ASA AI

RAPPORT D'AUDIT — SÉCURITÉ WEB APPLICATIVE

Rapport d'audit WebAppSec

juice-shop-v20.2.0

Web Application Security Assessment

CIBLE	demo.owasp-juice.shop
AUDITEUR	KMG
TYPE DE SCAN	Boite blanche
RÉFÉRENCE	ASA-AI-WAS-20261010-073026
DATE	2026-10-10 07:30:26



SPÉCIMEN

Page laissée intentionnellement vierge

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Sommaire

Sommaire : clic droit > Mettre à jour les champs (ou F9) pour afficher les numéros de page.

SPÉCIMEN • ASA AI

SPÉCIMEN • ASA AI

Introduction

Une application web n'est jamais un objet figé. Elle vit, s'enrichit de fonctionnalités, intègre des bibliothèques tierces et hérite de dépendances que personne n'a vraiment choisies. Chacune de ces évolutions déplace, souvent de façon imperceptible, la frontière entre ce qui est exposé et ce qui est protégé. C'est précisément dans ce mouvement permanent que naissent les vulnérabilités : rarement dans le code écrit un jour de pleine attention, mais plutôt dans l'accumulation silencieuse des petits changements que l'on n'a pas réévalués à l'aune de la sécurité. Auditer une application, ce n'est donc pas chercher la faute d'un développeur : c'est prendre, à un instant donné, la mesure d'un système qui n'a pas cessé de bouger depuis sa dernière revue.

Ce rapport part d'un principe simple, hérité de la pratique du test d'intrusion comme de l'exploitation d'outils tels qu'OWASP ZAP, Burp Suite ou Tenable : on ne protège bien que ce que l'on mesure, et on ne mesure utilement que ce que l'on mesure régulièrement. Les pages qui suivent ne prétendent pas à l'exhaustivité d'un audit manuel approfondi ; elles établissent une base factuelle, reproductible et comparable dans le temps, sur laquelle fonder les décisions de sécurité et en démontrer le sérieux.

Pourquoi la sécurité applicative ne se négocie plus

Les applications web concentrent aujourd'hui l'essentiel de la valeur d'une organisation : ses données clients, ses processus métier, ses accès internes, parfois sa réputation tout entière. Elles sont, par construction, exposées — c'est leur raison d'être. Un attaquant n'a pas besoin de franchir un périmètre réseau durci lorsqu'un formulaire mal validé, un en-tête absent ou un cookie trop permissif lui ouvre directement la porte. Les statistiques des grands référentiels le confirment année après année : la très grande majorité des compromissions commence par la couche applicative, non par une prouesse technique, mais par l'exploitation méthodique de défauts connus et documentés depuis longtemps.

Le coût d'une faille applicative ne se mesure pas seulement en données exfiltrées. Il se compte en temps d'indisponibilité, en confiance perdue, en obligations de notification réglementaire, en heures de réponse à incident et, souvent, en remise en cause de projets entiers. Face à cela, la sécurité d'une application ne se décrète pas à la livraison : elle se vérifie, encore et encore, tant que l'application vit. Une posture saine n'est pas un état que l'on atteint une fois, mais un équilibre que l'on entretient.

De l'audit ponctuel à la discipline continue

Un test de sécurité réalisé une fois, aussi rigoureux soit-il, ne dit la vérité que sur l'état du système au jour où il a eu lieu. Dès le lendemain, un correctif appliqué, une dépendance mise à jour, une fonctionnalité ajoutée peuvent rouvrir une brèche que l'on croyait refermée — ou en créer une nouvelle. La notion même de « sécurisé » n'a de sens qu'associée à une date. C'est pourquoi les équipes sérieuses ont abandonné l'idée de l'audit-événement au profit d'une pratique régulière, répétée, intégrée au cycle de vie du logiciel.

Un scan récurrent ne remplace pas l'expertise humaine : il la décharge de la partie répétitive pour la concentrer là où elle est irremplaçable. Il fait ce que l'humain ne peut tenir dans la durée : surveiller sans relâche la posture de l'application, détecter les régressions dès qu'elles apparaissent, et transformer la sécurité d'un contrôle subi en un indicateur que l'on pilote, dont on suit la tendance d'un rapport à l'autre. Le présent document s'inscrit dans cette logique : il est une photographie, destinée à être comparée aux suivantes, et non un verdict définitif. Sa valeur croîtra avec sa répétition.

Une surface d'attaque qui se dérobe : composants et chaîne d'approvisionnement

La façon de construire les logiciels a profondément changé. On n'écrit plus une application, on l'assemble : frameworks, bibliothèques open source, services tiers, conteneurs, modules téléchargés en quelques secondes. Cette productivité a un revers — chaque composant importé apporte son propre historique de vulnérabilités,

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

son propre rythme de mises à jour, et parfois ses propres dépendances cachées, invisibles dans l'inventaire déclaré. La surface réellement exposée dépasse presque toujours celle que l'on croit maîtriser.

L'actualité récente l'a rappelé avec brutalité : une faille découverte dans une bibliothèque de journalisation très répandue, ou une porte dérobée glissée dans un composant de compression, suffit à mettre en danger, du jour au lendemain, des dizaines de milliers d'applications qui l'embarquaient sans le savoir. Maîtriser la sécurité d'une application, c'est désormais maîtriser aussi celle de tout ce qu'elle emprunte : connaître ses composants, leurs versions exactes, et les vulnérabilités publiquement répertoriées qui les concernent. C'est l'objet de la corrélation avec les bases de vulnérabilités (CVE) que ce rapport intègre, et de l'analyse de composition logicielle (SCA) qu'il recommande d'activer.

Des attaquants augmentés par l'intelligence artificielle

Le rapport de forces s'est déplacé. La reconnaissance d'une cible, l'identification de ses technologies, la rédaction d'un courriel d'hameçonnage crédible ou l'adaptation d'un exploit à un environnement particulier — autant de tâches qui demandaient hier du temps et une expertise rare — sont aujourd'hui accélérées, voire automatisées, par l'intelligence artificielle. L'attaquant ordinaire dispose d'une puissance de feu autrefois réservée à quelques spécialistes : il teste plus de cibles, plus vite, avec des charges mieux adaptées et une ingénierie sociale nettement plus convaincante.

Cette asymétrie nouvelle ne rend pas la défense impossible, mais elle change les délais. Le temps qui sépare la publication d'une vulnérabilité de ses premières exploitations de masse se compte désormais en heures, parfois moins. Une faiblesse laissée en l'état « parce qu'elle est mineure » peut devenir, à la faveur d'un outil automatisé, la première marche d'une compromission complète. La réponse n'est pas de courir plus vite que la machine, mais de réduire la fenêtre d'exposition : détecter tôt, prioriser juste, corriger vite — donc mesurer souvent. C'est exactement le rôle d'un dispositif de scan régulier comme celui-ci.

Anatomie d'une compromission : pourquoi les « petites » failles comptent

On imagine souvent l'attaque comme un coup d'éclat : une faille spectaculaire, exploitée d'un seul geste. La réalité est plus patiente et plus méthodique. Un attaquant progresse par enchaînement. Un en-tête de sécurité manquant ne donne accès à rien en soi ; mais combiné à un cookie de session accessible en JavaScript et à une faille d'injection de contenu, il devient le maillon qui transforme un défaut mineur en vol de session complet. C'est la raison pour laquelle un rapport sérieux ne hiérarchise pas seulement par gravité isolée : il donne à voir la surface dans son ensemble, car la dangerosité d'un défaut dépend autant de ce qui l'entoure que de lui-même.

Cette logique de chaînage explique pourquoi la divulgation d'une version de serveur, l'exposition d'un fichier de configuration ou une politique CORS trop permissive méritent d'être relevées même lorsqu'elles ne constituent pas, à elles seules, une vulnérabilité critique. Chacune réduit l'effort de l'attaquant : elle lui indique où frapper, lui épargne une étape de reconnaissance, ou élargit le périmètre qu'il peut atteindre une fois un premier pied posé. Réduire la surface d'exposition, c'est casser ces chaînes avant qu'elles ne se forment — souvent à un coût bien moindre que celui d'un incident.

À l'inverse, il serait contre-productif de traiter toutes les alertes avec la même urgence. Un audit utile sépare le signal du bruit : il distingue ce qui expose réellement l'organisation de ce qui relève du durcissement de confort, et il assume cette hiérarchie plutôt que de noyer le lecteur sous une liste indifférenciée. C'est précisément l'intention de la synthèse et du plan de correction priorisé qui structurent ce document.

La conformité comme conséquence, non comme objectif

Les cadres réglementaires et normatifs — RGPD, NIS2, PCI-DSS, ISO/IEC 27001, recommandations de l'ANSSI — convergent tous vers une même exigence : démontrer, preuves à l'appui, que l'on maîtrise ses risques dans la durée. La conformité n'est pas une fin en soi ; elle est la trace laissée par une pratique de sécurité saine. Une

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

organisation qui audite régulièrement ses applications, documente ses constats et suit leur remédiation satisfait naturellement à l'essentiel de ces exigences, et dispose des éléments probants à présenter en cas de contrôle ou d'incident.

À l'inverse, chercher à « cocher les cases » sans démarche de fond produit des documents sans portée, vite démentis par le premier incident réel. La différence entre les deux approches tient en un mot : la preuve. Un rapport daté, reproductible, rattaché à des référentiels reconnus et assorti d'un plan de correction constitue cette preuve. Ce document est conçu pour servir cette double fonction : améliorer concrètement la posture de l'application, et attester que l'audit a été mené selon les standards de la profession.

Trois regards complémentaires : reconnaissance, DAST et SAST

Aucune technique ne suffit seule, et c'est leur combinaison qui fait la valeur d'un audit. La reconnaissance regarde l'application de loin, comme un attaquant qui prépare son approche : quels ports, quels services, quelles versions, quelle surface. Les tests dynamiques (DAST) la sollicitent de l'extérieur, en fonctionnement, pour observer son comportement réel face à des requêtes légitimes puis à des charges contrôlées — c'est le domaine d'OWASP ZAP. L'analyse statique (SAST) ouvre le capot et lit le code source, là où le test dynamique n'ira jamais, pour remonter à la racine une faiblesse avant même qu'elle ne devienne exploitable — c'est le rôle de SonarQube.

Ces regards se corrigent mutuellement. Le DAST prouve l'exploitabilité mais ne voit que ce qu'il atteint ; le SAST couvre tout le code mais signale aussi des problèmes théoriques ; la corrélation CVE relie le tout aux vulnérabilités déjà connues du monde entier. Un constat confirmé par deux approches mérite une attention immédiate ; un constat isolé appelle une vérification. C'est cette lecture croisée, et non l'accumulation d'alertes, qui distingue un rapport utile d'une simple liste.

Lire ce rapport : sévérités et priorisation

Chaque constat porte une sévérité — Critique, Élevé, Modéré, Faible ou Information — et, lorsqu'elle s'applique, une cotation CVSS v3.1. Le score CVSS n'est pas un caprice d'expert : il décompose objectivement le risque selon le vecteur d'attaque, la complexité requise, les privilèges nécessaires, l'interaction de la victime et les impacts sur la confidentialité, l'intégrité et la disponibilité. Deux failles de même nom peuvent ainsi recevoir des scores très différents selon leur contexte d'exposition.

La règle de priorisation est simple et elle gouverne le plan de correction : on traite d'abord ce qui combine un impact fort et une facilité d'exploitation élevée. Une vulnérabilité critique exploitable depuis Internet sans authentification passe avant un durcissement d'en-tête, même si ce dernier est légitime. Les constats de niveau « Information » ne sont pas des défauts à corriger d'urgence, mais des éléments de contexte utiles à l'attaquant comme au défenseur ; on les traite quand le reste est maîtrisé.

Ce que contient ce rapport

Les sections qui suivent cartographient la surface exposée (reconnaissance réseau), examinent la robustesse des échanges chiffrés (TLS/SSL), inspectent la configuration du serveur et ses en-têtes de sécurité, puis restituent les résultats des tests dynamiques (DAST, via OWASP ZAP), de l'analyse statique du code (SAST, via SonarQube) et de la corrélation avec les vulnérabilités connues (CVE/NVD). Chaque constat est coté selon le standard CVSS v3.1, rattaché à une catégorie OWASP Top 10 et à un identifiant CWE, puis accompagné d'une preuve, d'un impact et d'une recommandation de remédiation. Une matrice de couverture ASVS/WSTG indique honnêtement ce qui a été testé et ce qui nécessite un moyen complémentaire. La méthodologie et les limites de l'évaluation sont précisées en fin de document. L'objectif n'est pas d'accabler, mais d'outiller une décision : savoir où porter l'effort, et dans quel ordre.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

1. Périmètre et conditions de l'évaluation

> *Objet. Cette section fixe le cadre contractuel et technique de l'audit : ce qui a été testé, par qui, avec quelle autorisation et selon quelle approche. Elle conditionne la portée des constats : un résultat n'a de valeur que rapporté au périmètre et aux limites dans lesquels il a été obtenu.*

Application auditée : juice-shop-v20.2.0
URL auditée : https://demo.owasp-juice.shop/
Hôte / Port : demo.owasp-juice.shop : 443
Type de scan : Boite blanche
Auditeur : KMG
Numéro d'autorisation : AUTO-00093
Environnement : Laboratoire ASA AI
Opérateur : ASA AI
Date du scan : 2026-10-10 07:30:26
Référence du rapport : ASA-AI-WAS-20261010-073026
Outils mobilisés : WebAppSec 3.0 ; Nmap : oui ; OWASP ZAP : oui (scan actif) ; SonarQube : oui ; NVD (CVE) : non
Exclusions et limites : Aucun test de deni de service. Perimetre limite aux URL fournies.

2. Synthèse exécutive

> *Objet. La synthèse offre au lecteur pressé — direction, responsable sécurité — une lecture immédiate du niveau de risque et de sa répartition. Le détail technique vient ensuite ; ici, l'on retient l'essentiel.*

Audit de sécurité de l'application « juice-shop-v20.2.0 » (demo.owasp-juice.shop) réalisé le 2026-10-10 07:30:26 par KMG — autorisation AUTO-00093. Niveau de risque global : CRITIQUE. 104 constat(s) : 13 critique(s), 7 élevé(s), 32 modéré(s), 47 faible(s), 5 information(s).

2.1 Matrice des risques

Sévérité	Nombre de constats
Critique	13
Élevé	7
Modéré	32
Faible	47
Information	5

2.2 Conformité OWASP Top 10 2021 — vue d'ensemble

Constats	Catégorie OWASP
0	A01 — Contrôle d'accès défaillant
6	A02 — Défaillances cryptographiques
14	A03 — Injection
0	A04 — Conception non sécurisée
82	A05 — Mauvaise configuration de sécurité
0	A06 — Composants vulnérables et obsolètes
0	A07 — Identification et authentification de mauvaise qualité
0	A08 — Manque d'intégrité des données et du logiciel
0	A09 — Carences de journalisation et de supervision

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

0	A10 — Falsification de requête côté serveur (SSRF)

3. Reconnaissance et surface d'exposition (Nmap)

Objet. La reconnaissance dresse la carte de ce qui est visible depuis le réseau : ports ouverts, services et versions, système d'exploitation. Chaque service exposé est une porte potentielle ; en connaître la nature et la version est la première étape de tout attaquant — et donc de tout défenseur.

Système détecté : F5 BIG-IP Local Traffic Manager load balancer (TMOS 11.6) (91%)

3.1 Ports et services

Port	État	Service / Version
21/tcp	open	tcpwrapped
25/tcp	filtered	smtp
80/tcp	open	http-proxy F5 BIG-IP load balancer http proxy
443/tcp	open	http Apache httpd 2.4.69
8080/tcp	open	http-proxy F5 BIG-IP load balancer http proxy

3.2 Scripts NSE (http-*, ssl-enum-ciphers)

Objet. Sortie brute des scripts d'énumération Nmap (NSE). Le script ssl-enum-ciphers liste les protocoles et suites de chiffrement réellement acceptés par le service. Ces données ne se lisent pas telles quelles : leur interprétation — ce qui est acceptable et ce qui ne l'est pas, et pourquoi — est fournie aux sections 4.1 et 4.2.

```
[80/http-server-header]

  BigIP
  Heroku



[80/http-title]

OWASP Juice Shop


[80/http-headers]

  Date: Sat, 10 Oct 2026 05:30:47 GMT
  Server: Heroku
  Accept-Ranges: bytes
  Access-Control-Allow-Origin: *
  Cache-Control: public, max-age=0
  Content-Length: 9393
  Content-Type: text/html; charset=UTF-8
  Etag: W/"24b1-1a1243b680e"
  Feature-Policy: payment 'self'
  Last-Modified: Sat, 10 Oct 2026 05:13:55 GMT
  Nel: {"report_to":"heroku-nel","response_headers":
["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}
  Report-To: {"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D\u0026sid=812dcc77-0bd0-43b1-a5f1-b25750382959\
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

```
u0026ts=1791610247"}],"max_age":3600}
  Reporting-Endpoints: heroku-nel="https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D&sid=812dcc77-0bd0-43b1-a5f1-
b25750382959&ts=1791610247"
  Vary: Accept-Encoding
  Via: 1.1 heroku-router
  X-Content-Type-Options: nosniff
  X-Frame-Options: SAMEORIGIN
  X-Recruiting: /#/jobs
  Connection: close

  (Request type: HEAD)



[80/http-methods]


  Supported Methods: GET HEAD POST OPTIONS



[443/http-headers]


  Date: Sat, 10 Oct 2026 05:30:47 GMT
  Server: Heroku
  Accept-Ranges: bytes
  Access-Control-Allow-Origin: *
  Cache-Control: public, max-age=0
  Content-Length: 9393
  Content-Type: text/html; charset=UTF-8
  Etag: W/"24b1-1a1243b680e"
  Feature-Policy: payment 'self'
  Last-Modified: Sat, 10 Oct 2026 05:13:55 GMT
  Nel: {"report_to":"heroku-nel","response_headers":
["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}
  Report-To: {"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D\u0026sid=812dcc77-0bd0-43b1-a5f1-b25750382959\
u0026ts=1791610247"}],"max_age":3600}
  Reporting-Endpoints: heroku-nel="https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D&sid=812dcc77-0bd0-43b1-a5f1-
b25750382959&ts=1791610247"
  Vary: Accept-Encoding
  Via: 1.1 heroku-router
  X-Content-Type-Options: nosniff
  X-Frame-Options: SAMEORIGIN
  X-Recruiting: /#/jobs
  Connection: close

  (Request type: HEAD)



[443/http-server-header]


  Apache/2.4.69 (Unix)
  Heroku
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

```
[443/ssl-enum-ciphers]

  TLSv1.2:
    ciphers:
      TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
      TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (ecdh_x25519) - A
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (ecdh_x25519) - A
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
    compressors:
      NULL
    cipher preference: server
  TLSv1.3:
    ciphers:
      TLS_AKE_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
      TLS_AKE_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
      TLS_AKE_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
    cipher preference: server
  least strength: A



[443/http-methods]

  Supported Methods: GET HEAD POST OPTIONS



[443/http-title]

OWASP Juice Shop



[8080/http-server-header]

  BigIP
  Heroku



[8080/http-title]

OWASP Juice Shop



[8080/http-methods]

  Supported Methods: GET HEAD POST OPTIONS



[8080/http-headers]

  Date: Sat, 10 Oct 2026 05:30:47 GMT
  Server: Heroku
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

```
  Accept-Ranges: bytes
  Access-Control-Allow-Origin: *
  Cache-Control: public, max-age=0
  Content-Length: 9393
  Content-Type: text/html; charset=UTF-8
  Etag: W/"24b1-1a1243b680e"
  Feature-Policy: payment 'self'
  Last-Modified: Sat, 10 Oct 2026 05:13:55 GMT
  Nel: {"report_to":"heroku-nel","response_headers":
["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}
  Report-To: {"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D\u0026sid=812dcc77-0bd0-43b1-a5f1-b25750382959\
u0026ts=1791610247"}],"max_age":3600}
  Reporting-Endpoints: heroku-nel="https://nel.heroku.com/reports?s=
%2FJKHoZ71CTxevEpLvMgGefs4YUpFH5nngH292HeXtbg%3D&sid=812dcc77-0bd0-43b1-a5f1-
b25750382959&ts=1791610247"
  Vary: Accept-Encoding
  Via: 1.1 heroku-router
  X-Content-Type-Options: nosniff
  X-Frame-Options: SAMEORIGIN
  X-Recruiting: /#/jobs
  Connection: close

  (Request type: HEAD)
```

4. Couche transport (TLS/SSL)

> *Objet. TLS protège la confidentialité et l'intégrité des échanges entre le navigateur et le serveur. Un protocole obsolète ou une suite de chiffrement faible expose les communications à l'interception (attaque de l'homme du milieu) et, pour certaines, à un déchiffrement a posteriori. On vérifie ici que seuls des protocoles et algorithmes robustes sont acceptés, et que le certificat est digne de confiance.*

Protocole négocié (handshake) : Tls13

Suite de chiffrement négociée : Aes256 256 bits

Sujet du certificat : CN=*.owasp-juice.shop

Émetteur : CN=Sectigo Public Server Authentication CA DV R36, O=Sectigo Limited, C=GB

Expiration : 04/13/2027 01:59:59

Signature : sha256RSA

Taille de clé : 2048 bits

4.1 Protocoles — acceptables / non acceptables

> *Règle de lecture. TLS 1.2 et TLS 1.3 sont acceptables ; TLS 1.0, TLS 1.1 et toutes les versions de SSL sont considérés comme non acceptables (dépréciés par la RFC 8996, non conformes PCI-DSS). La colonne « Justification » rappelle la menace associée.*

Protocole	Verdict	Justification
TLSv1.2	ACCEPTABLE	Conforme si les suites sont AEAD avec PFS. Recommandé par l'ANSSI.
TLSv1.3	ACCEPTABLE	Recommandé : uniquement des suites AEAD, PFS obligatoire, handshake durci.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

4.2 Suites de chiffrement — acceptables / non acceptables

Règle de lecture. Sont acceptables les suites AEAD (GCM, CHACHA20-POLY1305) avec confidentialité persistante (ECDHE/DHE). Sont à limiter les modes CBC et les suites sans PFS. Sont non acceptables RC4, 3DES, EXPORT, NULL et les MAC MD5. Chaque suite est jugée individuellement ci-dessous.

Suite	Protocole	Verdict	Justification
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	TLSv1.2	ACCEPTABLE	Chiffrement authentifié (AEAD) avec confidentialité persistante (PFS).
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	TLSv1.2	ACCEPTABLE	Chiffrement authentifié (AEAD) avec confidentialité persistante (PFS).
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	TLSv1.2	À LIMITER	Mode CBC (Lucky13, oracle de padding) : toléré en TLS 1.2, à retirer au profit d'AEAD.
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLSv1.2	À LIMITER	Mode CBC (Lucky13, oracle de padding) : toléré en TLS 1.2, à retirer au profit d'AEAD.
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	TLSv1.2	À LIMITER	Mode CBC (Lucky13, oracle de padding) : toléré en TLS 1.2, à retirer au profit d'AEAD.
TLS_AKE_WITH_AES_256_GCM_SHA384	TLSv1.3	ACCEPTABLE	Chiffrement authentifié (AEAD) avec confidentialité persistante (PFS).
TLS_AKE_WITH_CHACHA20_POLY1305_SHA256	TLSv1.3	ACCEPTABLE	Chiffrement authentifié (AEAD) avec confidentialité persistante (PFS).
TLS_AKE_WITH_AES_128_GCM_SHA256	TLSv1.3	ACCEPTABLE	Chiffrement authentifié (AEAD) avec confidentialité persistante (PFS).

4.3 Commentaire et recommandations

- Seuls des protocoles acceptables (TLS 1.2/1.3) ont été observés : configuration conforme sur ce point.
- Privilégier les suites AEAD avec PFS (ECDHE + GCM/CHACHA20) et retirer progressivement les suites CBC ou sans PFS.

5. Configuration serveur et en-têtes HTTP

Objet. Les en-têtes de sécurité HTTP constituent la première ligne de défense côté navigateur : ils limitent le clickjacking, l'injection de contenu, le vol de session ou la fuite d'informations. Leur absence n'est pas une vulnérabilité exploitable en soi, mais elle élargit la surface d'attaque et aggrave l'impact d'autres failles. On vérifie aussi les méthodes HTTP exposées, la politique CORS et les ressources sensibles accessibles sans authentification.

5.1 En-têtes de sécurité

En-tête	Statut	Valeur actuelle / recommandation
Strict-Transport-Security (HSTS)	ABSENT	Recommandé : max-age=31536000; includeSubDomains
Content-Security-Policy (CSP)	ABSENT	Recommandé : default-src 'self'
X-Frame-Options	Présent	SAMEORIGIN
X-Content-Type-Options	Présent	nosniff
Referrer-Policy	ABSENT	Recommandé : strict-origin-when-cross-origin
Permissions-Policy	ABSENT	Recommandé : Restreindre geolocation/camera/micro
Cross-Origin-Opener-Policy	ABSENT	Recommandé : same-origin
Cross-Origin-Resource-Policy	ABSENT	Recommandé : same-origin

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

5.2 Méthodes HTTP autorisées

Non divulguées.

5.3 CORS

ACAO: * ; ACAC:

5.4 Ressources sensibles sondées

```
/robots.txt [200]

/backup.zip [200]

/config.json [200]

/web.config [200]

/server-status [200]

/grafana/api/health [200]

/grafana/login [200]

/.well-known/security.txt [200]
```

5.5 Technologies détectées

Heroku

6. Tests applicatifs dynamiques (DAST — OWASP ZAP)

> *Objet. Le DAST (Dynamic Application Security Testing) teste l'application en fonctionnement, de l'extérieur, comme le ferait un attaquant. Le scan passif observe le trafic sans rien injecter ; le scan actif envoie des charges contrôlées pour révéler injections, XSS, défauts de contrôle d'accès, etc. Il complète l'analyse du code (SAST) : l'un voit ce que fait l'application, l'autre comment elle est écrite.*

Alerte	Risque	CWE	Paramètre	URL
Content Security Policy (CSP) Header Not Set	Medium	CWE-693		https://demo.owasp-juice.shop/
Cross-Domain Misconfiguration	Medium	CWE-264		https://demo.owasp-juice.shop/
HTTPS Content Available via HTTP	Low	CWE-311		https://demo.owasp-juice.shop/
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	CWE-497		https://demo.owasp-juice.shop/ftp/coupons_2013.md.bak
Strict-Transport-Security Header Not Set	Low	CWE-319		https://demo.owasp-juice.shop/
Timestamp Disclosure - Unix	Low	CWE-497	Reporting-Endpoints	https://demo.owasp-juice.shop/assets/public/favicon_js.ico
Modern Web Application	Info	CWE--1		https://demo.owasp-juice.shop/
Re-examine Cache-control Directives	Info	CWE-525	cache-control	https://demo.owasp-juice.shop/
User Agent Fuzzer	Info	CWE-0	Header User-Agent	https://demo.owasp-

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

				juice.shop/assets

Alertes OWASP ZAP (scan actif).

7. Analyse statique du code (SAST — SonarQube)

> *Objet. Le SAST (Static Application Security Testing) analyse le code source sans l'exécuter, pour repérer les vulnérabilités à la racine : flux d'une donnée non fiable jusqu'à une opération sensible, secrets codés en dur, cryptographie faible, gestion d'erreurs défaillante. Il voit ce que le test dynamique ne peut atteindre (chemins rarement empruntés) mais ne prouve pas l'exploitabilité réelle — d'où la complémentarité avec le DAST.*

Projet SonarQube : juice-shop-v20.2.0. 1497 problème(s) ouvert(s), dont 85 à caractère sécurité.

Sévérité	Type	Règle	Emplacement	Message
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/about/about.component.ts:120	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/administration/administration.component.ts:74	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/administration/administration.component.ts:92	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/data-export/data-export.component.ts:58	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/last-login-ip/last-login-ip.component.ts:40	Make sure disabling Angular built-in sanitization is safe here.
Critical	CODE_SMELL	typescript:S7652	juice-shop-juice-shop-c4e9b29/frontend/src/app/mat-search-bar/mat-search-bar.component.ts:35	Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)
Critical	CODE_SMELL	typescript:S7652	juice-shop-juice-shop-c4e9b29/frontend/src/app/mat-search-bar/mat-search-bar.component.ts:36	Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)
Critical	CODE_SMELL	typescript:S7652	juice-shop-juice-shop-c4e9b29/frontend/src/app/mat-search-bar/mat-search-bar.component.ts:37	Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)
Critical	CODE_SMELL	typescript:S7652	juice-shop-juice-shop-	Output bindings,

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

			c4e9b29/frontend/src/app/mat-search-bar/mat-search-bar.component.ts:38	including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)
Critical	CODE_SMELL	typescript:S7652	juice-shop-juice-shop-c4e9b29/frontend/src/app/mat-search-bar/mat-search-bar.component.ts:39	Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/score-board/score-board.component.ts:83	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/search-result/search-result.component.ts:111	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/search-result/search-result.component.ts:144	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	typescript:S6268	juice-shop-juice-shop-c4e9b29/frontend/src/app/track-result/track-result.component.ts:49	Make sure disabling Angular built-in sanitization is safe here.
Critical	VULNERABILITY	secrets:S6706	juice-shop-juice-shop-c4e9b29/infrastructure/terraform/networking.tf:171	Make sure this private key gets revoked, changed, and removed from the code.
Critical	VULNERABILITY	secrets:S6706	juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:21	Make sure this private key gets revoked, changed, and removed from the code.
Critical	VULNERABILITY	typescript:S6437	juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:42	Revoke and change this password, as it is compromised.
Critical	VULNERABILITY	secrets:S6706	juice-shop-juice-shop-c4e9b29/terraform/networking.tf:171	Make sure this private key gets revoked, changed, and removed from the code.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/administration.spec.ts:18	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/b2bOrder.spec.ts:3	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/b2bOrder.spec.ts:34	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:36	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:44	Add at least one assertion to this test case.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:57	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:79	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:106	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:110	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:134	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/basket.spec.ts:147	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/chatbot.spec.ts:11	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:12	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:35	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:52	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:61	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:71	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:81	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:91	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:102	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:123	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/complain.spec.ts:137	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/contact.spec.ts:134	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/contact.spec.ts:184	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/dataErasure.spec.ts:7	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/	Add at least one assertion to this test

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

			e2e/ dataExport.spec.ts:23	case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/deluxe.spec.ts:3	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:3	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:18	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:28	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:39	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:47	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:57	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:67	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:75	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:83	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/ directAccess.spec.ts:93	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/forgedJwt.spec.ts:3	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/forgedJwt.spec.ts:16	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/forgedJwt.spec.ts:34	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:7	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:13	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:26	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:39	Add at least one assertion to this test case.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:52	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:65	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:78	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:91	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:104	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:117	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:137	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:147	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:153	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:166	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/login.spec.ts:178	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/metrics.spec.ts:3	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/noSql.spec.ts:10	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/noSql.spec.ts:31	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/noSql.spec.ts:56	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/noSql.spec.ts:79	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/noSql.spec.ts:123	Add at least one assertion to this test case.
Critical	CODE_SMELL	typescript:S2699	juice-shop-juice-shop-c4e9b29/test/cypress/e2e/profile.spec.ts:6	Add at least one assertion to this test case.

8. Composants tiers (SCA)

> *Objet. Le SCA (Software Composition Analysis) inventorie les dépendances de l'application et les confronte aux vulnérabilités connues. La plupart des compromissions récentes majeures sont passées par un composant tiers vulnérable, non par le code maison.*

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Le SCA exige les manifestes de dépendances (package.json, requirements.txt, pom.xml, etc.). À activer lors de l'analyse du code source : bibliothèques vulnérables (CVE/EPSS), licences, dépendances obsolètes.

9. Vulnérabilités connues (corrélation CVE — NVD)

> *Objet. On rapproche les composants détectés (services relevés par Nmap, en-têtes Server/X-Powered-By) des vulnérabilités publiquement répertoriées dans la base NVD. Une CVE connue sur un composant exposé est une cible de choix : son exploitation est souvent documentée et outillée. La précision dépend de l'exactitude de la version détectée — ces résultats sont à confirmer sur la version réellement déployée.*

Aucune CVE corrélée automatiquement (pas de version précise détectée, ou recherche NVD non réalisée). Pour l'activer : le poste doit disposer d'un accès Internet vers services.nvd.nist.gov ; connaître les versions exactes des composants améliore la précision. Bases gratuites : NVD (nvd.nist.gov), CVE Mitre (cve.org).

10. Matrice de couverture OWASP ASVS / WSTG

> *Objet. Cette matrice situe l'audit par rapport aux référentiels OWASP ASVS (exigences de vérification) et WSTG (guide de test). Elle distingue honnêtement ce qui a été réellement testé de ce qui nécessite un moyen complémentaire (scan actif, accès authentifié, revue manuelle). Le statut est calculé d'après les outils effectivement disponibles lors de ce scan.*

Chapitre ASVS	Réf WSTG	Statut
V1 Architecture	Revue	N/A (revue manuelle)
V2 Authentification	WSTG-ATHN	Testé (ZAP actif)
V3 Gestion de session	WSTG-SESS	Testé partiellement (cookies)
V4 Contrôle d'accès	WSTG-ATHZ	Testé (ZAP actif)
V5 Validation / Injection	WSTG-INPV	Testé (ZAP actif)
V6 Cryptographie stockée	WSTG-CRYP	Testé (SonarQube)
V7 Gestion des erreurs / journal.	WSTG-ERRH	Non testé (visibilité interne requise)
V8 Données / Confidentialité	WSTG-ATHN	Testé partiellement (TLS, cookies)
V9 Communications (TLS)	WSTG-CRYP	Testé (passif) + Nmap
V10 Code malveillant / SRI	WSTG-CLNT	Testé (SonarQube)
V11 Logique métier	WSTG-BUSL	Non testé (test manuel requis)
V12 Fichiers / ressources	WSTG-CONF	Testé (exposition, méthodes)
V13 API / GraphQL	WSTG-APIT	Testé (ZAP actif)
V14 Configuration	WSTG-CONF	Testé (en-têtes, divulgation) + Nmap

11. Conformité OWASP Top 10 2021 — détail par catégorie

Chaque catégorie est rappelée, puis les constats associés sont détaillés : description, vecteur CVSS v3.1, impact, preuve, remédiation et références.

A01 — Contrôle d'accès défaillant

Le contrôle d'accès garantit qu'un utilisateur ne peut agir au-delà de ses permissions. Ses défaillances mènent à la divulgation, la modification ou la destruction de données par un acteur non autorisé.

Aucun constat dans cette catégorie.

A02 — Défaillances cryptographiques

Protection insuffisante des données en transit et au repos : TLS obsolète, suites de chiffrement faibles, absence de chiffrement, gestion déficiente des clés et des certificats.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Critique] SAST-060 — SonarQube : Make sure this private key gets revoked, changed, and removed from the code.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle secrets:S6706.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/infrastructure/terraform/networking.tf:171
```

Remédiation

Corriger le code selon la regle SonarQube secrets:S6706.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-061 — SonarQube : Make sure this private key gets revoked, changed, and removed from the code.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle secrets:S6706.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:21
```

Remédiation

Corriger le code selon la regle SonarQube secrets:S6706.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-063 — SonarQube : Revoke and change this password, as it is compromised.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6437.

Score CVSS v3.1 : 8.6/10 (Critique)

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:42
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6437.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-084 — SonarQube : Make sure this private key gets revoked, changed, and removed from the code.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle secrets:S6706.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/terraform/networking.tf:171
```

Remédiation

Corriger le code selon la regle SonarQube secrets:S6706.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-062 — SonarQube : Make sure this weak hash algorithm is not used in a sensitive context here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S4790.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

`juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:41`

Remédiation

Corriger le code selon la regle SonarQube typescript:S4790.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-079 — SonarQube : Make sure this weak hash algorithm is not used in a sensitive context here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S4790.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A02:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

`juice-shop-juice-shop-c4e9b29/scripts/package.mjs:121`

Remédiation

Corriger le code selon la regle SonarQube javascript:S4790.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A02:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

A03 — Injection

Des données non fiables sont interprétées comme une commande ou une requête (SQL, NoSQL, OS, LDAP, XSS). L'attaquant peut exécuter des actions imprévues ou accéder à des données sans autorisation.

[Critique] SAST-038 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

`juice-shop-juice-shop-c4e9b29/frontend/src/app/about/about.component.ts:120`

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-039 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/administration/administration.component.ts:74
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-040 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/administration/administration.component.ts:92
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-043 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/data-export/data-export.component.ts:58
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-044 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/last-login-ip/last-login-ip.component.ts:40
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Critique] SAST-045 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/score-board/score-board.component.ts:83
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-046 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/search-result/search-result.component.ts:111
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-047 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/search-result/search-result.component.ts:144
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Critique] SAST-048 — SonarQube : Make sure disabling Angular built-in sanitization is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S6268.

Score CVSS v3.1 : 8.6/10 (Critique)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/track-result/track-result.component.ts:49
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S6268.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-068 — SonarQube : Make sure that this dynamic injection or execution of code is safe.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S1523.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/xml.ts:38
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S1523.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-069 — SonarQube : Make sure that this dynamic injection or execution of code is safe.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S1523.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/b2bOrder.ts:23
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S1523.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-075 — SonarQube : Make sure that this dynamic injection or execution of code is safe.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S1523.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:22
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S1523.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-076 — SonarQube : Make sure that this dynamic injection or execution of code is safe.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S1523.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/fileUpload.ts:109
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S1523.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Élevé] SAST-077 — SonarQube : Make sure that this dynamic injection or execution of code is safe.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S1523.

Score CVSS v3.1 : 7/10 (Élevé)

Référence : | OWASP A03:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/userProfile.ts:65
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S1523.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A03:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

A04 — Conception non sécurisée

Absence ou faiblesse de contrôles de sécurité dès la conception : modélisation des menaces, patrons de conception sécurisés et limites métier insuffisamment pensés.

Aucun constat dans cette catégorie.

A05 — Mauvaise configuration de sécurité

Configuration par défaut non durcie, en-têtes de sécurité absents, méthodes HTTP superflues, messages d'erreur verbeux, services et ressources exposés sans nécessité.

[Modéré] HDR-content-security-policy — Content-Security-Policy (CSP) absent

Description

En-tete 'content-security-policy' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N | Score 6.1/10 | Modéré

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Modifiée
Complexité d'attaque	Faible	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Faible
Interaction utilisateur	Requise	Disponibilité	Aucune

Référence : CWE-693 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : default-src 'self'

Validation : Automatique - a confirmer manuellement

Références

- OWASP Secure Headers Project : https://owasp.org/www-project-secure-headers/
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-693 : https://cwe.mitre.org/

[Modéré] HDR-strict-transport-security — Strict-Transport-Security (HSTS) absent

Description

En-tete 'strict-transport-security' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N | Score 5.3/10 | Modéré

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Élevée	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Faible
Interaction utilisateur	Requise	Disponibilité	Aucune

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Référence : CWE-319 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : max-age=31536000; includeSubDomains

Validation : Automatique - a confirmer manuellement

Références

- Transport Layer Protection Cheat Sheet : https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Protection_Cheat_Sheet.html
- HSTS — MDN : https://developer.mozilla.org/fr/docs/Web/HTTP/Headers/Strict-Transport-Security
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-319 : https://cwe.mitre.org/

[Modéré] ZAP-001 — Content Security Policy (CSP) Header Not Set

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 5.3/10 (Modéré)

Référence : CWE-693 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/
```

Remédiation

Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Secure Headers Project : https://owasp.org/www-project-secure-headers/
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-693 : https://cwe.mitre.org/

[Modéré] ZAP-002 — Cross-Domain Misconfiguration

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 5.3/10 (Modéré)

Référence : CWE-264 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/
```

Remédiation

Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance).Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-264 : https://cwe.mitre.org/

[Modéré] SAST-003 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:305
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-004 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:323
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-005 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:381
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-006 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:755
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-007 — SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S6258.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_1.tf:121
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S6258.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-009 — SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S6258.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_2.tf:121
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S6258.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Modéré] SAST-011 — SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S6258.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_3_correct.tf:121
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S6258.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-037 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/Services/conversation-storage.service.ts:17
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-041 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/chatbot/chat-welcome-screen/chat-welcome-screen.component.ts:71
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-042 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/app/coding-challenge-page/components/coding-challenge-fix-it/coding-challenge-fix-it.component.ts:120
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-049 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6359
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-050 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6426
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-051 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6426
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-052 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6434
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-053 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6442
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-054 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:6450
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-055 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:15504
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Modéré] SAST-056 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:15534
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-057 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/frontend/src/assets/private/three.js:15567
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-058 — SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S6258.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/infrastructure/terraform/networking.tf:122
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S6258.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-064 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:53
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-070 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:14
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-071 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:15
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-072 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:16
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-073 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:18
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-074 — SonarQube : Make sure that using this pseudorandom number generator is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S2245.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/routes/captcha.ts:19
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S2245.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Modéré] SAST-081 — SonarQube : Make sure the content length limit is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5693.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/server.ts:719
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5693.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Modéré] SAST-082 — SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S6258.

Score CVSS v3.1 : 5/10 (Modéré)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/terraform/networking.tf:122
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S6258.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] INF-server — Divulgation technologie (server)

Description

En-tete 'server' revele l implementation.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

CVSS v3.1 : CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N | Score 3.1/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Faible	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Aucune	Disponibilité	Aucune

Référence : CWE-200 | OWASP A05:2021 | Source : Analyse passive

Impact

Ciblage d exploits versionnes.

Preuve :

```
server: Heroku
```

Remédiation

Masquer 'server'.

Validation : Automatique - a confirmer manuellement

Références

- Information Exposure — CWE-200 : https://cwe.mitre.org/data/definitions/200.html
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-200 : https://cwe.mitre.org/

[Faible] INF-via — Divulgation technologie (via)

Description

En-tete 'via' revele l implementation.

CVSS v3.1 : CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N | Score 3.1/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Faible	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Aucune	Disponibilité	Aucune

Référence : CWE-200 | OWASP A05:2021 | Source : Analyse passive

Impact

Ciblage d exploits versionnes.

Preuve :

```
via: 1.1 heroku-router
```

Remédiation

Masquer 'via'.

Validation : Automatique - a confirmer manuellement

Références

- Information Exposure — CWE-200 : https://cwe.mitre.org/data/definitions/200.html
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

- CWE Mitre — CWE-200 : https://cwe.mitre.org/

[Faible] ZAP-003 — HTTPS Content Available via HTTP

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 3.1/10 (Faible)

Référence : CWE-311 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/
```

Remédiation

Ensure that your web server, application server, load balancer, etc. is configured to only serve such content via HTTPS. Consider implementing HTTP Strict Transport Security.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-311 : https://cwe.mitre.org/

[Faible] ZAP-004 — Server Leaks Version Information via "Server" HTTP Response Header Field

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 3.1/10 (Faible)

Référence : CWE-497 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/ftp/coupons_2013.md.bak
```

Remédiation

Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-497 : https://cwe.mitre.org/

[Faible] ZAP-005 — Strict-Transport-Security Header Not Set

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 3.1/10 (Faible)

Référence : CWE-319 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/
```

Remédiation

Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

Validation : Automatique - a confirmer manuellement

Références

- Transport Layer Protection Cheat Sheet : https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Protection_Cheat_Sheet.html
- HSTS — MDN : https://developer.mozilla.org/fr/docs/Web/HTTP/Headers/Strict-Transport-Security
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-319 : https://cwe.mitre.org/

[Faible] ZAP-006 — Timestamp Disclosure - Unix

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 3.1/10 (Faible)

Référence : CWE-497 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/assets/public/favicon_js.ico  |  Parametre: Reporting-Endpoints
```

Remédiation

Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-497 : https://cwe.mitre.org/

[Faible] SAST-001 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:86
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-002 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/datacreator.ts:173
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-008 — SonarQube : Make sure allowing clear-text traffic is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_1.tf:159
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Remédiation

Corriger le code selon la regle SonarQube terraform:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-010 — SonarQube : Make sure allowing clear-text traffic is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_2.tf:159
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-012 — SonarQube : Make sure allowing clear-text traffic is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/iacLeakedKeyChallenge_3_correct.tf:159
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Faible] SAST-013 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_1.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-014 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_1.ts:7
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-015 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_1.ts:9
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-016 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_2.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-017 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_2.ts:7
```

Remédiation

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-018 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_2.ts:9
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-019 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_3.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Faible] SAST-020 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_3.ts:7
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-021 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_3.ts:9
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-022 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_4_correct.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-023 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_4_correct.ts:7
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-024 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectChallenge_4_correct.ts:9
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-025 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_1.ts:5
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-026 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_1.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Faible] SAST-027 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_1.ts:8
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-028 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_2.ts:5
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-029 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_2.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-030 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_2.ts:8
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-031 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_3_correct.ts:3
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-032 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_3_correct.ts:4
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-033 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_3_correct.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Faible] SAST-034 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_4.ts:5
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-035 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_4.ts:6
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-036 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/data/static/codefixes/redirectCryptoCurrencyChallenge_4.ts:8
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-059 — SonarQube : Make sure allowing clear-text traffic is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/infrastructure/terraform/networking.tf:160
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-065 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:127
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-066 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:128
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-067 — SonarQube : Using http protocol is insecure. Use https instead.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/lib/insecurity.ts:130
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

[Faible] SAST-078 — SonarQube : Make sure the "PATH" variable only contains fixed, unwriteable directories.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle javascript:S4036.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/scripts/package.mjs:20
```

Remédiation

Corriger le code selon la regle SonarQube javascript:S4036.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-080 — SonarQube : Make sure this debug feature is deactivated before delivering the code in production.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle typescript:S4507.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/server.ts:701
```

Remédiation

Corriger le code selon la regle SonarQube typescript:S4507.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-083 — SonarQube : Make sure allowing clear-text traffic is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle terraform:S5332.

Score CVSS v3.1 : 3/10 (Faible)

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/terraform/networking.tf:160
```

Remédiation

Corriger le code selon la regle SonarQube terraform:S5332.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] SAST-085 — SonarQube : The "alpine" image runs with "root" as the default user. Make sure it is safe here.

Description

Probleme de securite releve par l analyse statique du code (SAST) - regle docker:S6471.

Score CVSS v3.1 : 3/10 (Faible)

Référence : | OWASP A05:2021 | Source : SonarQube (SAST)

Impact

Faiblesse presente dans le code source, corrigeable a la racine avant deploiement.

Preuve :

```
juice-shop-juice-shop-c4e9b29/test/smoke/Dockerfile:1
```

Remédiation

Corriger le code selon la regle SonarQube docker:S6471.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

[Faible] HDR-referrer-policy — Referrer-Policy absent

Description

En-tete 'referrer-policy' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N | Score 2.6/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Élevée	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Requise	Disponibilité	Aucune

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Référence : CWE-200 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : strict-origin-when-cross-origin

Validation : Automatique - a confirmer manuellement

Références

- Information Exposure — CWE-200 : https://cwe.mitre.org/data/definitions/200.html
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-200 : https://cwe.mitre.org/

[Faible] HDR-permissions-policy — Permissions-Policy absent

Description

En-tete 'permissions-policy' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N | Score 2.6/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Élevée	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Requise	Disponibilité	Aucune

Référence : CWE-693 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : Restreindre geolocation/camera/micro

Validation : Automatique - a confirmer manuellement

Références

- OWASP Secure Headers Project : https://owasp.org/www-project-secure-headers/
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-693 : https://cwe.mitre.org/

[Faible] HDR-cross-origin-opener-policy — Cross-Origin-Opener-Policy absent

Description

En-tete 'cross-origin-opener-policy' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N | Score 2.6/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Élevée	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Requise	Disponibilité	Aucune

Référence : CWE-668 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : same-origin

Validation : Automatique - a confirmer manuellement

Références

- Cross-Origin Resource Policy — MDN : https://developer.mozilla.org/en-US/docs/Web/HTTP/Cross-Origin_Resource_Policy
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — CWE-668 : https://cwe.mitre.org/

[Faible] HDR-cross-origin-resource-policy — Cross-Origin-Resource-Policy absent

Description

En-tete 'cross-origin-resource-policy' absent.

CVSS v3.1 : CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N | Score 2/10 | Faible

Métrique	Valeur	Métrique	Valeur
Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Élevée	Confidentialité	Faible
Privilèges requis	Aucun	Intégrité	Aucune
Interaction utilisateur	Requise	Disponibilité	Aucune

Référence : CWE-668 | OWASP A05:2021 | Source : Analyse passive

Impact

Surface d attaque navigateur accrue.

Preuve :

```
Absent de la reponse HTTP
```

Remédiation

Ajouter : same-origin

Validation : Automatique - a confirmer manuellement

Références

- Cross-Origin Resource Policy — MDN : https://developer.mozilla.org/en-US/docs/Web/HTTP/Cross-Origin_Resource_Policy
- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

• CWE Mitre — CWE-668 : https://cwe.mitre.org/

[Information] ZAP-007 — Modern Web Application

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 0/10 (Information)

Référence : CWE--1 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/
```

Remédiation

This is an informational alert and so no changes are required.

Validation : Automatique - a confirmer manuellement

Références

• OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
• CWE Mitre — CWE--1 : https://cwe.mitre.org/

[Information] ZAP-008 — Re-examine Cache-control Directives

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Score CVSS v3.1 : 0/10 (Information)

Référence : CWE-525 | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/  |  Parametre: cache-control
```

Remédiation

For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".

Validation : Automatique - a confirmer manuellement

Références

• OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
• CWE Mitre — CWE-525 : https://cwe.mitre.org/

[Information] ZAP-009 — User Agent Fuzzer

Description

Alerte remontee par OWASP ZAP lors du scan actif (DAST), sur l application en fonctionnement.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

Score CVSS v3.1 : 0/10 (Information)

Référence : | OWASP A05:2021 | Source : OWASP ZAP (DAST)

Impact

Faiblesse exploitable a distance sur l application exposee.

Preuve :

```
URL: https://demo.owasp-juice.shop/assets  |  Parametre: Header User-Agent
```

Remédiation

Appliquer la recommandation OWASP ZAP associee a cette alerte.

Validation : Automatique - a confirmer manuellement

Références

- OWASP Top 10 2021 — A05:2021 : https://owasp.org/Top10/
- CWE Mitre — : https://cwe.mitre.org/

A06 — Composants vulnérables et obsolètes

Utilisation de composants (serveur, frameworks, bibliothèques) présentant des vulnérabilités connues (CVE) ou arrivés en fin de support.

Aucun constat dans cette catégorie.

A07 — Identification et authentification de mauvaise qualité

Faiblesses de l'authentification et de la gestion de session : attaques par force brute possibles, secrets faibles, jetons de session mal protégés.

Aucun constat dans cette catégorie.

A08 — Manque d'intégrité des données et du logiciel

Mises à jour, données ou chaînes CI/CD non vérifiées (signature, intégrité), désérialisation non sûre de données non fiables.

Aucun constat dans cette catégorie.

A09 — Carences de journalisation et de supervision

Journalisation, détection et réponse insuffisantes : les attaques ne sont ni tracées ni alertées à temps, ce qui allonge le temps de compromission.

Aucun constat dans cette catégorie.

A10 — Falsification de requête côté serveur (SSRF)

L'application récupère une ressource distante sans valider l'URL fournie par l'utilisateur, permettant d'atteindre des systèmes internes normalement inaccessibles.

Aucun constat dans cette catégorie.

Autres constats (hors catégorie OWASP mappée)

[Information] EXP-/robots.txt — robots.txt expose

Description

Ressource de reconnaissance accessible.

Score CVSS v3.1 : 0/10 (Information)

Impact

Info utile a un attaquant.

Remédiation

Limiter si inutile.

[Information] EXP-/.well-known/security.txt — security.txt present

Description

Ressource de reconnaissance accessible.

Score CVSS v3.1 : 0/10 (Information)

Impact

Info utile a un attaquant.

Remédiation

Limiter si inutile.

12. Plan de correction priorisé

> *Objet. Le plan ordonne les actions correctives par sévérité décroissante : traiter d'abord ce qui expose le plus. Il constitue la feuille de route de remédiation à suivre jusqu'au prochain audit.*

Priorité	Sévérité	Constat	Action corrective
1	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
2	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
3	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
4	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
5	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
6	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
7	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
8	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
9	Critique	SonarQube : Make sure disabling Angular built-in sanitization is safe here.	Corriger le code selon la regle SonarQube typescript:S6268.
10	Critique	SonarQube : Make sure this private key gets revoked,	Corriger le code selon la regle SonarQube secrets:S6706.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

		changed, and removed from the code.	
11	Critique	SonarQube : Make sure this private key gets revoked, changed, and removed from the code.	Corriger le code selon la regle SonarQube secrets:S6706.
12	Critique	SonarQube : Revoke and change this password, as it is compromised.	Corriger le code selon la regle SonarQube typescript:S6437.
13	Critique	SonarQube : Make sure this private key gets revoked, changed, and removed from the code.	Corriger le code selon la regle SonarQube secrets:S6706.
14	Élevé	SonarQube : Make sure this weak hash algorithm is not used in a sensitive context here.	Corriger le code selon la regle SonarQube typescript:S4790.
15	Élevé	SonarQube : Make sure that this dynamic injection or execution of code is safe.	Corriger le code selon la regle SonarQube typescript:S1523.
16	Élevé	SonarQube : Make sure that this dynamic injection or execution of code is safe.	Corriger le code selon la regle SonarQube typescript:S1523.
17	Élevé	SonarQube : Make sure that this dynamic injection or execution of code is safe.	Corriger le code selon la regle SonarQube typescript:S1523.
18	Élevé	SonarQube : Make sure that this dynamic injection or execution of code is safe.	Corriger le code selon la regle SonarQube typescript:S1523.
19	Élevé	SonarQube : Make sure that this dynamic injection or execution of code is safe.	Corriger le code selon la regle SonarQube typescript:S1523.
20	Élevé	SonarQube : Make sure this weak hash algorithm is not used in a sensitive context here.	Corriger le code selon la regle SonarQube javascript:S4790.
21	Modéré	Content-Security-Policy (CSP) absent	Ajouter : default-src 'self'
22	Modéré	Strict-Transport-Security (HSTS) absent	Ajouter : max-age=31536000; includeSubDomains
23	Modéré	Content Security Policy (CSP) Header Not Set	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.
24	Modéré	Cross-Domain Misconfiguration	Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance).Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.
25	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
26	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
27	Modéré	SonarQube : Make sure that using this pseudorandom	Corriger le code selon la regle SonarQube typescript:S2245.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

		number generator is safe here.	
28	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
29	Modéré	SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.	Corriger le code selon la regle SonarQube terraform:S6258.
30	Modéré	SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.	Corriger le code selon la regle SonarQube terraform:S6258.
31	Modéré	SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.	Corriger le code selon la regle SonarQube terraform:S6258.
32	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
33	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
34	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
35	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
36	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
37	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
38	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
39	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
40	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
41	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
42	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
43	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube javascript:S2245.
44	Modéré	SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.	Corriger le code selon la regle SonarQube terraform:S6258.
45	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
46	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
47	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
48	Modéré	SonarQube : Make sure that	Corriger le code selon la regle

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

		using this pseudorandom number generator is safe here.	SonarQube typescript:S2245.
49	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
50	Modéré	SonarQube : Make sure that using this pseudorandom number generator is safe here.	Corriger le code selon la regle SonarQube typescript:S2245.
51	Modéré	SonarQube : Make sure the content length limit is safe here.	Corriger le code selon la regle SonarQube typescript:S5693.
52	Modéré	SonarQube : Omitting "access_logs" makes logs incomplete. Make sure it is safe here.	Corriger le code selon la regle SonarQube terraform:S6258.
53	Faible	Divulgation technologie (server)	Masquer 'server'.
54	Faible	Divulgation technologie (via)	Masquer 'via'.
55	Faible	HTTPS Content Available via HTTP	Ensure that your web server, application server, load balancer, etc. is configured to only serve such content via HTTPS. Consider implementing HTTP Strict Transport Security.
56	Faible	Server Leaks Version Information via "Server" HTTP Response Header Field	Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.
57	Faible	Strict-Transport-Security Header Not Set	Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.
58	Faible	Timestamp Disclosure - Unix	Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.
59	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
60	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
61	Faible	SonarQube : Make sure allowing clear-text traffic is safe here.	Corriger le code selon la regle SonarQube terraform:S5332.
62	Faible	SonarQube : Make sure allowing clear-text traffic is safe here.	Corriger le code selon la regle SonarQube terraform:S5332.
63	Faible	SonarQube : Make sure allowing clear-text traffic is safe here.	Corriger le code selon la regle SonarQube terraform:S5332.
64	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
65	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
66	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
67	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

68	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
69	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
70	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
71	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
72	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
73	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
74	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
75	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
76	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
77	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
78	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
79	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
80	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
81	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
82	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
83	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
84	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
85	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
86	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
87	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
88	Faible	SonarQube : Make sure allowing clear-text traffic is safe here.	Corriger le code selon la regle SonarQube terraform:S5332.
89	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

90	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
91	Faible	SonarQube : Using http protocol is insecure. Use https instead.	Corriger le code selon la regle SonarQube typescript:S5332.
92	Faible	SonarQube : Make sure the "PATH" variable only contains fixed, unwriteable directories.	Corriger le code selon la regle SonarQube javascript:S4036.
93	Faible	SonarQube : Make sure this debug feature is deactivated before delivering the code in production.	Corriger le code selon la regle SonarQube typescript:S4507.
94	Faible	SonarQube : Make sure allowing clear-text traffic is safe here.	Corriger le code selon la regle SonarQube terraform:S5332.
95	Faible	SonarQube : The "alpine" image runs with "root" as the default user. Make sure it is safe here.	Corriger le code selon la regle SonarQube docker:S6471.
96	Faible	Referrer-Policy absent	Ajouter : strict-origin-when-cross-origin
97	Faible	Permissions-Policy absent	Ajouter : Restreindre geolocation/camera/micro
98	Faible	Cross-Origin-Opener-Policy absent	Ajouter : same-origin
99	Faible	Cross-Origin-Resource-Policy absent	Ajouter : same-origin

13. Annexes techniques

13.1 En-têtes HTTP bruts

```
Vary: Accept-Encoding

Server: Heroku

X-Content-Type-Options: nosniff

Reporting-Endpoints: heroku-nel="https://nel.heroku.com/reports?s=RhSrCTT
%2F2bPrLl6tC2HcugsZXgnjrvQ7DhT%2Bk0hyEVI%3D&sid=812dcc77-0bd0-43b1-a5f1-b25750382959&ts=1791610227"

X-Recruiting: /#/jobs

Cache-Control: public, max-age=0

Content-Length: 9393

ETag: W/"24b1-1a1243b680e"

Keep-Alive: timeout=3, max=100

Last-Modified: Sat, 10 Oct 2026 05:13:55 GMT

Feature-Policy: payment 'self'

Nel: {"report_to":"heroku-nel","response_headers":
["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}

Content-Type: text/html; charset=UTF-8
```

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr

```
Connection: Keep-Alive

X-Frame-Options: SAMEORIGIN

Date: Sat, 10 Oct 2026 05:30:27 GMT

Report-To: {"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=RhSrCTT
%2F2bPrLl6tC2HcugsZXgnjrvQ7DhT%2Bk0hyEVI%3D\u0026sid=812dcc77-0bd0-43b1-a5f1-b25750382959\
u0026ts=1791610227"}],"max_age":3600}

Access-Control-Allow-Origin: *

Accept-Ranges: bytes

Via: 1.1 heroku-router
```

13.2 Liens internes découverts

Aucun

14. Méthodologie et limites

La collecte passive (PowerShell) est non intrusive : HTTP/TLS, reconnaissance, inventaire des en-têtes. Les tests actifs (injections, XSS, contrôle d'accès…) ne sont réalisés que via OWASP ZAP (DAST) ; l'analyse de code via SonarQube (SAST) ; la corrélation de vulnérabilités via la base NVD ; le tout dans un cadre explicitement autorisé (numéro d'autorisation en section 1). Les cotations CVSS v3.1 sont indicatives et doivent être confirmées manuellement selon le contexte d'exploitation. Référentiels appliqués : OWASP Top 10 2021, OWASP ASVS, OWASP WSTG, CWE, NVD, recommandations de l'ANSSI.

Document SPÉCIMEN — exemple de rapport ASA AI WebAppSec — non contractuel — contact@asa-ai.fr