer les découvertes.
- Confidentialité : Traiter le rapport comme un document sensible, car il contient des informations détaillées sur les vulnérabilités de l'organisation.

Structure du Rapport de Pentest :

1. Résumé Exécutif (pour la Direction Générale)

- Objectif : Fournir une vue d'ensemble non technique des résultats, des risques principaux, et des recommandations.
- Contenu: Un tableau récapitulatif des risques majeurs, avec une évaluation de l'impact sur l'entreprise.
    - Matrice des risques majeurs issue du plan d'action

- Format: Langage clair, sans jargon technique, axé sur les implications commerciales et les mesures correctives stratégiques.

2. Portée du Test et Méthodologie (pour le DSI et les Équipes Techniques)
- Objectif : Détail de la portée du pentest, des objectifs, et de la méthodologie utilisée.
- Contenu :
    - Matrice des Vulnérabilités
    - Matrice des risques
    - Matrice du plan d'action
    - Portée du Pentest : Liste des systèmes, applications, et réseaux testés.
- Méthodologie : Description des étapes du pentest(**reconnaissance, scanning, exploitation, post-exploitation**),des outils utilisés, et des approches (boîte noire, blanche, grise
- Format : Plus technique, incluant des détails spécifiques sur les tests réalisés.

3. Détails des Vulnérabilités et Analyse d'Impact (pour les Équipes Techniques)
- Objectif : Présenter en détail chaque vulnérabilité, avec des preuves et une analyse d'impact.
- Contenu :
    - Matrice des Vulnérabilités : Tableau listant les vulnérabilités, avec colonnes pour ID, description, sévérité, impact, et preuves (captures d'écran, logs).
- Analyse d'Impact : Discussion sur les implications de chaque vulnérabilité sur la sécurité et les opérations.
- Format : Technique, avec des preuves concrètes et une évaluation précise de l'impact.

4. Recommandations et Plan d'Action (pour le DSI et les Équipes Techniques)
- Objectif : Fournir des recommandations spécifiques pour remédier aux vulnérabilités identifiées.
- Contenu :
    - Matrice de Recommandations : Tableau avec des recommandations, priorités, responsables, et délais.
- Plan d'Action : Étapes proposées pour les corrections, avec une timeline et des indicateurs de suivi.
- Format : Pragmatique, avec des actions claires et des priorités basées sur l'impact et la sévérité.

5. Annexes (pour les Équipes Techniques)
- Objectif : Fournir des informations supplémentaires et des preuves techniques.
- Contenu : Détails techniques supplémentaires, scripts utilisés, réponses complètes des systèmes, configurations testées.
- Format : Très technique.

- Sources:
- [Effectuer un Pentest ou Test d'Intrusion pour améliorer la sécurité - Gplexpert](https://gplexpert.com/effectuer-pentest-test-intrusion/)