Le chiffrement source de multiples failles de sécurité

Les préoccupations en matière de sécurité poussent les développeurs à recourir de plus en plus fréquemment au chiffrement. Sauf que ces derniers n'en maîtrise pas toujours l'implémentation, conduisant à introduire de nouvelles failles dans les applications. Telle est en tout cas la conclusion d'une étude de Veracode (un fournisseur de solutions de sécurité), basée sur **l'analyse de vulnérabilités de plus de 200 000 applications** commerciales ou développées à façon et employées en entreprise. Selon celle-ci, les vulnérabilités dans les protocoles de chiffrement sont la seconde cause la plus commune de failles dans les applications, derrière les défauts dans la qualité du code. La plupart des logiciels touchés par ces vulnérabilités sont des applications Web. Mais le phénomène s'étend aussi les applications mobiles.

Vulnerability	Financial Services	Government	Healthcare	Manufacturing	Retail & Hospitality	Technology	Other	Rank
Code Quality	65%	70%	80%	56%	68%	70%	65%	1
Cryptographic Issues	60%	66%	61%	51%	63%	62%	59%	2
Information Leakage	58%	62%	60%	49%	55%	62%	53%	3
CRLF Injection	52%	52%	48%	45%	54%	54%	48%	4
Cross-Site Scripting (XSS)	49%	51%	46%	45%	52%	49%	47%	5
Directory Traversal	48%	48%	45%	40%	44%	48%	46%	6
Insufficient Input Validation	41%	45%	43%	33%	44%	37%	37%	7
SQL Injection	29%	40%	32%	31%	25%	30%	34%	8
Credentials Management	25%	20%	26%	24%	24%	28%	32%	9
Time and State	23%	19%	23%	17%	21%	26%	23%	10

Les failles cryptographiques seraient ainsi plus courantes que d'autres problématiques bien connues, comme le cross-site scripting, l'injection SQL ou le directory traversal (un exploit HTTP

donnant accès à des répertoires ou fichiers en accès restreint).Elles se nichent notamment dans des validations impropres de certificats TLS (Transport Layer Security), le stockage en clair d'informations sensibles, l'utilisation d'une longueur de clef insuffisante, la présence de clefs cryptographiques codées en dur, l'usage de vecteurs d'initialisation non aléatoires ou encore dans la vérification incorrecte des signatures cryptographiques. Liste non exhaustive.

La faute aux développeurs ?

Bref, selon Veracode, les développeurs veulent chiffrer, notamment pour répondre aux législations en matière de protection des données, mais ne savent pas s'y prendre. L'étude jette une lumière crue sur les conséquences de **l'absence de formation appropriée** pour les informaticiens, ce qui aboutit à créer « *un faux sentiment de sécurité* », selon Chris Wysopal, le directeur technique de Veracode, interrogé par nos confrères d'Infoworld. Nombre de développeurs ont tendance à penser qu'il leur suffit d'appeler une librairie de chiffrement, comme OpenSSL, pour sécuriser les données de leurs applications. Mais l'implémentation comporte de nombreux pièges.

Pour Matthew Green, un professeur de cryptographie de l'Université John Hopkins (Baltimore), accuser les seuls développeurs est toutefois un peu trop simple. La faute est, selon lui, au moins partagée avec les concepteurs de **librairies de crypto, pensées pour des spécialistes** du sujet et non pour des développeurs lambda. « *Forcer les développeurs à les utiliser, c'est comme demander à quelqu'un ayant son permis de conduire de faire décoller un avion* », illustre-t-il. Une lacune dont OpenSSL semble avoir pris conscience, son plan d'évolution, mis à jour pour la dernière fois en octobre dernier, prévoyant de réduire la complexité de l'API et d'améliorer la documentation.

A lire aussi :

E. Thomé, Inria : « Les clefs de chiffrement de 768 bits ne suffisent plus »
LogJam : nouvelle faille dans le chiffrement des sites Web
Le chiffrement stratégique progresse dans les entreprises

Crédit photo : Maksim Kabakou / Shutterstock