Informations sur la Sécurité par Contrast Labs

Explication des risques AppSec pour certaines vulnérabilités du Top 10 de l'OWASP

Chez Contrast Security, l'équipe Contrast Labs ne chôme jamais. Elle est notamment chargée de recueillir des informations sur les menaces et de comprendre le paysage dans lequel elles s'inscrivent afin d'analyser les risques que peuvent poser différentes vulnérabilités pour une entreprise.

Dans mon cas, après 20 ans dans le domaine de la sécurité des applications (AppSec), j'ai acquis l'intime conviction « qu'il y a toujours des problèmes », et que les développeurs doivent tout simplement les réparer. Mais, s'il en allait ainsi pour les approches traditionnelles de l'AppSec, cette hypothèse ne se vérifie peut-être plus de façon aussi systématique aujourd'hui.

Des tonnes de code

Les équipes de développement avancent très vite : Les développeurs écrivent en moyenne 325 à 750 lignes de code (LOC) par mois. Et avec plus de 23,9 millions de développeurs dans le monde, imaginez les volumes que cela représente. Certaines grandes entreprises possèdent des milliers d'applications et une dette technique qui ferait pâlir d'inquiétude tout professionnel de la sécurité. Notre cri du cœur, c'est qu'il est « ABSOLUMENT IMPOSSIBLE DE TOUT CORRIGER ». Certains en sont convaincus et s'en accommodent. C'est la dure réalité.

Analyse de la situation sous l'angle du Top 10 (2017) de l'OWASP

Dans ce contexte, comment utiliser à bon escient les données et les informations sur les menaces ? Commençons par examiner le Top 10 (2017) de l'OWASP, une référence dans tous les secteurs et même pour les organes de conformité et de réglementation comme le Payment Card Industry Software Security Framework (PCI SSF).

Le Top 10 de l'OWASP est un bon point de départ pour mettre sur pied un programme AppSec. Cependant, tous les risques énumérés ont-ils vraiment le même profil de risque ? Commençons par décomposer certains d'entre eux en comparant les données publiques sur les vulnérabilités et les expositions courantes (CVE) et l'énumération des failles courantes (CWE). Nous pouvons ensuite attribuer un score de risque sur la base du Système de notation des vulnérabilités courantes (CVSS).

Top 10 de l'OWASP et CWE

Contrast Labs a choisi les catégories ci-dessous parce que nous pouvons les associer à une CWE directe ou à quelques vulnérabilités plus flagrantes. Les autres catégories du Top 10 de l'OWASP sont beaucoup plus larges et correspondent à de nombreux CWE différents.

A1:2017-Injection

Injection de commande (CWE-77)/Injection de commande OS (CWE-78). Les acteurs malveillants utilisent l'injection de commande pour exécuter des commandes arbitraires sur un système d'exploitation hôte sous-jacent. Ces acteurs malveillants peuvent localiser où une application vulnérable peut recevoir une entrée utilisateur et en dériver une commande système. L'entrée peut provenir de n'importe quelle partie d'une requête HTTP, y compris les paramètres, les en-têtes, les cookies, etc. L'objectif ultime est d'exécuter des commandes sur le système d'exploitation sous-jacent, ce qui peut conduire à l'exfiltration de données, à l'élévation de privilèges ou, dans certains cas, à l'installation de logiciels malveillants.

Injection SQL (CWE-89). Une injection SQL se produit lorsqu'un acteur malveillant insère ou injecte du code dans une requête SQL via une entrée, généralement un paramètre ou un en-tête de requête HTTP. Un acteur malveillant procéderait de la sorte pour effectuer des actions CRUD sur une base de données back-end afin de dérober des données sensibles, de modifier des informations, d'agir comme un utilisateur privilégié, ou même de supprimer des données ou des tables. Ce type d'attaque peut totalement compromettre la confidentialité, l'intégrité et la disponibilité des données et l'accès à la base de données.

A4:2017-XXE

Entité externe XML (CWE-611). Les attaques par entité externe XML (XXE) sont utilisées par des acteurs malveillants qui cherchent à tirer profit des applications qui parsent le XML. Ces acteurs malveillants exploitent des parseurs XML qui permettent de référencer des entités externes qui sont ensuite traitées par l'analyseur XML sous-jacent. Les entités externes sont des valeurs définies chargées depuis l'extérieur de la définition de type de document (DTD) dans laquelle elles sont déclarées. Ces entités externes peuvent être dérivées d'un chemin de fichier ou d'une URL. Les acteurs malveillants utilisent ce type d'attaque pour accéder à des données sensibles et réaliser une attaque par déni de service distribué (DDoS). Dans certains cas, ils effectuent des attaques server side request forgery (SSRF).

A5:2017-Broken Access Control

Path Traversal (CWE-22). Le Path traversal, également connu sous le nom Directory traversal attack, est une attaque perpétrée par des acteurs malveillants pour accéder à des fichiers sur le système auxquels ils n'auraient pas nécessairement accès dans le cadre d'une utilisation normale de l'application. Celle-ci consiste à manipuler des parties d'une requête HTTP, qu'il s'agisse de paramètres, d'en-têtes, etc., en ajoutant le fameux point-point-barre oblique (../) à de multiples reprises. Certains, dont l'OWASP, classent le path traversal comme un problème de contrôle d'accès – ce qui est en partie vrai – bien qu'elle puisse aussi être considérée comme un problème d'injection du fait de la manipulation de l'entrée HTTP avec le *point-point-barre oblique*.

A7:2017-XSS

Cross-Site Scripting (CWE-79). Les pirates utilisent le Cross-Site Scripting (XSS) pour injecter des scripts malveillants sur les sites Internet. Ils créent un code malveillant, généralement sous la forme de JavaScript, utilisé pour inciter un autre utilisateur à interagir, généralement par le biais d'un lien ou en stockant le code

sur le site Internet pour qu'il soit consulté par la suite par une victime innocente. Les vulnérabilités qui permettent à ces attaques de réussir sont assez répandues et se produisent partout où une application web génère une entrée utilisateur dans l'output sans validation ou encodage.

A8:2017-Désérialisation non sécurisée

Désérialisation non sécurisée (CWE-502). La désérialisation non sécurisée est une vulnérabilité des applications web qui permet aux utilisateurs de transmettre des objets ou des codes arbitraires à un désérialiseur. Dans ce type d'attaque, les données non fiables abusent de la logique d'une application pour infliger une attaque DDoS, contourner l'authentification, exécuter le code à distance et même exécuter du code arbitraire pendant qu'il est désérialisé.

Examen du risque des CVE

Contrast Labs a choisi les vulnérabilités ci-dessus, car elles peuvent être rapidement associées à des CWE. Nous pouvons également évaluer le risque de chaque CVE connu en fonction de son score CVSS. Pour ce faire, nous avons étudié le pourcentage de chaque CVE associé à un CWE spécifique en fonction de son score CVSS de base :

- 1-3,9 : Risque faible
- 4,0-6,9 : Risque moyen
- 7,0-8,9 : Risque élevé
- 9,0-10,0 : Risque critique

Nous avons examiné chaque CWE et décomposé les pourcentages de CVE en examinant les pourcentages supérieurs ou égaux à neuf, supérieurs ou égaux à huit, etc., jusqu'à quatre. Les pourcentages inférieurs à quatre n'ont pas été retenus, car de nombreuses entreprises accordent moins d'importance à ces faibles risques.

A1:2017-Injection. A1:2017-Injection et les classes de vulnérabilité CWE-77, CWE-78 et CWE-89 sont communément considérées comme flagrantes et dangereuses, et la plupart des entreprises les jugeraient à haut risque. Les chiffres ne mentent pas non plus :

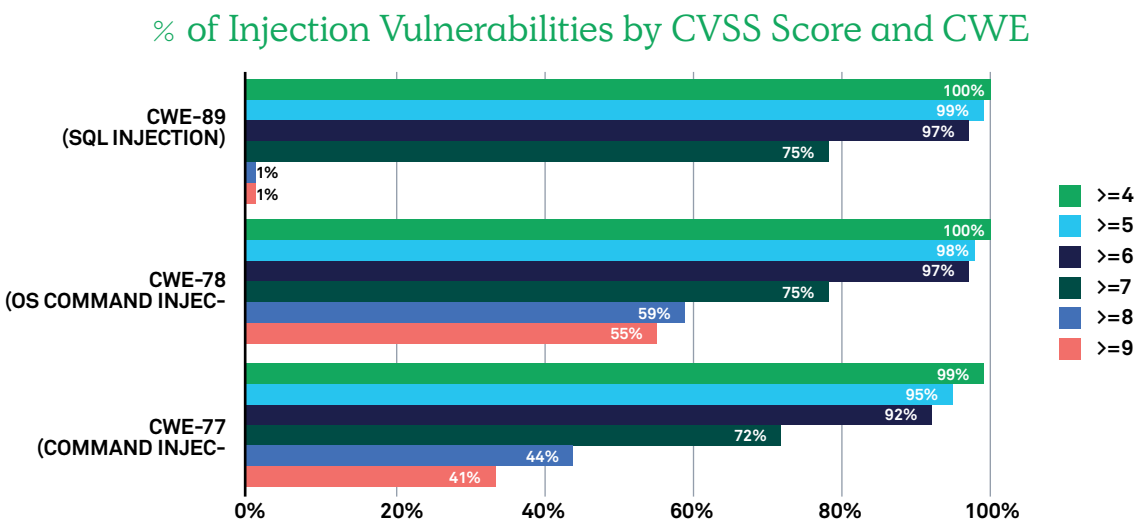


Schéma 1 : Trois vulnérabilités d'injection, dont la majorité avec un score CVSS de 6,0 ou plus.

Comme on pouvait s'y attendre, ces chiffres prouvent que ces types d'injection sont très risqués pour une entreprise. Tous trois sont supérieurs à 90 %, pour un score CVSS de six ou plus – soit un risque moyen élevé, élevé ou critique. Il serait judicieux qu'une entreprise s'intéresse en priorité à ces classes de vulnérabilités lors de l'élaboration d'un programme AppSec pour protéger correctement ses applications.

A4:2017-XXE. Examinons maintenant A4:2017-XXE – autrement dit, CWE-611. La vulnérabilité XXE est encore très répandue parmi les applications web, car les stacks technologiques modernes permettent toujours le traitement des données XML. Les chiffres ne mentent pas et ressemblent beaucoup à une injection A1:2017 :

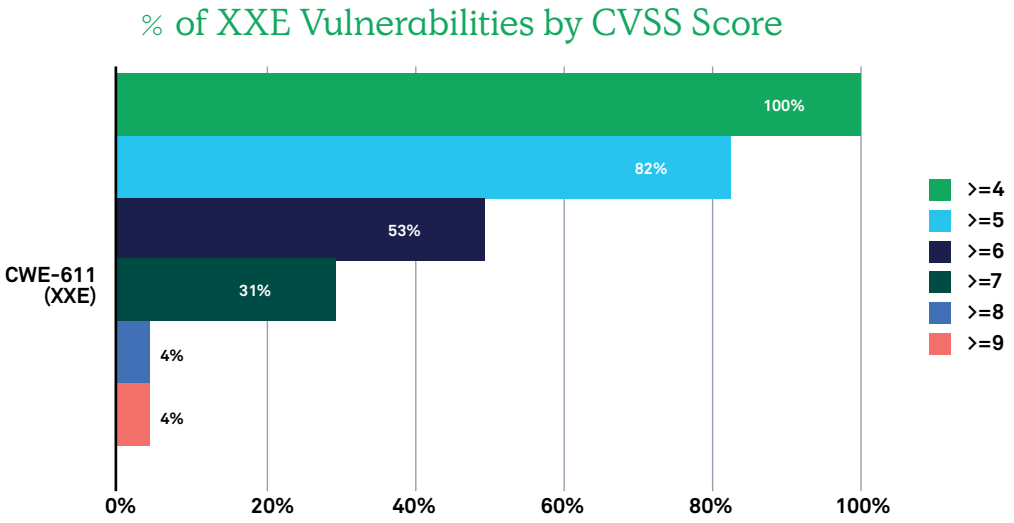


Schéma 2 : vulnérabilités XXE dont la majorité présente un score CVSS de 6,0 ou plus.

A5:2017-Broken Access Control Suivant dans la liste : A5:2017- Broken Access Control – autrement dit CWE-22. Les Path traversal existent depuis toujours, et les serveurs web et d'application possèdent maintenant des protections intégrées contre l'accès à certains fichiers. Cependant, les Path traversal restent un problème, et les chiffres CVSS en témoignent.

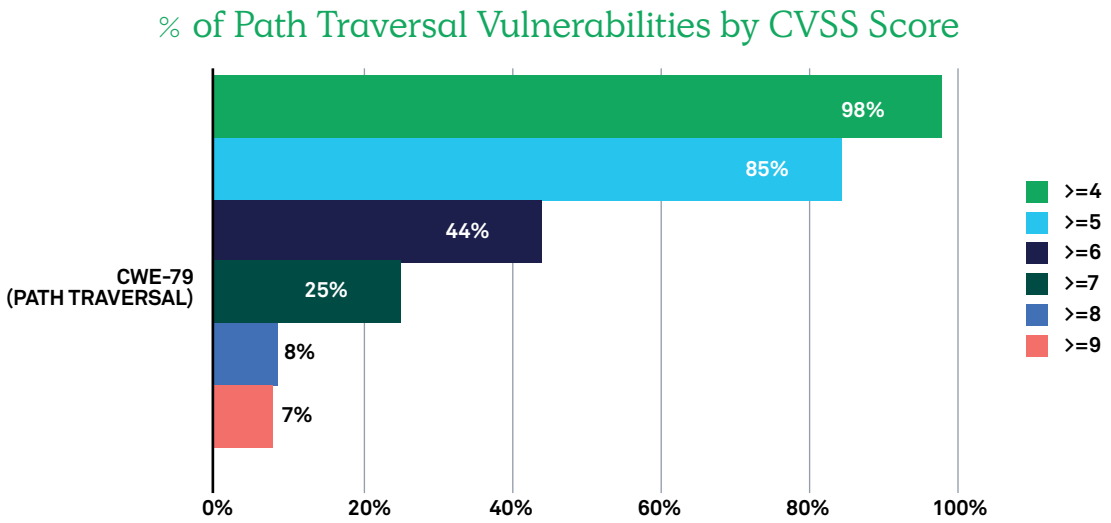


Schéma 3 : vulnérabilités de Path traversal dont la majorité présente un score CVSS de 5,0 ou plus.

À y regarder de plus près, nous nous apercevons que CWE-22 et CWE-611 ne posent pas autant de risques que les vulnérabilités liées aux injections. Tant les Path traversal que les XXE voient la majorité de leurs scores de base CVSS à cinq et plus, contre sept et plus pour les vulnérabilités d'injection. Par conséquent, nous pouvons conclure que la plupart des vulnérabilités XXE et de Path traversal présentent un risque moyen, et que les injections présentent des risques plus élevés ou critiques.

A7:2017-XSS. Les attaques XSS sont répandues dans les applications web et ont toujours été considérées comme très risquées. L'une des attaques XSS les plus connues a visé MySpace en 2005. Surnommé le « ver Samy », Samy Kamkar a infecté MySpace en utilisant un exploit XSS stocké qui s'est propagé d'utilisateur en utilisateur jusqu'à submerger la base de données back-end.

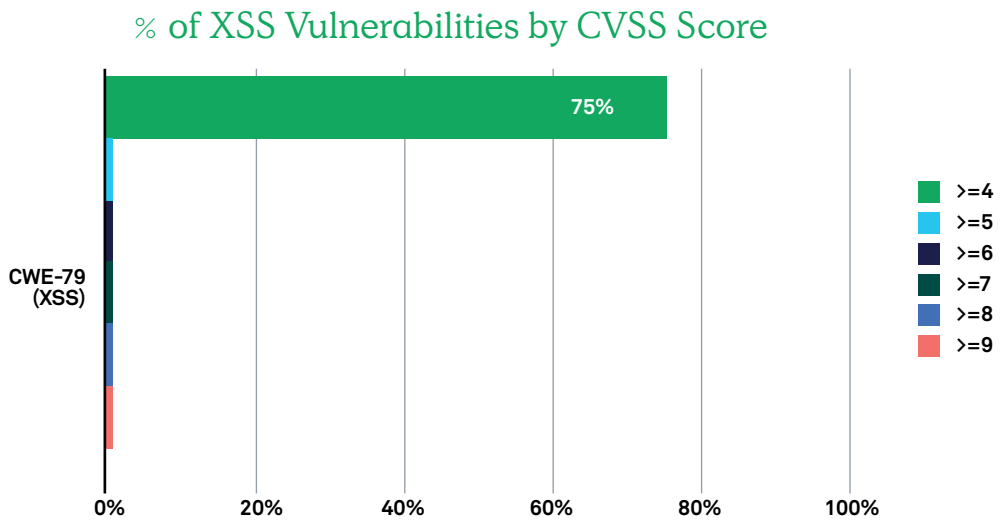


Schéma 4 : vulnérabilités XSS dont la majorité présente un score CVSS de 4,0 ou plus.

Waow ! Ces résultats sont surprenants – les scores de base CVSS de l'écrasante majorité des CVE XSS s'échelonnent entre quatre et cinq, soit la limite basse d'un risque moyen. Une analyse plus poussée révèle d'autres constatations intéressantes : 22 % des CVE XSS ont un score de base CVSS compris entre trois et quatre. Lorsqu'on regroupe ces données, 97 % des CVE XSS présentent un score CVSS de trois à cinq – soit un risque faible à moyen.

Dès lors, pourquoi les exploits XSS sont-ils le centre de toutes les attentions ? Tout d'abord, la grande majorité des CVE proviennent de bibliothèques tierces ou open source qui pourraient inévitablement avoir un impact sur les scores de base CVSS calculés. De plus, la plupart des bibliothèques ne possèdent pas vraiment l'interface web nécessaire à la réussite d'un exploit XSS. Quoi qu'il en soit, les données sont intéressantes et méritent d'être prises en compte.

A8:2017-Désérialisation non sécurisée. Dans la plupart des cas, la désérialisation non sécurisée conduit à une forme d'exécution de code à distance ou d'attaque DDoS –, ce qui devrait la classer dans la catégorie des risques élevés. Il faut savoir que l'attaque bien connue subie par Equifax était due à une désérialisation non sécurisée du framework Java Apache Struts 2 (CVE-201-5638). Mais que nous disent les chiffres CVE et CVSS ?

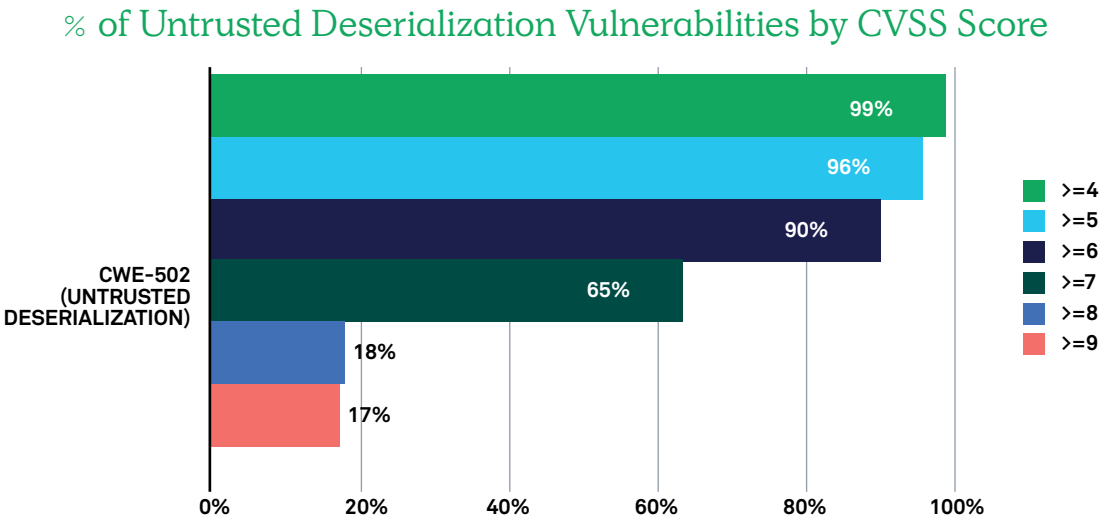


Schéma 5 : vulnérabilités de désérialisation non fiables dont la majorité présente un score CVSS de 7,0 ou plus.

Comme on pouvait s'y attendre, la majorité des CVE de désérialisation non sécurisée ont obtenu un score de base CVSS de sept ou plus –, ce qui indique qu'ils présentent un risque élevé.

Rassembler toutes les pièces du puzzle

Maintenant que nous avons examiné chaque problème séparément, rassemblons-les pour obtenir une vue d'ensemble du Top 10 de l'OWASP.

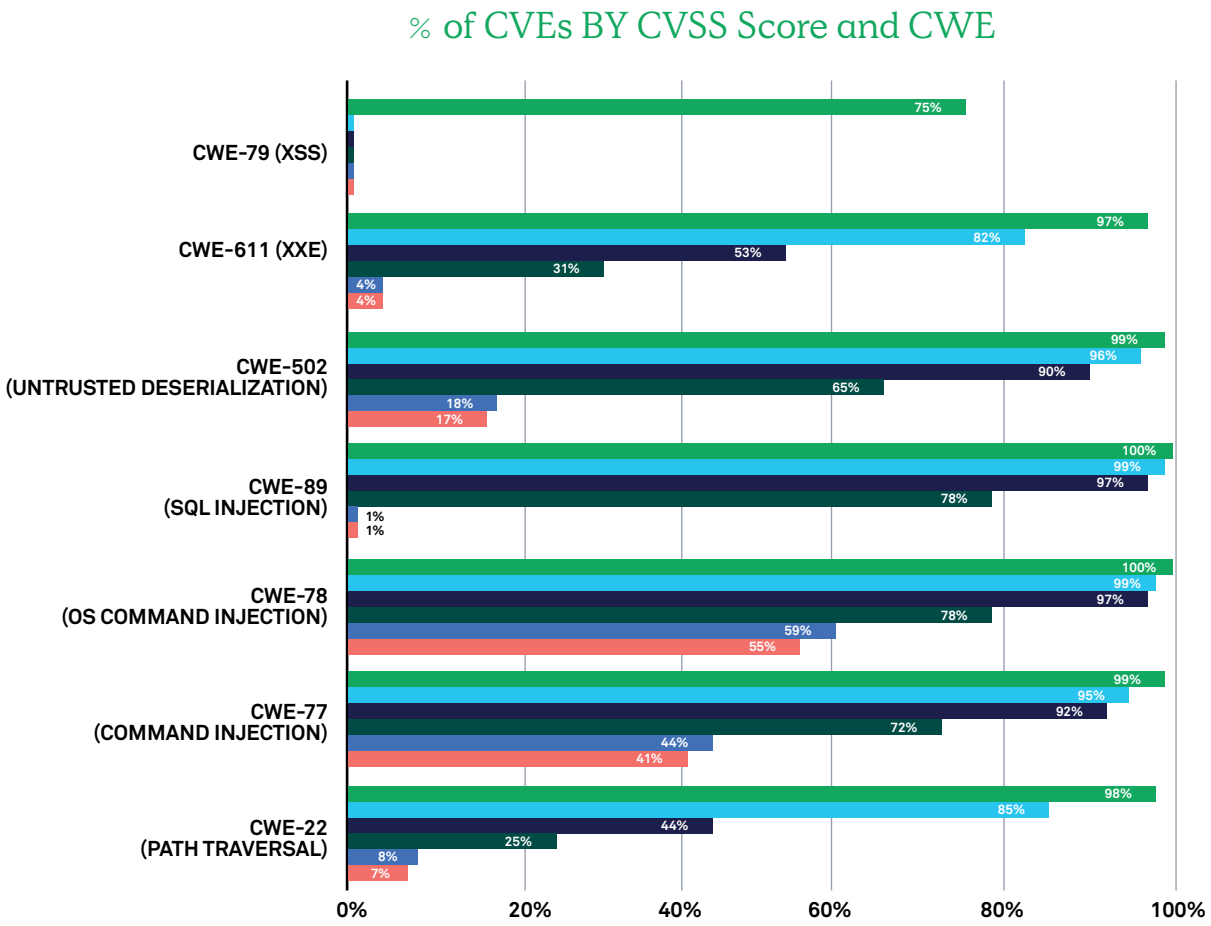


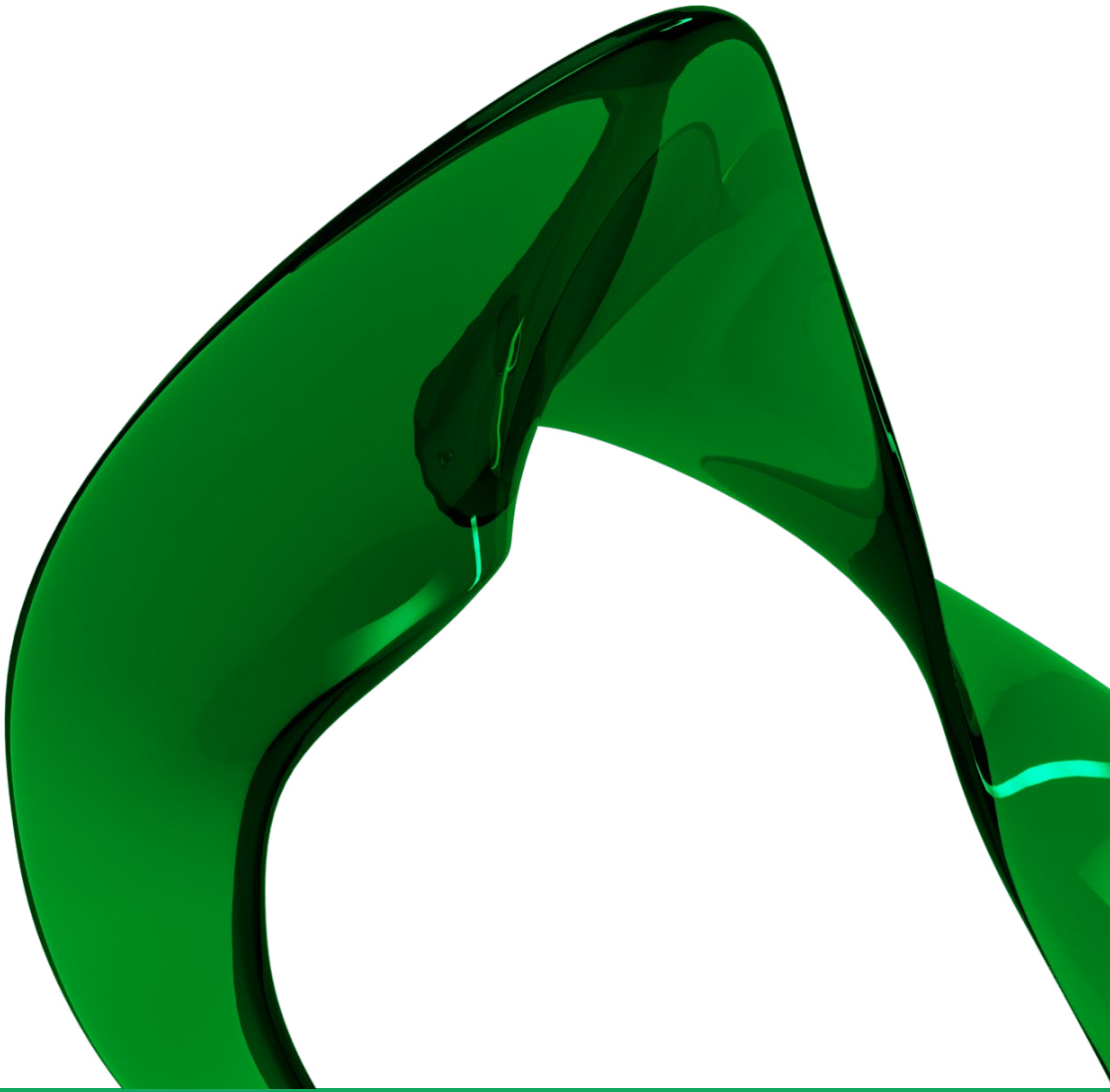
Schéma 6 : Les données montrent que le risque XSS est en moyenne beaucoup plus faible que les autres.

Les décisions relatives aux risques ne devraient pas être prises sur la base d'un seul point de référence comme celui-ci. Cependant, pour les développeurs et les équipes de sécurité qui doivent prioriser des réparations ou s'interrogent sur les classes de vulnérabilités ayant le plus d'impact (soit le but premier du score de base CVSS), il est intéressant de commencer par regarder les données du domaine public, par exemple les CVE, et la façon dont elles ont été notées.

Nous ne disons absolument pas qu'il ne faut pas considérer XSS comme une classe de vulnérabilité. Cependant, au fil du temps, XSS recule lentement dans le Top 10 de l'OWASP. Cette vulnérabilité y figure toujours, car elle demeure très importante pour de nombreux analystes sécurité – elle est utilisée dans les chaînes d'attaque et est toujours très présente sur Internet. D'ailleurs, aucun des autres CWE cités dans cet article n'arrive ne serait-ce qu'à la moitié du nombre de CVE affiché par XSS, le plus proche étant l'injection SQL avec 48 %.

À elles seules, ces données ne peuvent justifier la présence ou l'absence de XSS – ou des autres CWE cités dans cet article – dans votre top 10 des vulnérabilités. J'ai traqué des bugs sur de nombreuses plateformes de bug bounty, et j'ai pu remarquer une baisse des primes pour les soumissions XSS en raison de leur prévalence et du risque global. En résumé, faut-il corriger XSS ? OUI, absolument, mais il existe peut-être d'autres problèmes plus aigus sur lesquels nous devrions nous concentrer en priorité du fait de leur risque global.

Pour en savoir plus sur les vulnérabilités et attaques les plus récentes, consultez notre dernier rapport « Bimonthly Application Security Intelligence Report. »



Contrast Security est leader du marché de sécurité des applications grâce à une approche moderne qui intègre l'analyse du code et la prévention des attaques au cœur même des applications. Sa technologie d'instrumentation brevetée Deep Security bouscule totalement les approches traditionnelles par son observabilité intégrée et complète en fournissant une évaluation très précise et en assurant une protection continue du portefeuille d'applications tout entier. Ainsi, plus besoin de disposer d'infrastructures gourmandes en performances et d'experts spécialisés en sécurité. La plateforme de sécurité des applications Contrast accélère les cycles de développement, augmente l'efficacité, réduit les coûts et permet un déploiement rapide tout en protégeant les applications contre les menaces connues et inconnues.

240 3rd Street
2nd Floor
Los Altos, CA 94022
Phone: 888.371.1333
Fax: 650.397.4133

Contrast SECURITY

contrastsecurity.com