



Le guide complet de
**la formation
au codage
sécurisé pour les
développeurs**



+ Table des matières

04 Comprendre le point de vue
des développeurs

05 Comprendre la Formation au codage
sécurisé pour les développeurs

06 Passer par le jeu

11 Apprentissage contextuel

12 Récapitulatif

+ Préface

À la suite de la publication d'un document du département de la Sécurité intérieure des États-Unis, la sécurité des logiciels est devenue une question d'ingénierie qui doit être prise en compte tout au long du cycle de développement du logiciel.

Il est généralement accepté que les ingénieurs sont le point de départ de la sécurité des logiciels. Cependant, d'après de récentes recherches menées par Node.js® et Squeenix, 60 % des ingénieurs informaticiens n'ont pas confiance dans la sécurité de leurs propres applications. Ceci correspond aux résultats du Rapport de sécurité State of Application de 2016 de Sanz qui révèle que le manque de compétences, d'outils et de méthodes en matière de sécurité logicielle (AppSec) est l'un des trois obstacles les plus importants dans la mise en place de l'AppSec.

D'où vient le manque de compétences relatives à l'AppSec ?

D'une part, une étude de cloudpassage a montré qu'un seul des 36 cursus informatique américains exige que les étudiants suivent un module de sécurité pour valider leur diplôme.

D'autre part, selon StackOverflow, 69,1 % des ingénieurs informaticiens sont autodidactes. La conclusion est claire : les entreprises qui souhaitent assurer que leurs ingénieurs créent du code sécurisé doivent leur dispenser une formation au codage sécurisé de pointe.

Il est intéressant de noter que les organisations comprennent ce besoin. D'après SANS, les entreprises considèrent la formation des développeurs comme la procédure d'AppSec la plus utile, plus encore que la détection des vulnérabilités. Malheureusement, malgré l'adoption généralisée par les entreprises de divers programmes et méthodes de formation des développeurs, le fossé entre les besoins de sécurité du code d'une entreprise et le niveau de formation des ingénieurs informaticiens reste gigantesque.

Ce guide vise à combler ce fossé. Il vous expliquera tout ce que vous devez savoir pour fournir à vos ingénieurs non seulement la meilleure formation au codage sécurisé possible, mais surtout une formation qu'ils vont utiliser.

+ Comprendre le point de vue des développeurs

Dans l'environnement de développement moderne frénétique où des livraisons rapides avec le minimum de bugs sont nécessaires dans un but d'intégration et de livraison continues (CI/CD), le temps des développeurs est la ressource la plus importante. Les solutions de codage sécurisé qui ralentissent les développeurs seront considérées comme une nuisance et seront logiquement mises au placard. C'est la raison pour laquelle les entreprises doivent envisager d'utiliser des solutions d'analyse de code source qui s'intègrent parfaitement dans un environnement DevOps.

Concernant le manque de compétences des développeurs en matière de sécurité, il ne faut pas oublier que leur mission principale est d'écrire du code. Les offres d'emploi, les contrats et les packs d'intégration mentionnent rarement « écriture de code sécurisé » dans les responsabilités des développeurs.

On considère souvent la sécurisation du code comme un petit plus intéressant « si on a le temps », mais comme nous l'avons montré plus tôt, le temps est une ressource rare et précieuse. Les ingénieurs informaticiens sont jugés sur leur vitesse d'exécution et le nombre de bugs existant dans leur code, pas selon le nombre de vulnérabilités.

Malgré tout et sans perdre de vue la position des développeurs, ces derniers doivent quand même créer du code sécurisé. Pour cela, il est indispensable que les chefs d'équipe de développement traitent les vulnérabilités comme les bugs dans le code. Les développeurs savent très bien qu'ils doivent produire un code sans bug et la plupart feront des efforts pour éviter leur apparition. Une fois l'importance de la sécurisation acquise, une bonne formation au codage sécurisé doit être mise en place.

Choisissez une solution
d'analyse du
code sécurisé qui

s'intègre
PARFAITEMENT

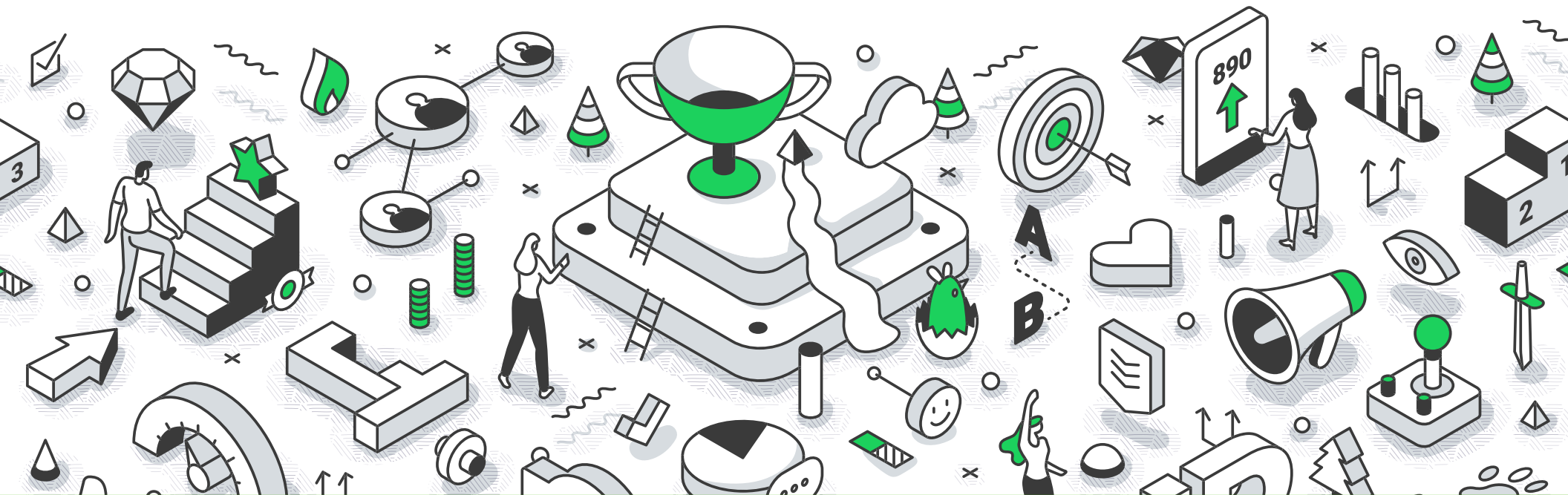
dans un environnement
DEVOPS

+ Comprendre la formation au codage sécurisé pour les développeurs

Les tutoriels vidéo, les formations régulières en classe et les cours en ligne obligatoires sont courants. Pourtant ils n'ont pas les résultats escomptés. Ils sont généralement considérés comme un élément à valider, pas comme un outil de sécurité crucial pour une application. Comme expliqué plus tôt, les développeurs ont d'autres choses à penser, c'est pourquoi une

formation hors contexte et obligatoire ne sera pas considérée comme intéressante.

Avec ces informations, comment les organisations peuvent-elles investir dans des formations au codage sécurisé ? En passant par le jeu et l'apprentissage contextuel.



+ Passer par le jeu

La Gamification (aspect ludique) est l'application de principes et d'éléments issus du monde du jeu dans d'autres contextes. Les avantages de la gamification dans les formations en ligne sont connus depuis longtemps, pourtant la plupart des solutions de formation au codage sécurisé ne l'ont pas adoptée. Nous apprenons mieux lorsque nous nous amusons, lorsque nous sommes actifs et que nous ne nous ennuyons pas. Les développeurs passent le plus clair de leur temps devant un écran rempli de lignes de code. Notre expérience auprès de milliers de développeurs nous a montré que les formations qui apportent un peu de piment, qui ne sont pas ennuyeuses et qui n'exigent pas une grande concentration sont très prisées.

Dans le cadre de la mise en place une formation au codage sécurisé dans votre entreprise, prenez en compte les quatre étapes ci-dessous pour intégrer la gamification :



1 Faites-en une formation interactive



Le Chief Learning Officer ne pourrait pas l'avoir mieux dit : « Ça n'est pas parce qu'un apprenant clique souvent que le contenu est engageant. L'apprenant pourrait juste essayer de finir le cours le plus vite possible. » Nous voyons ce comportement très souvent dans de nombreuses organisations lorsque les apprenants sont confrontés aux formations de sécurité. Les employés doivent suivre un cours en ligne sur un sujet dont ils ne comprennent pas bien l'importance. Ils ont déjà beaucoup à faire, donc ils veulent se débarrasser de la formation obligatoire le plus vite possible pour retourner travailler. La conséquence, c'est que ces cours et ces quiz sont inefficaces, car les employés vont cliquer le plus vite possible pour passer à la suite sans en intégrer les points importants.

« L'interactivité est souhaitable pour plusieurs raisons. Les histoires et exemples contenus dans ces leçons sont un pilier de la formation et permettent de maximiser l'engagement de l'apprenant.

Ces histoires créent une situation où l'apprenant est directement et émotionnellement lié au contenu, ce qui peut améliorer la rétention. » - CLO.

En outre, il est plus difficile de finir une session de formation interactive en cliquant le plus vite possible. Si vous assurez l'interactivité de vos formations, vos développeurs feront plus attention au contenu, ce qui leur donnera de meilleures chances d'apprendre. En outre, beaucoup de gens apprennent mieux en faisant qu'en écoutant ou en regardant. L'interactivité répond à leurs besoins.



2 Racontez une histoire

Des personnages, une notion de jeu de rôle et un fil narratif peuvent faire des merveilles quand il s'agit d'intégrer des informations. Un grand corpus de recherche montre comment les histoires stimulent notre cerveau et déclenchent une réaction. La formation au codage sécurité présentée sous forme de liste à puces, de questions et réponses ou d'un simple texte ne manquera pas d'ennuyer les développeurs. À l'inverse, présenter une formation sous forme narrative, avec des personnages et un problème à résoudre est très efficace. Présentez à vos développeurs des personnages auxquels ils peuvent s'attacher, dans une histoire qu'ils peuvent suivre (dans votre cas, une vulnérabilité qui doit être résolue), cela leur permettra de se souvenir de ce qu'ils ont appris. L'aspect narratif est l'essence de nombreux jeux et est souvent source d'amusement. C'est là qu'entre en jeu la gamification.

3

Restez concis

On entend souvent que notre capacité d'attention diminue, ce qui n'est pas forcément vrai ou prouvé scientifiquement. Néanmoins, il vaut mieux rester concis lorsque vous présentez des informations. Cela vaut pour une session de formation ou pour une présentation PowerPoint. Un contenu plus concis a plus de chance d'aller à l'essentiel, limite les informations superflues potentielles et améliore les chances que les apprenants prêtent attention au contenu. En outre, gardez à l'esprit ce que nous répétons depuis le début de ce guide : le temps est une ressource rare pour les développeurs. La conclusion est sans appel : plus vos sessions seront courtes, mieux ce sera.

4

Assurez-vous qu'ils gagnent

D'après les recherches du Dr Ian Robertson, dont les résultats sont publiés dans le livre The Winner Effect, l'agent d'amélioration neurologique le plus sous-estimé est l'autonomisation. Les recherches du Dr Robertson portent sur la manière dont gagner peut être très bien adapté à l'éducation car le fait de gagner déclenche la libération de beaucoup de bons composés dans notre corps, comme la testostérone, qui améliore ensuite les niveaux de dopamine dans le cerveau, ce qui nous fait nous sentir bien.



THE WINNER EFFECT

l'agent d'amélioration neurologique le plus sous-estimé est l'autonomisation

Pour conclure...

Nous avons vu qu'une courte histoire interactive est importante pour mettre en place un programme de formation au codage sécurisé qui fonctionne. Toutefois, il ne faut pas négliger le fait que les développeurs doivent « gagner » la session de formation. Terminez la session sur une victoire, par exemple la résolution de la vulnérabilité. Faites en sorte que la formation de vos ingénieurs informaticiens soit récompensée, cela les mettra dans de bonnes dispositions et ils se tourneront volontairement vers la solution pour découvrir les vulnérabilités qui étaient passées inaperçues.

+ Apprentissage contextuel

Préparer un programme de formation au codage sécurisé surprenant et ludique est indispensable pour que les développeurs utilisent ce qu'ils auront appris, mais si l'apprentissage n'est pas remis dans son contexte, tous vos efforts peuvent être réduits à néant. Nous avons commencé par expliquer que les formations périodiques en classe ne suffisent pas. Ça n'est pas juste le manque d'aspect ludique qui pose problème, mais aussi le manque de contexte. Lorsque vos développeurs quittent leurs écrans, ils quittent leur environnement habituel, et se souvenir d'une ligne de code problématique devient encore plus difficile.

La formation au codage sécurisé devrait apparaître au moment exact où elle est nécessaire : pendant l'écriture du code. Le secret d'une bonne formation au codage sécurisé est de l'intégrer dans le quotidien des développeurs. Toute novatrice et ludique que soit votre solution de formation au codage sécurisé,

vos développeurs n'ont pas besoin de tout faire d'un coup. Nous avons remarqué que si vos développeurs ont la possibilité d'accéder à vos sessions de formation en continu pendant qu'ils codent, grâce à leur intégration dans leur IDE, ils ont plus tendance à y faire référence lorsqu'ils rencontrent une vulnérabilité.

Notre dernier conseil est de faire en sorte que la formation soit contextuelle et soit à jour au niveau des derniers langages informatiques. Les pratiques de codage sécurisé sont différentes d'un langage à l'autre, il n'y a pas de solution universelle. Découvrez les guides au codage sécurisé de Checkmarx, comme le [Guide du Javascript : pratiques de codage sécurisé pour les applications Web](#) ou le [Guide du Go : pratiques de codage sécurisé pour les applications Web pour en savoir plus](#).

**La formation
au codage sécurisé**
devrait apparaître

MOMENT
EXACT

où elle est nécessaire
**PENDANT L'ÉCRITURE
DU CODE**

```
getById?(d.filter.ID=function(a){var b=a.replace(_,"");return mentsById(a)}),n.g
find.ID=function(a,b){if("undefined"!=typeof b.getElementById){var c=b.getElementById(a);return c?c.value:b}}:(d.filter.ID=function(a){var b=a.replace(_,"");return mentsById(a)}),n.g
butNode&&a.getAttributeNode("id");return c&&c.value===b}},d,function(a){var c=
getElementById&&p){var c,d,e,f=b.getElementById(a);if(f){if(f=find.ID function(a,b){
```

+ Récapitulatif

Nous espérons que vous avez désormais une idée claire de ce à quoi la formation au codage sécurisé doit ressembler afin d'assurer son adoption et l'amélioration des compétences des développeurs. D'après les nombreuses études présentées dans ce guide ainsi que selon la longue expérience de Checkmarx auprès des développeurs, la contextualité et l'aspect ludique de la formation sont cruciaux.

Si vous souhaitez voir toutes les meilleures pratiques présentées dans ce guide traduites concrètement, essayez les leçons en ligne gratuites de Codebashing pour la formation des développeurs au codage sécurisé. Elles allient l'aspect ludique et les conseils présentés en contexte de ce guide et vous assurent que vos développeurs vont apprécier leurs sessions, s'en souvenir et les utiliser dans leur travail.

Leçons en ligne gratuites

de formation au codage sécurisé
de Codebashing



À propos de Checkmarx

Checkmarx est une entreprise de sécurité applicative dont la mission est de fournir aux entreprises des produits et des services qui permettent aux développeurs de livrer des applications sécurisées plus rapidement. La combinaison de la technologie, de la méthodologie et de l'expertise de Checkmarx est la meilleure méthode pour obtenir des résultats haute fidélité, favoriser l'adoption parmi les développeurs, faciliter la remédiation et développer la sécurité des applications. Cinq des 10 plus grands fournisseurs de logiciels, de nombreux organismes gouvernementaux et des entreprises Fortune 500, dont SAP, Samsung et Salesforce.com font partie des plus de 1 400 clients qui ont fait appel à Checkmarx.

Pour en savoir plus sur Checkmarx, consultez www.checkmarx.com