SE DÉFENDRE CONTRE LE TOP 10 OWASP DES RISQUES LIÉS AUX API

Les produits UBIKA vous protègent grâce à des mécanismes avancés de protection API.

A1 AUTORISATION CASSÉE AU NIVEAU DES OBJETS

- Les API exposent des points de terminaison qui manipulent des identifiants d'objets, ce qui crée une vaste surface d'attaque - Level Access Control Issue.
- Les contrôles d'autorisation au niveau de l'objet font défaut dans chaque fonction qui accède à une source de données en utilisant une entrée del'utilisateur.

Produits UBIKA

Rendent aléatoire les identifiants d'application (ids) par session avec des jetons aléatoires pour augmenter la difficulté à deviner les ids non autorisés.

A2 AUTHENTIFICATION CASSÉE AU NIVEAU DES UTILISATEURS

- Les mécanismes d'authentification sont mis en oeuvre de manière incorrecte, ce qui permet aux attaquants de compromettre les jetons d'authentification pour prendre l'identité d'autres utilisateurs.
- Cela compromet la sécurité de l'API dans son ensemble.

Produits UBIKA

Attribuent un jeton Web JSON (JWT) à l'authentification de l'utilisateur. Appliquent une limitation de débit sur les terminaux d'authentification.

A3 EXPOSITION EXCESSIVE DE DONNÉES

- Les développeurs ont tendance à exposer toutes les propriétés desobjets sans tenir compte de leur sensibilité individuelle.
- Ils comptent sur les clients pour effectuer le filtrage des donnéesavant de les afficher à l'utilisateur.

Produits UBIKA

Détectent les messages d'erreur sensibles et les données confidentielles grâce à des capacités de filtrage sortant avec validation de schéma sur toutes les réponses API.

A4 MANQUE DE RESSOURCES & LIMITATION DU TAUX

- Les APL n'imposent aucune restriction sur la taille ou le nombre de ressources qui peuvent être demandées par le client/utilisateur.
- Cela peut affecter les performances du serveur API, entraînant un déni de service (DoS), et des failles d'authentification telles que des attaques parbrute force.

Produits UBIKA

Comprennent une limitation du débit des demandes d'API, un analyseur d'API et une validation des schémas pour mettre fin aux attaques par brute force.

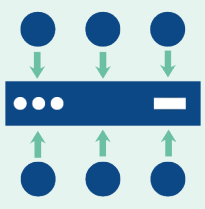
A5 AUTORISATION CASSÉE AU NIVEAU DES FONCTIONS

- Des politiques de contrôle d'accès complexes avec des hiérarchies, des groupes et des rôles différents, ainsi qu'une séparation peu claire entre les fonctions administratives et les fonctions normales, entraînent desdéfauts d'autorisation.
- En exploitant ces problèmes, les attaquants peuvent accéder aux ressources et/ou aux fonctions administratives d'autres utilisateurs.

Produits UBIKA

Impliquent la validation du schéma de l'API comme OpenAPI3 pour bloquer les actions non autorisées ou inconnues définies au moment de la conception.

A6 AFFECTATION DES MASSES

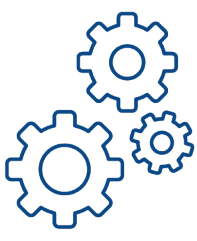


- Lier des données fournies par le client (par exemple JSON) à des modèles de données, sans filtrage approprié des propriétés sur la base d'une liste d'autorisations, conduit généralement à une affectation en masse.
- Les attaquants peuvent modifier des propriétés d'objet qu'ilsne sont pas censés modifier en devinant les propriétés des objets, en explorant d'autres points d'accès à l'API, en lisant la documentation, ou en fournissant des propriétés d'objet supplémentaires dans les charges utiles de la requête.

Produits UBIKA

Incluent la validation du schéma de l'API comme OpenAPI3 pour bloquer les actions non autorisées définies au moment de la conception.

A7 MAUVAISE CONFIGURATION DE LA SÉCURITÉ

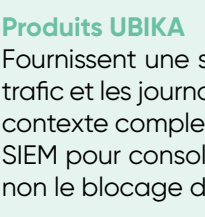


- La mauvaise configuration de la sécurité est due à des configurations par défaut non sécurisées, à des configurations incomplètes ou ad hoc, à un stockage open cloud, à des en-têtes HTTP mal configurés, à des méthodes HTTP inutiles, à un partage de ressources inter-origine (CORS) permissif et à des messages d'erreur verbeux contenant des informations sensibles.

Produits UBIKA

Ajoutent des en-têtes de sécurité aux clients. Détectent les messages d'erreur avec le filtrage sortant. Permettent des capacités de patching virtuel sur le contrôle d'accès en utilisant Worktlow-as-Code.

A8 INJECTION



- Les failles d'injection, telles que SOL, NoSQL, l'injection de commande, etc., se produisent lorsque des données non fiables sont transmises àun interprète dans le cadre d'une commande ou d'une requête.
- Les données malveillantes de l'attaquant peuvent inciterl'interprète à exécuter des commandes involontaires ou à accéder à des données sans autorisation.

Produits UBIKA

Détectent les injections avec des moteurs de listes noires (ICX) et bloque les ressources/données inconnues avec la validation du schéma API.

A9 GESTION INAPPROPRIÉE DES ACTIFS

- Les API tendent à exposer plus de points de terminaison que les applications web traditionnelles. Il est d'autant plus important d'avoir une documentation appropriée et mise à jour.
- Un inventaire approprié des hôtes et des versions d'API déployées joue également un rôle important pour atténuer les problèmes tels que les versionsd'API dépréciées et les points de fin de débogage exposés.

Produits UBIKA

Incluent la validation du schéma de l'API comme OpenAPI3 pour bloquer les actions non autorisées définies au moment de la conception. Permettent d'établir une liste blanche ou noire des points de terminaison d'API connus, inconnus ou obsolètes.

A10 ENREGISTREMENT ET SURVEILLANCE INSUFFISANTS

- L'absence ou l'inefficacité des moyens de surveillance et d'enregistrement des événements entraîne une sensibilité du système.
- La plupart des études sur les intrusions montrent que le temps moyen de détection d'une attaque est supérieur à 200 jours.
- Une attaque est détectée par des prestataires de services externes plutôt que par la surveillance et les processus internes.

Produits UBIKA

Fournissent une surveillance détaillée via les journaux d'accès sur le trafic et les journaux de sécurité sur les événements détectés (avec le contexte complet de la demande). Envoient les journaux au système SIEM pour consolidation. Permettent uniquement la journalisation et non le blocage des demandes lors de la détection.