Meilleures pratiques de sécurité des applications Web en 2021

Voici quelques bonnes pratiques que vous pouvez utiliser pour améliorer la sécurité de vos applications Web.

AUTHENTIFICATION ET CONTRÔLE D'ACCÈS

Bien que cela puisse sembler évident, de nombreuses applications Web n'implémente pas de mesures de contrôle d'accès de base. Assurez-vous de suivre ces principes :

- Appliquez des mots de passe forts : utilisez la récupération sécurisée des mots de passe, définissez des politiques d'expiration et de rotation des mots de passe raisonnables et, de préférence, utilisez l'authentification multifacteur.
- Forcez la réauthentification lors de l'accès à des fonctionnalités sensibles ou lors de l'exécution de transactions.
- Utilisez le principe du moindre privilège (POLP) et accordez à chaque utilisateur uniquement les privilèges dont il a besoin pour remplir son rôle dans le système.
- Utilisez SSL et le cryptage et assurez-vous que les mots de passe et les informations d'identification sont toujours cryptés, à la fois au repos et en transit.
- Surveillez les comptes d'utilisateurs et verrouillez les utilisateurs ou demandez un changement de mot de passe si vous détectez une activité suspecte.

Conseils d'experts

Pour les entreprises, qui doivent en outre assurer la sécurité des données et une stricte conformité légale avec des lois telles que GDPR* ou HIPAA**, une solution avancée est nécessaire. En savoir plus sur le contrôle d'accès basé sur les attributs, qui permet un accès dynamique et spécifique au contexte aux

ressources qui peuvent être adaptées à différentes politiques de contrôle d'accès.

*The General Data Protection Regulation 2016/679 is a regulation in EU law on data protection and privacy in the European Union and the European Economic Area.

**The Health Insurance Portability and Accountability Act of 1996 is a United States federal statute enacted by the 104th United States Congress and signed into law by President Bill Clinton on August 21, 1996.

ÉVITER LES ERREURS DE CONFIGURATION DE SÉCURITÉ

Quel que soit le CMS ou le framework de développement Web que vous utilisez, il existe de nombreuses possibilités de mauvaise configuration. Surveillez les problèmes suivants :

- Définissez toujours des mots de passe administrateur forts et modifiez les noms d'utilisateur par défaut
- Protégez les fichiers et les répertoires avec une configuration ou un contenu sensible
- Ne laissez pas les ports ouverts si vous n'avez pas besoin qu'ils soient ouverts inutilement
- Mettez à jour régulièrement vers la dernière version stable pour toutes les bibliothèques logicielles, les plugins et le framework lui-même
- Analysez régulièrement tous vos packages à la recherche de failles de sécurité
- Restez à l'écoute des vulnérabilités de sécurité et des mises à jour affectant vos logiciels et votre infrastructure
- Utiliser des protocoles de communication et de mise en réseau sécurisés
- Assurez-vous que les certificats numériques sont à jour

Conseils d'experts

Nous utilisons le vérificateur de sécurité PHP local après chaque mise à jour du composeur PHP. Cela nous aide à éviter d'utiliser des bibliothèques présentant des vulnérabilités connues. Généralement, des mises à jour régulières vers la dernière version stable résolvent de nombreux problèmes de sécurité.

L'utilisation de bibliothèques et de composants open source dans le développement de logiciels est presque omniprésente, avec environ 99% des

applications ayant au moins un composant open source, selon le rapport 2020 sur la sécurité et l'analyse des risques Open Source.

Certaines industries, telles que la vente au détail, la santé et l'éducation ont connu une croissance exponentielle de leurs revenus au cours de l'année 2020, en grande partie en raison des changements de comportement des consommateurs et des interactions sociales pendant COVID. Comme ces industries utilisaient davantage l'open source dans leurs applications, elles présentaient le plus grand nombre de vulnérabilités et de vulnérabilités à haut risque. Déterminer quels composants open source sont sécurisés doit être une préoccupation majeure pour tout groupe de sécurité d'application.

GESTION DES EXCEPTIONS

Les exceptions sont un aspect souvent négligé de la sécurité des applications Web. Il est courant de voir des exceptions ou des erreurs affichant de longues traces de pile pour l'utilisateur. Ces informations sont extrêmement précieuses pour les attaquants. Vous ne devez jamais afficher quoi que ce soit à un utilisateur autre qu'un message d'erreur qui explique ce qui s'est mal passé et ce qu'il peut faire pour le résoudre.

Assurez-vous de planifier votre application Web pour les scénarios « heureux » et « mécontents » de chaque opération utilisateur. Anticipez toutes les erreurs possibles et gérez-les avec des exceptions significatives. Cela empêchera les attaquants d'exploiter les cas extrêmes pour provoquer un comportement inattendu.

GÉRER LES CONTENEURS AVEC SOIN

Suivant les dernières tendances, de nombreuses applications Web s'exécute sur des conteneurs, à l'aide de Docker . Les conteneurs peuvent créer des problèmes de sécurité majeurs s'ils ne sont pas gérés correctement. Prenez note des consignes de sécurité suivantes :

- Images de confiance : lors de la création d'un conteneur, utilisez toujours des images de base de confiance et analysez les images à la recherche de vulnérabilités (même vos propres images) avant de les utiliser.

- Utiliser des secrets : c'est une très mauvaise pratique d'enregistrer les informations d'identification ou d'autres informations sensibles directement dans une image de conteneur, car elles seront librement disponibles sur tout conteneur créé à partir de cette image. À la place, utilisez le mécanisme des secrets dans Docker ou Kubernetes pour stocker des informations sensibles.
- N'accordez jamais l'accès root : un conteneur qui a un accès root sur un système accordera également l'accès root à un attaquant, dans le cas où le conteneur est compromis. Définissez toujours un utilisateur dans vos images et évitez d'utiliser des options telles que « conteneur privilégié » dans Kubernetes.
- Segmentation du réseau : assurez-vous que les conteneurs ne peuvent accéder à d'autres systèmes que s'ils en ont vraiment besoin. Exécutez les conteneurs dans un sous-réseau protégé et évitez de les exposer à Internet, sauf en cas d'absolue nécessité.

Conseils d'experts

Le service Amazon Elastic Container Registry (ECR) fournit des analyses de sécurité pour les images Docker. C'est une fonctionnalité très utile pour comprendre les problèmes de sécurité potentiels que vous rencontrez dans vos images Docker privées.

ASSURANCE QUALITÉ ET TESTS

Les tests de sécurité sont essentiels pour la sécurité des applications Web.

Suivez ces bonnes pratiques :

- Utilisez l'analyse statique et dynamique : analysez votre code source à la recherche de vulnérabilités pendant le développement à l'aide de Static Application Security Testing (SAST) et en production à l'aide de Dynamic Application Security Testing (DAST).
- Utilisez des tests d'intrusion : vous pouvez utiliser des solutions de test d'intrusion léger en tant que service (PTaaS) ou pour des applications à grande échelle, ainsi qu'un test d'intrusion périodique à grande échelle effectué par un pirate informatique certifié.
- Adoptez CI/CD—chaque fois que vous mettez à jour votre application, exécutez votre code via un processus de test automatisé et déployez-le

automatiquement, pour vous assurer de ne pas introduire de risques de sécurité dus à des problèmes d'installation.

- Tenez compte de la conformité : la plupart des organisations dans le monde sont soumises à la réglementation GDPR de l'Union européenne. Les applications qui traitent les données des cartes de crédit sont soumises à la norme PCI/DSS. Vérifie-s'il existe d'autres normes ou réglementations de conformité qui affectent votre application et mettez en œuvre les mesures nécessaires.

Dans la plupart des cas, notre processus de déploiement habituel en CI/CD comprend les 5 étapes suivantes :

1. Analyser le code source pour les violations de style de code
2. Analyser les bibliothèques tierces utilisées pour les problèmes de sécurité
3. Exécuter le test unitaire
4. Préparez le nouveau code source pour le déploiement.
5. Créez et déployez de nouvelles images Docker dans un environnement sélectionné. Toutes les informations d'identification sensibles requises pour exécuter votre application Web sont stockées dans un stockage sécurisé du CI/CD.

Le déploiement sera arrêté si l'une de ces étapes échoue. Cela garantit des mises à jour stables et un environnement stable et sûr.

Références :

https://mobidev.biz/blog/