Programme de cours
Sécurité avancée Github (GitHub Advanced Security)

Durée :	1 jour soit 7 heures	Référence :	GH-500T00

Public visé :

Ce cours destiné aux étudiants qui souhaitent comprendre et implémenter des pratiques de sécurité avancées avec l'aide de GitHub Advanced Security (GHAS). Ils apprendront à améliorer considérablement les processus de développement de logiciels et à créer un écosystème de développement plus résilient et sécurisé grâce à des solutions centrées sur les développeurs, afin de maintenir le code, la chaîne d'approvisionnement et les informations sensibles en sécurité avant de les déployer en production. Ils apprendront comment GHAS offre aux équipes de sécurité une visibilité sur la posture de sécurité inter-organisationnelle et la sécurité de la chaîne d'approvisionnement, ainsi qu'un accès sans précédent aux renseignements de sécurité soigneusement organisés provenant de millions de développeurs et de chercheurs en sécurité dans le monde entier.

Pré-requis :

Objectifs pédagogiques :

- Comprendre les fonctionnalités de GitHub Advanced Security
- Obtenez les compétences nécessaires pour reconnaître, appliquer et évaluer ces fonctionnalités dans votre propre environnement GitHub

Compétences acquises à l'issue de la formation :

Modalités pédagogiques :

Session dispensée en présentiel ou téléprésentiel, selon la modalité inter-entreprises ou intra-entreprises sur mesure. La formation est animée par un(e) formateur(trice) durant toute la durée de la session et présentant une suite de modules théoriques clôturés par des ateliers pratiques validant l'acquisition des connaissances. Les ateliers peuvent être accompagnés de Quizz. L'animateur(trice) présente la partie théorique à l'aide de support de présentation, d'animation réalisée sur un environnement de démonstration. En présentiel comme en téléprésentiel, l'animateur(trice) accompagne les participants durant la réalisation des ateliers.

Note relative au programme : Les ateliers, démonstrations, travaux pratiques et scénarios présentés dans ce programme sont fournis à titre d'exemples illustratifs et ne sont pas exhaustifs. Afin de garantir une adéquation permanente avec les évolutions technologiques, le contenu pédagogique ainsi que les environnements de travaux pratiques (TP) sont susceptibles d'être adaptés, ajustés ou mis à jour au cours du temps.

Moyens et supports pédagogiques :

Cadre présentiel

Salles de formation équipées et accessibles aux personnes à mobilité réduite.

- Un poste de travail par participant
- Un support de cours numérique ou papier (au choix)
- Un bloc-notes + stylo
- Vidéoprojection
- Connexion Internet
- Accès extranet pour partage de documents et émargement électronique

Cadre téléprésentiel

Session dispensée via notre solution iClassRoom s'appuyant sur Microsoft Teams.

- Un compte Office 365 par participant
- Un poste virtuel par participant
- Un support numérique (PDF ou Web)

- Accès extranet pour partage de documents et émargement électronique

Modalités d'évaluation et de suivi :

Avant

Afin de valider le choix d'un programme de formation, une évaluation des prérequis est réalisée à l'aide d'un questionnaire en ligne ou lors d'un échange avec le formateur(trice), qui valide la base de connaissances nécessaires.

Les stagiaires disposent également, depuis leur extranet, d'un **outil d'auto-positionnement** leur permettant d'évaluer leur niveau de maîtrise des compétences visées par la formation. Cette auto-évaluation est réalisée avant le démarrage de la formation à l'aide d'un curseur de positionnement allant du niveau le plus faible au niveau le plus élevé.

Pendant

Après chaque module théorique, un ou plusieurs ateliers pratiques permettent de valider progressivement l'acquisition des connaissances. Un quiz peut accompagner les ateliers pratiques afin de mesurer la compréhension des notions abordées.

Après

À l'issue de la formation, les stagiaires réalisent une nouvelle **auto-évaluation** sur les mêmes compétences que celles évaluées avant la formation. Cette comparaison permet de mesurer leur perception de la progression réalisée et de visualiser les compétences développées au cours de la formation.

Lorsque le programme de formation le prévoit, un examen de certification est organisé conformément aux modalités définies par l'éditeur ou le centre de certification (TOSA, Pearson VUE, ENI, PeopleCert…).

Le formateur réalise également une **évaluation individuelle des compétences** de chaque stagiaire sur l'ensemble des compétences visées par la formation. Pour chacune d'elles, il indique si la compétence est :

- Acquise ;
- En cours d'acquisition ;
- Non acquise.

Lorsque la compétence est évaluée comme **en cours d'acquisition** ou **non acquise**, le formateur formule des observations précisant les difficultés rencontrées ainsi que les actions ou recommandations permettant au stagiaire de poursuivre sa montée en compétences.

Enfin

Un questionnaire de satisfaction permet au participant d'évaluer la qualité de la prestation de formation.

Informations sur l'accessibilité :

Nos formations sont, dans la mesure du possible, conçues pour être accessibles à toutes et à tous. Afin de garantir les meilleures conditions d'accueil et d'apprentissage pour les personnes en situation de handicap, nous vous invitons à contacter notre référente handicap certifiée :
Céline SOLATGES – 05 61 34 39 80 – csolatges@iform.fr
Nous vous remercions de bien vouloir nous communiquer toute information utile à ce sujet en amont de la formation, afin de mettre en place les adaptations nécessaires et d'assurer un accompagnement optimal.

Pour en savoir plus sur les dispositifs d'accompagnement existants, vous pouvez consulter les sites suivants :

- AGEFIPH
- FIPHFP
- MDPH de votre département

Description / Contenu

GitHub Advanced Security (GHAS) joue un rôle crucial dans l'amélioration de la posture de sécurité des projets de développement de logiciels sur GitHub. Il fournit un ensemble complet d'outils et de fonctionnalités conçus pour identifier et résoudre les vulnérabilités de sécurité tout au long du cycle de vie du développement. En intégrant la sécurité directement dans le processus de développement avec GHAS, votre équipe peut créer des logiciels plus sécurisés et fiables. Le cours explorera comment utiliser GHAS pour maximiser l'impact sur la sécurité et comprendre GHAS et son rôle dans l'écosystème de sécurité.

Module 1 : Présentation de GitHub Advanced Security
Ce module vous aidera à vous familiariser avec les fonctionnalités et les meilleures pratiques de GitHub Advanced Security (GHAS). À mesure que vous découvrirez

ces fonctionnalités, vous identifierez les points critiques pour éliminer les failles de sécurité.

- Définir GHAS et l'importance de ses fonctionnalités fondamentales
- Comment utiliser GHAS pour maximiser son impact
- Comprendre GHAS et son rôle dans l'écosystème de sécurité

Module 2 : Configurer les mises à jour de sécurité Dependabot sur votre dépôt GitHub

Gérez vos dépendances avec GitHub Dependabot.

- Gérer vos dépendances sur GitHub
- Alertes Dependabot
- Mises à jour de sécurité Dependabot
- Gérer les notifications et les rapports Dependabot
- Révision des dépendances
- Exercice : Configurer les mises à jour de sécurité Dependabot

Module 3 : Configurer et utiliser l'analyse des secrets dans votre dépôt GitHub

Découvrez comment fonctionne l'analyse des secrets pour la configurer et l'utiliser efficacement.

- Qu'est-ce que l'analyse des secrets ?
- Configurer l'analyse des secrets
- Utiliser l'analyse des secrets
- Exercise

Module 4 : Configurer l'analyse du code sur GitHub

Ce module présente l'analyse du code et ses fonctionnalités. Vous allez apprendre à implémenter l'analyse du code à l'aide de CodeQL, d'outils tiers et de GitHub Actions.

- Qu'est-ce que l'analyse du code ?
- Activer l'analyse du code avec des outils tiers
- Configurer l'analyse du code
- Configurer l'exercice d'analyse du code

Module 5 : Identifier les vulnérabilités de sécurité dans votre base de code en utilisant CodeQL

Dans ce module, vous apprendrez à connaître CodeQL et comment vous pouvez l'utiliser pour analyser le code de votre référentiel GitHub et d'identifier les vulnérabilités de sécurité.

- Préparer une base de données pour CodeQL
- Exécuter CodeQL dans une base de données
- Comprendre les résultats de CodeQL
- Résoudre les problèmes des résultats CodeQL

Module 6 : Analyse du code avec GitHub CodeQL

Apprenez à utiliser CodeQL, un puissant outil d'analyse statique, pour implémenter l'analyse du code sur GitHub.

- Qu'est-ce que CodeQL ?
- Comment CodeQL analyse-t-il le code ?
- Qu'est-ce que QL ?
- Analyse du code et CodeQL
- Personnaliser votre workflow d'analyse du code avec CodeQL : Partie 1
- Exercice : Référencer une requête CodeQL
- Personnaliser votre workflow d'analyse du code avec CodeQL - Partie 2
- Utiliser l'interface CLI de CodeQL
- Personnaliser les langages et les builds pour l'analyse du code
- Exercice : Configurer une matrice de langage CodeQL

Module 7 : Administration de GitHub pour la sécurité avancée de GitHub

Comprendre la place de GitHub Advanced Security dans votre cycle de développement logiciel et comment l'activer et le déployer dans votre organisation.

- Qu'est-ce que GitHub Advanced Security ?
- Activer GitHub Advanced Security
- Gérer l'accès à GitHub Advanced Security
- Gérer les alertes et les fonctionnalités de GitHub Advanced Security

Module 8 : Gérer les données sensibles et les stratégies de sécurité dans GitHub

Familiarisez-vous avec les outils de sécurité de base de GitHub, qui préparent les dépôts au développement sécurisé et à une réponse aux menaces alignée sur les standards du secteur.

- Définition de stratégies de sécurité
- Créer et gérer des ensembles de règles de référentiel
- Génération de rapports et journalisation
- Exercice