**Sensitive data exposure**

Définition

Toute application Web est sujette à une manipulation de données. Dans une application Web, on distingue, selon leur état, deux types de données : les données stockées et les données en transit.

Dans les deux cas, des problèmes se présentent lorsque des données sensibles (banque, santé, données personnelles…) ne sont pas suffisamment protégées (transmises ou stockées en texte clair) ou sont mal protégées.

Sources :

Causes

Les données sensibles méritent des précautions particulières lors de leur stockage mais aussi lors de leur transport. Si de telles mesures ne sont pas prises, des individus malintentionnés peuvent voler ou modifier ces données faiblement protégées pour effectuer un vol d'identité, de la fraude à la carte de crédit, entre autres crimes.

L’exposition de données sensibles peut se produire lorsque les contrôles de sécurité nécessaires assurant la confidentialité des données ne sont pas appliqués ou sont appliquées de façon incorrecte, permettant ainsi aux attaquants de voler des données sensibles telles que les mots de passe, les informations de paiement, les adresses ou toute autre donnée présentant une certaine valeur pour l’attaquant. Les applications doivent garantir que l'accès soit authentifié et que les données qui revêtent une certaine importance soient bien protégées en stockage et lors de leur transport. Un échec à assurer cela peut conduire à une violation majeure de la vie privée et à une compromission de l’application.

Vecteurs

* Attaque par l’homme du milieu
* Vol de données stockées en texte clair sur le serveur
* Vol de données stockées sur le « navigateur » du client
* Vol de clés de cryptage

Preventions

* Classifier les données manipulées par l’application, identifier parmi celles-ci lesquelles sont sensibles en se basant sur le contexte de l’application mais aussi sur la loi et appliquer des contrôles par rapport à la classification
* Ne conserver que les données sensibles nécessaires
* Crypter toutes les données sensibles lors de leur stockage : S’assurer que des protocoles de chiffrements forts sont utilisés et que les clés sont bien gérées, Choisir des algorithmes éprouvés et générer des clés robustes, S'assurer qu'une bonne gestion des clés est en place, Privilégier des modules cryptographiques certifiés. Mettre à jour les protocoles de chiffrement selon les avancées de la cryptanalyse
* Crypter toutes les données en transit avec des protocoles sécurisés tels que TLS
* Désactiver la mise en cache pour les réponses contenant des données sensibles et l'attribut "autocomplete" dans les formulaires collectant des données sensibles.
* Stocker les mots de passe par hachage fort avec utilisation d’un sel. Hacher les mots de passe au moyen d’un algorithme adapté à cet usage, tel que bcrypt, PBKDF2, ou scrypt. Mettre à jour les algorithmes utilisés pour le hachage
* Mettre les cookies en http only et secure
* Mettre en place une bonne politique de mots de passe

Supplémentaires

* Vérifier l’efficacité de la configuration et des paramètres
* Ne pas utiliser le chiffrement automatique des données par le SGBD
* Spécifier le header HSTS forçant le client à ne communiquer qu’en https avec le serveur pour toutes les ressources qu’il va demander. Cependant il ne faut pas se fier qu’à cela, il faut plutôt spécifier carrément https en donnant le chemin d’une ressource ou privilégier les chemins relatifs
* CERTIFICATE PINNING

Protocoles de chiffrements recommandés par l’Owasp

* Echange de clés : Diffie–Hellman key exchange with minimum 2048 bits
* Intégrité de message : HMAC-SHA2
* Message Hash: SHA2 256 bits
* Chiffrement asymétrique : RSA 2048 bits
* Chiffrement sym : AES 128 bits
* Password Hashing: PBKDF2, Scrypt, Bcrypt.

while attackers can make any number of mistakes and still succeed, all you have to do is make one security mistake to [pwned](https://en.wikipedia.org/wiki/Pwn).