QUALITÉ ET SÉCURITÉ DES APPLICATIONS : SÉCURISER UNE APPLICATION

Durée	**3 jours**	**Référence Formation**	**4-JA-SEA**

Objectifs
Connaitre les différents types d'attaques (attaques par injection SQL, attaques XSS, attaques CSRF, attaques "brute force…) et les moyens à mettre en œuvre pour s'en prémunir

Participants
Cette formation s'adresse aux développeurs souhaitant connaître les différentes techniques de sécurisation d'une application

Pré-requis
Pour suivre ce stage, il est nécessaire d'avoir une bonne connaissance de la programmation orientée objet et de la programmation d'applications Web

Moyens pédagogiques
Accueil des stagiaires dans une salle dédiée à la formation équipée d'un vidéo projecteur, tableau blanc et paperboard ainsi qu'un ordinateur par participant pour les formations informatiques.
Positionnement préalable oral ou écrit sous forme de tests d'évaluation, feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation.
En fin de stage : QCM, exercices pratiques ou mises en situation professionnelle, questionnaire de satisfaction, attestation de stage, support de cours remis à chaque participant.
Formateur expert dans son domaine d'intervention
Apports théoriques et exercices pratiques du formateur
Utilisation de cas concrets issus de l'expérience professionnelle des participants
Réflexion de groupe et travail d'échanges avec les participants
Pour les formations à distance : Classe virtuelle organisée principalement avec l'outil ZOOM. Assistance technique et pédagogique : envoi des coordonnées du formateur par mail avant le début de la formation pour accompagner le bénéficiaire dans le déroulement de son parcours à distance.

PROGRAMME

Concepts de sécurité logicielle
- Pourquoi sécuriser une application
- Identifier et comprendre les vulnérabilités de vos applications attaques « brute-force »
- Attaques par « déni de services » (DOS - Denial Of Service)
- Attaques par analyse de trames IP
- Attaques par « Injection SQL »
- Attaques « XSS » (Cross site scripting)
- Attaques « CSRF » (Cross site request forgery)
- Autres types d'attaques
- Outils de détection de faille de sécurité

- Travaux pratiques : tests de ces différents types de problèmes sur une application mal développée et utilisation des outils de détection de faille de sécurité

Validation des données entrantes
- Protection contre les entrées d'utilisateurs nuisibles
- Utilisation d'expressions régulières
- Détecter et contrer les « injections SQL »
- Détecter et contrer les attaques « XSS »
- Détecter et contrer les attaques « CSRF »
- Détecter et contrer les attaques « bruteforce »
- Sécuriser les données en Cookie
- Protection contre les menaces de déni de service
- Ne pas présenter à l'utilisateur les détails des erreurs techniques
- Travaux pratiques : modification du code de l'application initialement proposée pour interdire ces différents types d'attaques

Sécuriser les données stockées en base
- Authentification et Autorisation du SGBDr (Système de Gestion de Base de Données relationnelle)
- Rôles serveur et rôles de base de données
- Propriété et séparation utilisateur schéma
- Chiffrement de données dans la base de données
- Travaux pratiques : stocker de manière sécurisée les mots de passe en base de données

Sécuriser le système de fichier
- Crypter les données sensibles dans les fichiers de configuration
- Détecter les tentatives de remplacement des fichiers sources de l'application "signer les fichiers"
- Protéger les informations des fichiers de log

Oauth 2.0 et l'authentification au niveau du navigateur
- Présentation de l'architecture Oauth 2.0
- Utilisation de l'API Oauth 2.0
- Travaux pratiques : mise en œuvre de Oauth

Sécuriser les échanges de données
- Modèle de chiffrement
- Conception orientée flux
- Configuration du chiffrement
- Choix d'un algorithme
- Mettre en œuvre le chiffrage symétrique
- Mettre en œuvre le chiffrage asymétrique
- Travaux pratiques : réaliser une communication sécurisée à l'aide d'un certificat