Préambule

Ce TP porte sur les pratiques proactives de sécurité web, visant à intégrer la sécurité dans le cycle de développement logiciel de manière continue et automatisée. Vous découvrirez des outils et méthodes qui permettent de détecter et corriger les vulnérabilités dès les phases de développement et de déploiement.

Contrairement aux précédents TP, ce travail peut être réalisé en binôme. Travailler à deux vous permettra de partager les rôles, d'échanger sur les bonnes pratiques et de mieux appréhender la collaboration entre développeurs, opérateurs et équipes sécurité.

Votre rapport devra présenter vos travaux, incluant les étapes réalisées, les outils utilisés, vos analyses critiques sur la sécurisation automatisée, ainsi que vos recommandations pour une culture DevSecOps efficace.

Merci d'envoyer votre rapport et les annexes dans une archive ZIP par email à : fpitance@bunkerity.com

Mise en place de l'environnement GitHub

Cette étape vise à créer un dépôt GitHub pour votre projet et à initialiser un pipeline d'intégration continue automatisée grâce à GitHub Actions. Ce pipeline servira de base à l'intégration des outils de sécurité dans les étapes suivantes.

Actions à réaliser :

1) Créez un nouveau dépôt GitHub à partir de votre compte personnel
2) Initialisez-le avec un projet simple (exemple : une petite application web en Python, Node.js, ou tout autre langage de votre choix)
3) Poussez votre code source dans ce dépôt
4) Réalisez le quickstart sur les GitHub actions : https://docs.github.com/en/actions/get-started/quickstart

La validation consiste à s'assurer que l'action est automatiquement déclenchée lors d'un push sur le dépôt (cf. onglet actions sur GitHub).

Intégration d'outils de sécurité automatisée

Objectif : automatiser la détection des vulnérabilités dans le code source et les dépendances grâce à des outils embarqués dans le pipeline CI/CD, afin de garantir une sécurité continue dès les phases de développement.

Actions à réaliser :

- **Configuration de CodeQL pour l'analyse statique de code (SAST) :**
  - Ajoutez un job dans votre workflow GitHub Actions pour lancer CodeQL.
  - Utilisez le template officiel GitHub pour scanner le code source selon le langage utilisé dans votre projet.
  - Exécutez l'analyse sur chaque push et examinez les rapports générés dans l'onglet Sécurité du dépôt.
- **Activation de Dependabot pour la gestion des dépendances :**
  - Activez Dependabot dans les paramètres du dépôt pour vérifier automatiquement la sécurité des bibliothèques utilisées.
  - Définissez la fréquence de vérification des vulnérabilités (ex : hebdomadaire).
  - Surveillez les alertes et les pull requests automatiques pour mise à jour des dépendances vulnérables.
- **Exploitation des résultats :**
  - Analysez les vulnérabilités détectées par CodeQL et Dependabot.
  - Commencez à corriger ou à documenter les potentielles failles dans votre code ou les dépendances.

Ressources complémentaires :

- Documentation CodeQL : https://docs.github.com/en/code-security/secure-coding
- Guide Dependabot : https://docs.github.com/en/code-security/dependabot/dependabot-alerts
- Exemple d'intégration CodeQL dans un workflow : https://github.com/github/codeql-action

Important : il faudra faire en sorte de « forcer » la présence de vulnérabilités dans votre code ainsi que la présence de dépendances vulnérables / qui n'est pas à jour.

Validations :

- Vérifiez que CodeQL s'exécute automatiquement à chaque push et produit un rapport d'analyse dans l'onglet Sécurité.
- Confirmez l'apparition d'alertes Dependabot si des vulnérabilités dans les dépendances sont détectées.
- Documentez dans votre rapport les vulnérabilités trouvées, les actions entreprises et les recommandations.

Sensibilisation par la pratique

Objectif : renforcer la compréhension des vulnérabilités web et des bonnes pratiques de sécurité à travers des activités ludiques et pédagogiques, afin de mieux assimiler les risques et leurs mitigations.

Actions à réaliser :

- **Création d'un quiz interactif (exemple : Kahoot) :**
  - Préparez un quiz comportant des questions sur les vulnérabilités web, leurs impacts et les mesures de prévention.
- **Création d'un mini CTF (Capture The Flag) :**
  - Mettez en place un petit CTF avec 3 challenges basés sur :
    - Injection SQL simple.
    - Upload de fichier malveillant.
    - Exécution de commande shell via vulnérabilité.
  - Fournissez les instructions de résolution et les éléments nécessaires pour chaque challenge.
- **Déploiement de la plateforme CTFd :**
  - Installez et configurez CTFd, une plateforme open source pour organiser des CTF.
  - Intégrez vos challenges au sein de CTFd afin que des formés puissent les résoudre dans un environnement commun.
  - Mettez à disposition des ressources pédagogiques (documents, liens) pour accompagner la résolution.