Sécurisation des applications, les méthodes de développement

Cours Pratique de 2 jours - 14h
Réf : APD - Prix 2024 : 1 660CHF HT

Avec l'explosion du digital qui a multiplié les opportunités de développement, la sécurité dans la réalisation de logiciels est devenue un enjeu majeur pour les entreprises. Cette formation très riche vous apprendra méthodes et solutions nécessaires permettant d'assurer et tester la sécurité dans vos développements.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Maîtriser le modèle de maturité pour le développement d'applications sécurisées OpenSAMM

Savoir réaliser une analyse de la sécurité du logiciel audité

Identifier les parties essentielles du code source à vérifier

Tester la sécurité des applications

LE PROGRAMME

dernière mise à jour : 10/2018

1) Introduction

- Qu'est-ce que la sécurisation du code ?
- Les acteurs de la sécurité : le CERT, l'OWASP, Le BSIMM…
- Quels sont les risques liés au développement d'une application ?
- Les traces laissées par les développeurs : mémoire, journaux…
- Qu'est-ce que le codage sécurisé d'une application ?
- Les types d'attaques.

2) La sécurité des applications avec OpenSAMM

- Le modèle de maturité pour le développement d'applications sécurisées.
- Les 4 niveaux de maturité.
- Niveau implicite de départ.
- Compréhension initiale et mise en place de pratiques de sécurité.
- Amélioration de l'efficacité/efficience des pratiques de sécurité.
- Maîtrise complète des pratiques de sécurité.

3) Mise en place d'OpenSAMM

- Préparer.
- Evaluer.
- Définir la cible souhaitée.
- Définir le plan.
- Mettre en place.
- Mettre à disposition.

Travaux pratiques : Calcul du niveau de maturité d'une organisation.

4) Introduction à BSIMM

- Qu'est-ce que le BSIMM (Building Security In Maturity Model) ?

PARTICIPANTS

Développeurs, architectes applicatifs, chefs de projets amenés à sécuriser des applications.

PRÉREQUIS

Connaître le guide d'hygiène sécurité de l'ANSSI. Avoir suivi le parcours introductif à la cybersécurité. Connaissance d'un langage de programmation.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques…
Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
• À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Constituer une base solide pour le développement d'une application.
- Les bonnes pratiques.

5) Analyse de la sécurité de l'application auditée.

- Identifier les parties critiques de son code.
- Définir le périmètre de l'audit et se limiter aux parties critiques.

6) Les parties essentielles du code source à vérifier

- Identifier les parties du code source essentielles à vérifier.
- Les mécanismes d'authentification et cryptographiques.
- La gestion des utilisateurs.
- Le contrôle d'accès aux ressources.
- Les mécanismes d'interactions avec d'autres applications.
- L'accès aux bases de données.
- La conformité des exigences de sécurité établies pour l'application.

Travaux pratiques : Exemple d'identification des parties du code source essentielles à vérifier.

7) Tester la sécurité des applications

- Identifier les parties du code source essentielles à vérifier.
- Les processus projet et les tests.
- L'approche globale.
- Le plan de test et ses déclinaisons. La stratégie de test.
- L'approche par les risques. L'estimation.

Travaux pratiques : Exemple de test d'une application.

LES DATES

CLASSE À DISTANCE
2024 : 03 juin, 09 sept., 05 déc.