Sécurité défensive

Sécurité applicative avec PHP

3 jours (21h00) | ⭐⭐⭐⭐⭐ 5/5 | SEC-PHP | Évaluation qualitative de fin de stage | Formation délivrée en présentiel ou distanciel

Contenu mis à jour le 18/10/2024. Document téléchargé le 08/12/2024.

Objectifs de formation

A l'issue de cette formation, vous serez capable de :

- Sécuriser un code PHP ou une interface avec du PHP
- Créer des tests visant à éprouver la sécurité des applications Web, notamment sous PHP
- Analyser et organiser la sécurité d'une application Web développée en PHP
- Formuler des exigences de sécurité aux autres corps de métiers.

Modalités, méthodes et moyens pédagogiques

Formation délivrée en présentiel ou distanciel* (blended-learning, e-learning, classe virtuelle, présentiel à distance).

Le formateur alterne entre méthode** démonstrative, interrogative et active (via des travaux pratiques et/ou des mises en situation).

Variables suivant les formations, les moyens pédagogiques mis en oeuvre sont :

- Ordinateurs Mac ou PC (sauf pour certains cours de l'offre Management), connexion internet fibre, tableau blanc ou paperboard, vidéoprojecteur ou écran tactile interactif (pour le distanciel)
- Environnements de formation installés sur les postes de travail ou en ligne
- Supports de cours et exercices

En cas de formation intra sur site externe à M2i, le client s'assure et s'engage également à avoir toutes les ressources matérielles pédagogiques nécessaires (équipements informatiques...) au bon déroulement de l'action de formation visée conformément aux prérequis indiqués dans le programme de formation communiqué.

* *nous consulter pour la faisabilité en distanciel*

** *ratio variable selon le cours suivi*

Prérequis

Avoir des connaissances généralistes en programmation Web et en langage PHP. Avoir connaissance de la gestion de base d'un serveur Web est un plus.

Public concerné

Pentesters et développeurs.

Cette formation :

- Est animée par un consultant-formateur dont les compétences techniques, professionnelles et pédagogiques ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou par M2i Formation
- Bénéficie d'un suivi de son exécution par une feuille de présence émargée par demi-journée par les stagiaires et le formateur.

Programme

Jour 1

Introduction

- Panorama de la sécurité Web
- Les normes et lois
- Les référentiels
- Les principaux groupes de réflexion et de travail sur la sécurité des applications Web
- L'évolution du langage PHP, des technologies et des usages du Web
- L'apport du Full Stack, de DevOps et DevSecOps

Protocole HTTP avec PHP

- Rappel des fondamentaux sur les protocoles HTTP et HTTP/2
  - La pile applicative
  - Les méthodes
  - Les codes erreurs
  - Les principaux champs liés à la sécurité
- Le fonctionnement d'AJAX
- Architecture des applications Web (monolithe, N-tiers, SOA / ROA...)
- Le cas des API
- Le header HTTP et la fonction header()
- Le retour de requête via http_response_code()
- Les méthodes HTTP via le module cURL pour PHP

Exemple de travaux pratiques (à titre indicatif)

- *Création d'une requête GET et d'une requête POST en PHP avec cURL*

Les outils connexes

- Faire des tests et des validations
  - Les devtools des navigateurs
  - La capture via proxy
  - La capture via tcpdump ou Wireshark
  - Les tests avec Postman
- Les "vulnerability scanners"
  - Burp Suite
  - Acunetix
  - Les outils spécialisés CMS
  - Les sites Web d'analyse

Exemples de travaux pratiques (à titre indicatif)

- *Capture d'une requête GET et d'une requête POST vers Google*
- *Simulation d'une recherche Google avec Postman*

L'OWASP

- Présentation de l'OWASP et de ses projets
- Le PHP Security Cheat Sheet
- Le Top 10
- Le Top 25 du SANS
- Les Google Dorks
- Les guides de l'OWASP (Test Guide, Dev Guide...)
- L'ASVS (Application Security Verification Standard)
- Rappel des recommandations
  - Sur le logging et le monitoring
  - En termes de suivi des vulnérabilités
- Les grandes familles d'attaques
  - Broken Access Control
  - Cryptographic Failures
  - Injection
  - Insecure Design
  - Security Misconfiguration
  - Vulnerable and Outdated Components
  - Identification and Authentication Failures
  - Software and Data Integrity Failures
  - Security Logging and Monitoring Failures
  - Server-Side Request Forgery

Jour 2

Exemples de travaux pratiques (à titre indicatif)

- *Exploitation*
  - *D'une injection d'entête HTTP*
  - *D'une injection SQL*
  - *D'une Cross-Site Scripting*
  - *D'une Cross-Site Request Forgery*
  - *D'une Server-Side Request Forgery*
  - *D'un vol de session*
  - *D'une désérialisation*
  - *D'une vulnérabilité dans un produit Open Source (WordPress, Joomla, Drupal, Magento...)*
- *Démonstration de la rétro-ingénierie sur une plateforme souffrant de "misconfiguration"*
- *Exploitation d'une faille LFI / RFI*
- *Analyse automatisée via des "vulnerability scanners" (WPScan, Nikto, OpenVas, Nmap) et un framework offensif (Metasploit, BeEF...)*
- *Réalisation et exécution d'un "stress test" d'une application Web*

La sécurité du code externe

- Le cas des librairies
- Le "pruning"
- La gestion de la mise à jour

Jour 3

Les bonnes pratiques pour le renforcement de la sécurité du code

- Les forces et les faiblesses du langage PHP
- La validation et assainissement des entrées
- La sécurisation de l'authentification
- L'authentification multifactorielle et par challenge

- La gestion des mots de passe (création, stockage et vérification)
- La gestion des sessions (création, stockage et vérification)
- La gestion des droits et du contrôle d'accès
- La norme RBAC (Role-Based Access Control) appliquée aux applications Web
- La validation et assainissement des sorties
- Le cas des downloaders et uploaders
- La gestion des CSRF (Cross-Site Request Forgery)
- La gestion du logging et du monitoring serveur et applicatif
- L'utilisation de composants sécurisés

Exemples de travaux pratiques (à titre indicatif)

- *Création d'un portail d'authentification sécurisé en PHP*
- *Création d'un downloader sécurisé*
- *Analyse de code statique et dynamique avec OWASP ZAP, SonarQube, RIPS ou Accunetix*

Le renforcement de la sécurité côté client et serveur

- La sécurité du système d'exploitation
  - L'installation
  - Les règles obligatoires pour le firewall
  - Les antivirus
  - Les IPS (Intrusion Prevention System) et IDS (Intrusion Detection System)
  - Le logging et le monitoring
  - La surveillance de l'intégrité
  - Les tâches de maintenance
- La sécurité du serveur Web
  - Les réglages de base
  - Les modules complémentaires
  - La limitation des droits
  - La gestion des logs
- Le langage PHP et HTML
  - Le fichier php.ini
  - Les fonctions phpinfo() et PHPSecInfo
  - La Content Security Policy
  - L'utilisation de SOP / CORS
- La confidentialité et l'intégrité
- Le chiffrement SSL/TLS
- Les certificats
- Outils et technologie de sécurité (firewall, IPS, IDS...)

Le contenu de ce programme peut faire l'objet d'adaptation selon les niveaux, prérequis et besoins des apprenants.

Modalités d’évaluation des acquis

- En cours de formation, par des études de cas ou des travaux pratiques
- Et, en fin de formation, par un questionnaire d'auto-évaluation

Accessibilité de la formation

Le groupe M2i s'engage pour faciliter l'accessibilité de ses formations. Toutes nos formations sont accessibles aux personnes en situation de handicap : les détails de l'accueil des personnes sont consultables sur la page Accueil PSH.

Modalités et délais d’accès à la formation

Les formations M2i sont disponibles selon les modalités proposées sur la page programme.
Les inscriptions sont possibles jusqu'à 48 heures ouvrées avant le début de la formation.
Dans le cas d'une formation financée par le CPF, ce délai est porté à 11 jours ouvrés.