IA : des crawlers « affamés » qui saturent vos serveurs et rendent vos sites lents ou inaccessibles

IA : des crawlers « affamés » qui saturent vos serveurs et rendent vos sites lents ou inaccessibles

Bernard Sfez - 2025-10-28 17:14

Dans la course effrénée à la capture de données que se livrent les modèles d'IA, des armées de robots ratissent le Web sans relâche, souvent sans respecter les règles comme les fichiers robots.txt et autres. Bande passante siphonnée, serveurs sous pression, pages ralenties voire inaccessibles, une consommation anormale et extrême qui se fait au détriment des éditeurs, des entreprises et des utilisateurs. De plus en plus de voix, comme le souligne *The Register*, dénoncent des bots qui ne renvoient aucun trafic mais génèrent des coûts d'infrastructure et laissent des équipes débordées.

Chez OpenSource Solutions, nous privilégions une approche claire et mesurable. Nous analysons en continu les logs de trafic pour détecter surconsommations, nous croisons les signaux de plusieurs serveurs afin de distinguer les bons crawlers (moteurs de recherche) des scrapers d'IA agressifs. Enfin, nous mettons en place des mesures proactives (quotas, rate limiting, allowlists IP, pièges à bots, voir GEOip filtering) pour protéger les performances des sites sans empêcher l'indexation essentielle au référencement. Faisons donc le point.

Pourquoi les crawlers d'IA posent désormais problème

L'explosion des modèles d'IA a déclenché une collecte frénétique de contenu. Contrairement aux moteurs de recherche traditionnels — qui visent l'indexation et renvoient du trafic — beaucoup de robots liés à l'IA n'apportent aucun visiteur : ils aspirent massivement des pages, des API et même des assets médias pour entraîner des modèles, en externalisant le coût (bande passante, CPU, stockage, support) sur les éditeurs et les entreprises. Ces derniers jours, alertes et enquêtes se multiplient (The Register, Cloudflare, Ars Technica, Search Engine Land).

Au-delà du simple non-respect des directives de votre fichier robots.txt, ces crawlers « ne jouent pas le jeu » de multiples façons :

- User-agents opaques ou falsifiés : chaînes génériques ("Mozilla/5.0", "Python-requests…") ou usurpation d'un UA légitime.
- Masquage réseau : rotations d'IP, hébergeurs multiples, proxys, ASN variés pour contourner les blocages.

- Cadences excessives : rafales de requêtes parallèles, absence de backoff, ignorance des signaux de surcharge (429/Retry-After).
- Contournement des normes : non-respect des rate limits, du crawl-delay, des sitemaps et des zones d'exclusion.
- Techniques d'exploration agressives : découverte par dictionnaires/paramètres, scan de répertoires, suivi de liens transitoires, fetch d'APIs internes ou d'exports non destinés au public.
- Bypass des caches : variation d'en-têtes et de paramètres pour forcer des *miss* et multiplier le coût serveur.
- Automatisation "humaine" : navigateurs sans interface (headless), scripts qui exécutent du JS et simulent des interactions pour franchir des barrières simples.

Cette asymétrie crée des effets en chaîne avec des budgets bande passante et serveur dilapidé, pages plus lentes voire indisponibles pendant plusieurs jours, bruit analytique et équipes support submergées sans bénéfice business. À l'échelle, le problème n'est pas un simple manque de courtoisie robotique, c'est un risque opérationnel (SLO, coûts), marketing (dégradation SEO/UX) et parfois contractuel (conditions d'usage/API, droits sur les contenus).
Le défi n'est pas d'interdire tout robot, mais de distinguer l'exploration utile (celle qui sert votre visibilité dans les moteurs de recherche) de l'aspiration opportuniste, puis de réguler cette dernière.

Les symptômes et le diagnostic

Ce que vos visiteurs et vous-même pouvez remarquer

- Le site devient lent : les pages mettent plusieurs secondes, voire minutes, à s'afficher.
- Le site tombe par moments : messages d'erreur, page blanche, ou chargement infini.
- Actions qui "coincent" : panier d'achat qui ne s'ouvre pas, recherche qui ne répond plus, formulaire qui ne s'envoie pas.
- Pics d'indisponibilité aux "mauvaises" heures : tard le soir ou tôt le matin, alors que vous n'attendez pas de trafic.
- Coûts qui grimpent sans explication : facture d'hébergement/CDN soudainement plus élevée.
- Clients qui se plaignent : "C'est super lent", "j'ai une erreur", "je n'arrive pas à éditer mes pages".

Si vous vivez au moins deux de ces situations, il est possible que votre site soit "mangé" par des robots d'IA trop gourmands.

Pourquoi ça arrive

Des robots automatisés téléchargent vos pages bien plus souvent et plus vite qu'un humain. Résultat : ils occupent la ligne (bande passante) et surchargent la machine (serveur), ce qui ralentit tout le monde et peut couper l'accès.

Que faire tout de suite

Au minimum, vous devez effectuer ces vérifications:

- Essayez depuis un autre réseau (pas juste un autre station dans l'entreprise). À la maison, via votre mobile sans Wi-Fi (4G/5G) pour écarter un souci local.
- Mettez un message clair pour vos utilisateurs lorsque vous constatez des problèmes (“Nous investiguons, merci de votre patience”). Cela réduira les demandes support et donnera plus de temps aux techniciens.
- Vérifiez sur 24–48 h si les lenteurs/erreurs reviennent par vagues (et notez l’heure).
- Contactez-nous : Nous pouvons conduire les vérifications initiales à vos côtés — ou en totalité pour vous — et traiter les incidents, en maintenant l’accès des moteurs de recherche légitimes.

Contrôle plus technique du problème

À lire si vous avez des compétences techniques et un accès à votre hébergeur/CDN, à des métriques (ex. Zabbix, Matomo) et aux logs d’accès.

- Indices techniques typiques (dans vos métriques d’hébergement, outils de monitoring et logs)
- Pics de requêtes à des heures creuses, sans hausse des visiteurs réels.
- Rafales sur des URL profondes (archives, filtres, images lourdes).
- User-Agent générique ou changeant (“Mozilla/5.0”, scripts Python, outils d’aspiration).
- Rotation d’IP (hébergeurs cloud, pays variés), parfois plusieurs par minute.
- Cache contourné : mêmes pages redemandées avec des paramètres inutiles (?v=…, ?rand=…).
- Erreurs 429/5xx qui grimpent pendant les rafales, puis retombent.

Mini check-list (rapide)

- Top User-Agents et IP/AS sur les dernières 24–72 h.
- Taux de cache-miss et pages les plus coûteuses (images, JSON, recherche).
- Créneaux horaires des pics vs. trafic humain.
- Volume d’erreurs 429/5xx pendant les pics.

Si tout cela vous parle mais que ce n’est pas votre quotidien, pas de souci : nos équipes OpenSource Solution peuvent réaliser ces contrôles pour vous (diagnostic guidé ou clé en main).

Impacts concrets : technique, business, légal

Comprendre simplement

- Expérience client dégradée : pages qui traînent, paiements qui échouent, formulaires qui

“tournent” — et des visiteurs qui partent.
- Revenus en baisse : moins de conversions (ventes, demandes de contact), plus d’abandons de panier.
- Image de marque abîmée : avis négatifs, perception “site peu fiable”.
- Coûts qui explosent : factures d’hébergement/CDN et support client qui grimpent sans bénéfice.
- Risques de conformité : extraction non souhaitée de contenus, violation de conditions d’usage, exposition involontaire de données publiques mal cadrées.

Approche plus technique, pour quantifier l’impact

- Technique / SLO : hausse du TTFB, dégradation des Core Web Vitals, pics d’erreurs 5xx/429, saturation CPU/I/O, files de jobs encombrées.
- Business : baisse du taux de conversion, hausse du taux de rebond, sessions écourtées pendant les rafales ; sur e-commerce, impact direct sur le CA horaire.
- Financier : surcoût d'egress CDN, stockage (logs, caches), compute (autoscaling intempestif), temps support/N2/N3.
- Légal / Contractuel : non-respect de vos CGU/API, réutilisation de contenus sans autorisation, obligations RGPD si des endpoints exposent trop d’infos publiques.

Notre réponse avec l'implantation de différents éléments

Surveillance intelligente (Zabbix + Matomo)

- Seuils d’alerte pragmatiques (ex. TTFB > 1,2 s pendant 5 min, erreurs 5xx > 2 %, pics de requêtes sans hausse des visiteurs).
- Alertes multi-canaux (mail, SMS, etc.) priorisées selon l’impact business (panier, tunnel de commande, pages clés).
- Corrélation Zabbix ↔ Matomo : si la charge monte **sans** hausse du trafic humain, on **déclenche la défense anti-bots**.

Protection en temps réel (WAF + Fail2ban)

- Ralentissement ciblé (rate limiting) dès qu’un robot dépasse la cadence “normale”.
- Blocage automatique des IP/ASN suspects via Fail2ban (listes grises puis noires si récidive).
- Règles WAF dynamiques : on coupe les comportements typiques de scrapers (bypass cache, floods sur images lourdes, scan d’URL).
- Honeypages discrètes pour piéger les bots qui ignorent les règles et accélérer leur mise à l’écart.

Pilotage fin des bons crawlers (Virtualmin + WAF)

- Allowlist vérifiée (reverse DNS/IP officielles) pour Google, Bing, Applebot, etc.
- Quotas souples pour les moteurs légitimes (ils passent, mais ne saturent pas).

- Sitemaps & priorités : on guide l’indexation vers les pages utiles pour conserver/renforcer le SEO.

Opérations fluides (Virtualmin)

- Redémarrages ciblés et répartition de charge (sans couper le site).
- Cache et réglages PHP-FPM adaptés à votre saisonnalité.
- Playbooks d’incident : qui fait quoi, dans quel ordre, quand un pic arrive.

Les résultats attendus, ce que ça change pour vous : Des pages qui restent rapides, même en cas de pic robotique. Des factures maîtrisées (bande passante/egress évités), moins de tickets et de demandes d’assistance pour “site lent”, et plus de visites utiles ou de conversions aux moments clés. Le SEO est préservé : les bons moteurs continuent d’indexer ce qui compte.
Exemples d’alertes & d’actions automatiques

- Alerte : “Taux d’erreurs 5xx > 3 % pendant 3 min sur /search” → Action : limitation des IP fautives + cache forcé des réponses légères.
- Alerte : “+300 % requêtes sur images originales en 10 min” → Action : délai progressif et **waterfall block** sur ASN source.
- Alerte : “Pics de requêtes sans hausse Matomo (trafic humain)” → Action : Fail2ban passe en mode agressif et **WAF** durcit les règles de scraping.

Conclusion : un Web utile est un Web équitable

Le constat est désormais public : comme l’ont documenté plusieurs sources, des vagues de crawlers liés à l’IA provoquent des pics de trafic, dégradent les performances et font porter la facture aux éditeurs — sans retour de trafic ni de valeur.

Face à cette dérive, de grands acteurs d’infrastructure ont commencé à fermer le robinet par défaut aux crawlers d’IA non autorisés et à lancer des mécanismes de “Pay Per Crawl” pour redonner aux éditeurs un droit de regard, et le cas échéant, une rémunération lorsque des robots exploitent leur contenu. (Cloudflare, Wired, The Verge, Nieman Lab.)

Demain, il n’est pas exclu que ces pratiques s’inscrivent dans la loi : obligation d’identification claire des bots, consentement explicite pour certains usages d’IA, responsabilité en cas de dommages — voire redevances contractuelles pour l’accès machine à grande échelle. Ce débat avance à mesure que se multiplient les incidents, comme l’affaire Perplexity mise en cause pour un “stealth crawling” contournant les règles d’accès. (Business Insider)

Au-delà de la technique, c’est une question d’équilibre : laisser passer ce qui apporte de la valeur (référencement, découvertes, trafic), réguler ce qui prélève sans donner (aspiration opportuniste, charges abusives). Le “métier” de webmaster retrouve alors sa place de chef d’orchestre : publier de l’information utile, poser un cadre d’indexation clair, surveiller avec tact et dialoguer avec les robots légitimes. Notre rôle est d’organiser cet équilibre pour vous : protéger vos visiteurs, vos coûts et votre image — sans casser votre SEO.

Parlons-en : nous pouvons poser le premier diagnostic avec vous (ou pour vous) et mettre en place les bonnes protections, à votre rythme.