OWASP

Table des matières

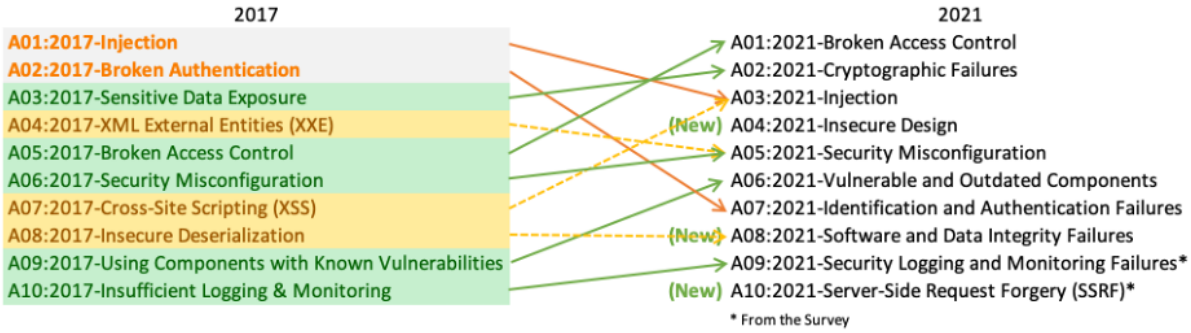
Présentation

Open Worldwide Application Security Project (OWSAP) est une association à but non lucratif dont l'objectif est de renforcer la sécurité de logiciels déjà existants. Elle a été fondée le 1er décembre 2001. Ses principales missions sont d'aider à la réalisation de projets ayant une grande ampleur, de créer et rendre plus actives les communautés en organisant des événements à l'échelle mondiale ainsi que de proposer des ressources à but pédagogique.

Principaux projets

Top 10 OWASP

La sortie de Top 10 OWSAP est prévue dans le 1er semestre de 2025. Il concerne des standards de sécurisation pour des applications web. Il souligne notamment les principales failles de sécurités .



Comparaison des principales failles de sécurité entre 2017 et 2021

A01:2021-Broken Access Control est une catégorie de failles impactant le contrôle d'accès des applications.

A02:2021-Cryptographic Failures concerne les failles exposant des données sensibles ou exposant le système sur lequel l'installation est située.

A03:2021-Injection est une catégorie liée à tous les types d'injections possibles (XSS, SQL etc.) .

A04:2021-Insecure Design concerne les défauts de conception des sites.

A05:2021-Security Misconfiguration désigne les erreurs présentes dans la configuration des sites.

A06:2021-Vulnerable and Outdated Components pointe l'utilisation de dépendances présentant des failles de sécurités ou obsolètes.

A07:2021-Identification and Authentication Failures concerne les problèmes liés à l'authentification des utilisateurs.

A08:2021-Software and Data Integrity Failures désigne les violations d'intégrité des données.

A09:2021-Security Logging and Monitoring Failures concerne les problèmes de collecte de données d'analyse.

A10:2021-Server-Side Request Forgery est une faille permettant aux hackers d'exécuter des requêtes préfabriquées même en utilisant un pare-feu ou un VPN.

L’objectif de ce projet est de collecter des jeux de données liés à des applications à jour pour identifier des vulnérabilités.

ASVS

ASVS est un projet dont l’objectif est de normaliser la portée et la rigueur des vérifications de sécurité présentes sur les sites web à l'aide d'une norme ouverte commercialement exploitable. La norme fournit une base pour tester les contrôles de sécurité technique des applications, ainsi que tous les contrôles de sécurité technique dans l'environnement de l’installation, permettant de se protéger des injections SQL et XSS.

Cheat Sheets OWASP

La série de fiches pratiques OWASP (OWASP Cheat Sheet Series) constitue une ressource essentielle pour les développeurs et les professionnels de la sécurité souhaitant intégrer des pratiques de sécurité efficaces dans le développement d'applications. Ces fiches offrent des conseils concis et concrets sur une multitude de sujets liés à la sécurité des applications, allant de la prévention des vulnérabilités courantes comme l'injection SQL ou le Cross-Site Scripting (XSS), à des domaines plus spécifiques tels que la gestion des secrets, la désérialisation sécurisée ou la configuration du Transport Layer Security (TLS) .

Conçues comme des guides pratiques, ces fiches sont particulièrement utiles pour les développeurs et les ingénieurs DevSecOps. Par exemple, la fiche sur la gestion des secrets fournit des recommandations pour centraliser et sécuriser le stockage des informations sensibles, tandis que celle sur la désérialisation met en garde contre l'utilisation de fonctions non sécurisées comme unserialize() en PHP, suggérant l'adoption de formats plus sûrs comme JSON .

L'un des atouts majeurs de cette série est son alignement avec d'autres projets phares de l'OWASP, tels que l'ASVS (Application Security Verification Standard) et les Proactive Controls, facilitant ainsi l'intégration des bonnes pratiques de sécurité dans les cycles de développement et d'audit . Avec plus de 90 fiches disponibles, régulièrement mises à jour par une communauté active d'experts, la OWASP Cheat Sheet Series s'impose comme une référence incontournable pour renforcer la sécurité des applications de manière pragmatique et accessible.

Impact de OWASP sur la sécurisation web

Utilisation en entreprise

Les entreprises utilisent les référentiels OWASP comme des guides de référence pour renforcer la sécurité de leurs applications tout au long du cycle de développement logiciel. Par exemple, le OWASP Top 10 sert souvent de point de départ pour identifier et corriger les vulnérabilités les plus critiques, en particulier lors des audits de sécurité ou des revues de code. Le OWASP ASVS (Application Security Verification Standard) est quant à lui utilisé pour définir des exigences de sécurité précises selon le niveau de sensibilité de l'application, ce qui permet aux équipes de développement, de test et de conformité de travailler avec un cadre commun. D'autres référentiels comme le OWASP SAMM (Software Assurance Maturity Model) permettent aux entreprises d'évaluer la maturité de leurs pratiques de sécurité et de planifier des améliorations à long terme. En intégrant ces standards dans leurs politiques internes, leurs formations et leurs outils DevSecOps, les entreprises gagnent en efficacité, réduisent les risques de sécurité et répondent plus facilement aux exigences réglementaires et contractuelles.

Exemples d'audits et de bonnes pratiques

Référentiel OWASP	**Exemples d'Audits**	**Bonnes Pratiques Associées**
OWASP Top 10	Vérification de la présence de failles comme XSS, injections, CSRF	Valider les entrées utilisateur, utiliser des requêtes préparées, activer les CSP
OWASP ASVS	Évaluation de conformité aux exigences de sécurité (authentification, logique, etc.)	Implémenter une gestion robuste des sessions, limiter les tentatives de connexion
OWASP SAMM	Analyse de la maturité des pratiques de sécurité dans le SDLC	Intégrer la sécurité dès la phase de conception, former les développeurs à la sécurité
OWASP Cheat Sheet Series	Revue des configurations (ex. HTTP headers, désérialisation, gestion des secrets)	Utiliser Content-Security-Policy, stocker les secrets dans un coffre-fort sécurisé
OWASP Testing Guide	Tests dynamiques et manuels pour identifier des vulnérabilités techniques	Utiliser des outils de test DAST/SAST, automatiser les scans de sécurité

OWASP Proactive Controls	Revue de code axée sur la prévention des erreurs de sécurité	Adopter le principe du "least privilege", sécuriser les APIs, appliquer le chiffrement fort

Exemple d’analyse d’une installation de WordPress

OWASP Top 10 (2021)	Risque identifié sur WordPress	Recommandations de sécurité
A01 – Broken Access Control	Accès non restreint à des pages d'administration via /wp-admin	Limiter l'accès par IP, utiliser un plugin de restriction d'accès, désactiver l’indexation
A02 – Cryptographic Failures	Connexions HTTP encore possibles	Forcer HTTPS avec redirection 301, activer HSTS via .htaccess
A03 – Injection (SQL, XSS, etc.)	Plugins vulnérables à des injections XSS ou SQL	Mettre à jour les plugins/thèmes, utiliser des extensions réputées et vérifiées
A04 – Insecure Design	Pages critiques accessibles sans CAPTCHA ou logique anti-abus	Implémenter des protections (reCaptcha, jetons CSRF)
A05 – Security Misconfiguration	Affichage des erreurs PHP ou page de login WordPress exposée	Désactiver le debug en prod (WP_DEBUG = false), masquer la version WP
A06 – Vulnerable and Outdated Components	Thèmes ou plugins non mis à jour depuis longtemps	Mettre à jour WP, ses plugins et thèmes régulièrement. Supprimer ceux inutilisés
A07 – Identification and Authentication Failures	Force brute possible sur /wp-login.php	Activer la limitation de tentatives de connexion, utiliser 2FA
A08 – Software and Data Integrity Failures	Installation de plugins via ZIP non vérifiés	Installer uniquement depuis le dépôt officiel WordPress
A09 – Security Logging and Monitoring Failures	Aucune alerte sur les connexions suspectes	Installer un plugin de sécurité comme Wordfence ou Sucuri
A10 – Server-Side Request Forgery (SSRF)	Fonctions PHP comme wp_remote_get() mal sécurisées	Filtrer les requêtes sortantes, désactiver les accès externes non nécessaires

Sources

https://owasp.org/about/

https://owasp.org/www-project-top-ten/

https://owasp.org/Top10/A10_2021-Server-Side_Request_Forgery_%28SSRF%29/

https://github.com/OWASP/CheatSheetSeries

https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

https://cheatsheetseries.owasp.org/cheatsheets/Deserialization_Cheat_Sheet.html

https://fr.wikipedia.org/wiki/Open_Worldwide_Application_Security_Project

https://devguide.owasp.org/05-implementation/01-documentation/03-cheatsheets/