Le chiffrement, seconde source de failles de sécurité dans le code

Pour l'édition 2016 de son rapport (*State of Software Security*), Veracode a analysé le code source de plus de 337 000 applications 18 mois durant, d'octobre 2014 à mars 2016. Moins de 4 applications sur 10 répondent à des exigences sécurité (référentiel OWASP, Open Web Application Security Project) lors de l'évaluation initiale, selon le fournisseur américain spécialisé.

« Cette proportion ne change pas vraiment année après année. Cela montre que de nombreux logiciels en circulation ne passent toujours pas par un processus formel d'amélioration de sécurité — soit le code hérité n'a pas été corrigé, soit le code nouveau n'a pas été développé dans le cadre d'un cycle de vie du développement logiciel (SDLC) rigoureux et sécurisé », déclarent les auteurs du rapport.

10 sources de vulnérabilités

Les développeurs accélèrent le recours au chiffrement dans leurs applications. Mais la rigueur de l'implémentation n'est pas toujours au rendez-vous. Résultat, la cryptographie reste la deuxième source de failles de sécurité dans le code des applications étudiées par Veracode.

Devant les problèmes de chiffrement, la fuite d'informations (sur la configuration ou le fonctionnement interne des applications, par exemple) arrive en tête des dix sources de vulnérabilités les plus fréquemment trouvées dans les applications scannées par Veracode. Suivent : la qualité du code, la faille CRLF (Carriage Return Line Feed), la faille XSS (ou Cross-site scripting), la vulnérabilité de type « traversée de répertoires », la validation d'entrée insuffisante, la violation de gestion d'authentifications, la faille d'injection SQL et, enfin, les problèmes d'encapsulation.

Lire aussi :

Sécurité : Linux attaquable sans une seule ligne de code
DDoS : le code du botnet IoT Mirai mis en libre-service
Le chiffrement source de multiples failles de sécurité

crédit photo © Morrowind