Directives Robots.txt : Un guide de toutes les règles standard et cachées

By rankstudio.net Publié le 17 octobre 2025

Résumé

Le fichier `robots.txt` (Protocole d'Exclusion des Robots, REP) est un mécanisme textuel de longue date permettant aux webmasters d'indiquer aux robots d'exploration automatisés quelles parties d'un site peuvent ou non être accédées. Il a été proposé pour la première fois par Martijn Koster en juillet 1994 (Source: www.webdesignmuseum.org) et est depuis devenu un standard de facto. En 2022, il a été formellement standardisé sous le nom de RFC 9309 sur la voie des standards de l'IETF (Source: www.rfc-editor.org). Robots.txt n'est *pas* un mécanisme de contrôle d'accès ou de sécurité – c'est plutôt une « demande » volontaire aux robots d'exploration amicaux (moteurs de recherche et autres bots) concernant les préférences d'exploration (Source: www.rfc-editor.org). En 2021, environ **81,9 %** des sites web indexés possédaient un fichier `robots.txt` (Source: almanac.httparchive.org), ce qui témoigne de son omniprésence.

Ce rapport propose un examen approfondi de **toutes les directives et paramètres connus** dans `robots.txt`, y compris les plus obscurs et ceux spécifiques aux moteurs de recherche. Nous couvrons le noyau standard (par exemple, `User-agent`, `Disallow`, `Allow`) et la syntaxe des motifs (caractères génériques, `$` fin de ligne), ainsi que des extensions telles que les liens `Sitemap:`. Nous détaillons ensuite les directives non standard ou moins connues – par exemple, `Crawl-delay`, `Clean-param` et `Host` de Yandex, `Request-rate`/`Visit-time` de Seznam, et les règles historiques `noindex` – en précisant quels robots d'exploration majeurs les supportent. Tout au long du rapport, les affirmations sont étayées par la documentation officielle, l'analyse d'experts et des études de cas réelles. Par exemple, les directives de Google Search Central confirment que les règles robots telles que « noindex » (dans robots.txt) sont *non prises en charge* (Source: developers.google.com), et que les pages bloquées par robots peuvent *toujours être indexées* si elles sont liées depuis ailleurs (Source: searchengineland.com) (et même apparaître dans les résultats de Google avec des extraits (Source: searchengineland.com). Yandex documente ses fonctionnalités uniques telles que `Clean-param` (pour ignorer les paramètres de requête) (Source: yandex.com), tandis que les ingénieurs de Bing ont noté que Bing n'a jamais respecté une directive `noindex` dans robots.txt (Source: www.seroundtable.com).

Nous analysons également les tendances d'utilisation (par exemple, un **tableau** de support des directives des moteurs de recherche) et les incidents réels. Par exemple, une étude de cas SEO a rapporté comment des règles robots configurées par l'hébergeur (ajoutées à l'insu du webmaster) ont par inadvertance désautorisé des sections clés du site au fil du temps – les pages affectées ont lentement disparu de l'index de Google (Source: searchengineland.com). Du point de vue de la sécurité, les chercheurs avertissent qu'inclure des chemins sensibles (comme `/admin/`, `/backup/`, `/debug/`) dans robots.txt *désigne en fait une cible* pour les attaquants (Source: www.theregister.com) (Source: nemocyberworld.github.io). S'appuyant sur des données (par exemple, du HTTP Archive SEO Almanac 2021 (Source: almanac.httparchive.org) et des blogs de moteurs de recherche (Source: www.askapache.com) (Source: developers.google.com), le rapport conclut avec les implications pour les webmasters (bonnes pratiques, fiabilité de l'exploration) et les orientations futures (le potentiel d'étendre le REP avec de nouvelles règles de consensus).

Introduction et Contexte

Le Protocole d'Exclusion des Robots (REP) est la norme originale de contrôle des robots d'exploration sur le web. Il a commencé comme un simple fichier `/robots.txt` à la racine d'un site, lu par les robots d'indexation, indiquant quelles URL **interdire** ou **autoriser**. Martijn Koster a introduit l'idée pour la première fois sur la liste de diffusion www-talk du W3C en juillet 1994 (Source: www.webdesignmuseum.org); il en a eu besoin, de manière célèbre, après que son site ait été victime d'une attaque par déni de service (DOS) par un robot d'exploration agressif. Au cours des décennies suivantes, le REP est resté une norme de facto *informelle* utilisée par pratiquement tous les principaux moteurs de recherche. Malgré son ancienneté, le REP a persisté avec peu de changements : en 2025, il avait « à peine eu besoin d'évoluer » — Google note que la seule extension universellement prise en charge qu'il a acquise était la directive `Allow` (Source: developers.google.com).

En septembre 2022, l'IETF a formalisé ces pratiques sous la forme de la RFC 9309 (Source: blog.seznam.cz). Ce document codifie le langage REP et les règles de traitement sur la voie des standards Internet (Source: www.rfc-editor.org). La RFC reconnaît la spécification originale de Koster de 1994 (Source: www.rfc-editor.org) et clarifie comment les robots d'exploration doivent analyser et mettre en cache `robots.txt`, gérer les redirections, les erreurs et la lecture des règles **User-agent**, **Allow**, **Disallow** (Source: www.rfc-editor.org) (Source: www.rfc-editor.org). Il est important de noter que la RFC stipule explicitement que les directives robots ne

sont *pas* un schéma d'autorisation – si vous listez un chemin dans `robots.txt`, il est ouvertement explorable par tout humain ou bot malveillant, la sécurité réelle doit donc utiliser un contrôle d'accès approprié (par exemple, l'authentification HTTP) (Source: www.rfc-editor.org).

En pratique, les fichiers `robots.txt` utilisent une grammaire simple. Ils se composent d'un ou plusieurs « groupes » (blocs), chacun commençant par une ou plusieurs lignes `User-agent:`, suivies de règles correspondantes. Par exemple :

```
User-agent: *
Disallow: /private/
Allow: /private/special/
Sitemap: https://example.com/sitemap.xml
```

Cela signifie « pour tous les robots d'exploration, interdire `/private/` sauf autoriser `/private/special/`. De plus, voici notre sitemap. » Dans le **Groupe 1**, le chemin vide après `Disallow:` implique *tout autoriser*. Chaque règle est une paire clé-valeur séparée par deux points. La clé peut être `Allow:` ou `Disallow:`, suivie d'un motif de chemin d'URL. La syntaxe REP (selon la RFC 9309) spécifie que pour un groupe correspondant, la règle *la plus spécifique* (la plus longue correspondance de chemin) a la priorité, et en cas d'égalité, un `Allow` l'emporte sur un `Disallow` (Source: www.rfc-editor.org). Les règles sont sensibles à la casse dans le chemin d'URL – par exemple, `Disallow: /Example/` ne bloquera *pas* `/example/` (Source: searchengineland.com). Si aucune règle ne correspond à un robot d'exploration ou une URL donné, l'exploration est implicitement autorisée (la valeur par défaut est ouverte) (Source: www.rfc-editor.org) (Source: yandex.com).

Depuis les débuts d'Internet, le paradigme est que les robots d'exploration amicaux doivent **obéir** à ces directives. Google, Bing, Yandex et d'autres ont construit leurs bots pour respecter les règles standard et de nombreuses extensions courantes. Cependant, cette nature volontaire signifie que chaque robot d'exploration peut choisir les directives à prendre en charge. Comme nous le verrons, certaines directives (comme `Crawl-delay` ou `Clean-param`) ne sont respectées que par quelques moteurs, et d'autres (comme une ligne `Noindex` dans robots) sont ignorées par les principaux robots d'exploration (Source: developers.google.com) (Source: www.seroundtable.com). Les sections suivantes détaillent la syntaxe, le support officiel et de nombreux paramètres « cachés » ou moins connus utilisés dans les fichiers `robots.txt` actuels.

Syntaxe et Directives de Base de `robots.txt`

Directives Standard : `User-agent`, `Disallow`, `Allow`

Les directives fondamentales dans un fichier `robots.txt` sont **`User-agent`** (identifiant le robot d'exploration cible) et **`Disallow`** (bloquant les chemins). La RFC 9309 formalise cette syntaxe de base : chaque règle est soit `Allow:`, soit `Disallow:`, chacune suivie d'un motif de chemin (Source: www.rfc-editor.org). Un groupe de règles s'applique aux lignes `User-agent` précédentes (Source: www.rfc-editor.org). Par exemple :

> *`User-agent: Googlebot` `Disallow: /admin/` `Allow: /admin/help/`*

Cela indique à Googlebot qu'il ne doit pas explorer `/admin/` sauf qu'il peut explorer `/admin/help/`. Le mot-clé `*` est utilisé pour correspondre à tous les robots d'exploration (par exemple, `User-agent: *`) (Source: stackoverflow.com). Par convention, un `Disallow:` vide (sans chemin) signifie *tout autoriser* (aucune restriction). Si aucune règle ne correspond, le contenu est par défaut explorable (Source: www.rfc-editor.org).

Les robots d'exploration font correspondre la règle la plus longue : si plusieurs directives correspondent à une URL, celle avec la chaîne de chemin la plus longue l'emporte. Si un `Allow` et un `Disallow` ont exactement la même longueur, **`Allow` a la priorité** (Source: www.rfc-editor.org). La casse compte : la correspondance de chemin est sensible à la casse (Source: searchengineland.com). (En pratique, comme les noms de domaine peuvent être insensibles à la casse, la RFC conseille d'utiliser le punycode ou la conversion UTF-8, mais ces détails affectent principalement la localisation (Source: yandex.com).

La plupart des moteurs de recherche prennent en charge la directive `Allow` aujourd'hui (Source: visual-seo.com). Par exemple, la documentation webmaster de Yandex liste explicitement `Allow` aux côtés de `Disallow` comme une directive qui permet l'exploration (Source: yandex.com). Google utilise également les règles `Allow:` pour créer des exceptions dans un arbre `Disallow` (Source: visual-seo.com) (Source: www.askapache.com). (À l'origine, `Allow` était une extension non officielle de Google introduite dans les années 1990 ; elle est maintenant omniprésente.)

Une autre directive courante est `Sitemap:` qui indique au robot d'exploration où trouver les sitemaps XML du site. Bien qu'elle ne fasse pas partie du REP original, c'est « une norme supplémentaire » reconnue par Google, Bing, Yandex et d'autres (Source: www.askapache.com). Dans l'exemple ci-dessus, `Sitemap: https://example.com/sitemap.xml` pointe les robots d'exploration vers l'URL du sitemap. Google, en fait, encourage le placement des sitemaps dans robots.txt (et l'a rendu explorable via cette règle) (Source: www.askapache.com). Tous les principaux moteurs de recherche respectent la ligne `Sitemap:` par commodité **même si elle est en dehors de la grammaire formelle de la RFC** (Source: developers.google.com).

Motifs et Caractères Génériques

Les robots d'exploration modernes prennent également en charge la correspondance de motifs simples dans les chemins. Les plus utilisés sont le caractère générique `*` (correspond à n'importe quelle séquence de caractères) et l'ancre `$` (correspond à la fin de l'URL). Par exemple, `Disallow: /*.pdf$` signifie « interdire toutes les URL se terminant par `.pdf` ». Le REP de Google a longtemps autorisé `*` et `$` dans les motifs `Disallow` (son analyseur open-source et sa documentation prennent en charge cette syntaxe) (Source: visual-seo.com) (Source: www.askapache.com). Yandex accepte également ces caractères génériques. Selon l'aide de Baidu, son `Baiduspider` prend en charge à la fois « `*` et `$` » pour la correspondance d'URL (Source: www.baidu.com). En pratique, de nombreux sites exploitent cette capacité pour bloquer des types de fichiers entiers ou des paramètres de requête d'URL. (Par exemple, `Disallow: /*?*` bloquera toute URL contenant « ? ».) Un processus de correspondance détaillé s'applique : une fois qu'un robot d'exploration a collecté toutes les règles `Disallow` et `Allow` pour son user-agent, il trouve la règle avec le préfixe correspondant le plus spécifique (le plus long) ; si cette règle est `Allow`, l'URL peut être explorée, et si elle est `Disallow`, l'URL est bloquée (Source: www.rfc-editor.org).

Sensibilité à la Casse et Normalisation

Les directives correspondent aux chemins d'URL textuellement. Cela signifie que la casse compte et que la chaîne exacte doit correspondre à partir du premier caractère. Par exemple, une directive `Disallow: /Category/` bloquera les URL comme `/Category/Item1` mais *pas* `/category/item1` – la non-concordance entre minuscules et majuscules signifie que la deuxième URL n'est pas interceptée (Source: searchengineland.com). De même, l'analyseur robots décode les caractères encodés en pourcentage avant la correspondance (Source: www.rfc-editor.org). Notez, cependant, que bien que la correspondance de chemin dans les règles soit sensible à la casse, la plupart des robots d'exploration traitent les noms de `User-agent:` et de directives (`User-agent`, `Allow`, etc.) comme des mots-clés insensibles à la casse (Source: yandex.com). En résumé, les règles robots.txt suivent une correspondance de chaîne précise sur la partie chemin des URL, il faut donc tenir compte de la normalisation des URL et de la casse lors de l'écriture des règles.

Directives Étendues et Moins Connues

Au-delà de la grammaire de base, un certain nombre de **directives non standard** sont apparues. Celles-ci ne font pas partie du REP original de 1994, mais beaucoup sont prises en charge en pratique par des robots d'exploration particuliers.

- `Allow` : Nous avons déjà couvert `Allow:` comme une extension pour annuler les interdictions. Son support s'est développé jusqu'à ce qu'il soit effectivement une norme dans tous les principaux moteurs (Source: visual-seo.com). L'analyseur robots de Google garantit que les règles `Allow` sont respectées, et si une règle `Allow` et une règle `Disallow` correspondent toutes deux à une URL, Google utilise le chemin le plus long (souvent le `Allow` s'il est plus long) (Source: www.rfc-editor.org).

- `Crawl-delay` : Cette directive a été inventée pour réguler la vitesse d'exploration (quelques pages par seconde). Elle ne fait *pas* partie de la grammaire officielle du REP【26†】, mais certains moteurs l'utilisent. **Yandex** prend en charge `Crawl-delay` dans robots.txt, par exemple `Crawl-delay: 10` pour attendre 10 secondes entre les récupérations (Source: yandex.com). **Bing** respecte également `Crawl-delay`. Google **ne prend pas en charge** une directive crawl-delay dans `robots.txt` : comme l'a expliqué Matt Cutts de Google, de nombreux webmasters la configurent mal (par exemple, en la réglant à 100 000) ce qui entraîne une exploration pratiquement nulle (Source: www.askapache.com). Au lieu de cela, Google propose des contrôles de taux d'exploration dans la Search Console (et gère en interne l'exploration avec des paramètres de « charge d'hôte ») (Source: www.askapache.com). Ainsi, si vous écrivez `Crawl-delay` dans `robots.txt`, seuls Yandex, Bing (et peut-être certains robots d'exploration personnalisés) le prendront en compte, pas Google.

- `Host` **spécifique aux moteurs de recherche** : À l'origine, Yandex a introduit une directive `Host:` pour permettre aux webmasters de déclarer le domaine préféré du site parmi les miroirs. Par exemple, si un site est accessible à la fois sous `example.com` et `example.net`, Yandex prenait la première ligne `Host:` comme hôte canonique (Source: stackoverflow.com). En

pratique, seul Yandex reconnaissait `Host:`. Cependant, **depuis mars 2018, Yandex a abandonné le support de `Host:`**, conseillant plutôt d'utiliser des redirections (Source: robotstxt.ru). (Toutes les lignes `Host:` après la première sont ignorées (Source: stackoverflow.com).) Les autres moteurs de recherche ignorent entièrement cette directive.

- **`Clean-param` de Yandex** : Yandex prend en charge une directive inhabituelle `Clean-param: p0[&p1&p2...] [path]` pour canoniser les URL en supprimant les paramètres de requête non pertinents (Source: yandex.com). Par exemple, pour regrouper les paramètres de suivi, on pourrait écrire :

  ```
  User-agent: Yandex
  Clean-param: ref /some_dir/get_book.pl
  ```

  Ceci indique à Yandex que les URL de la forme `/some_dir/get_book.pl?ref=XYZ&book_id=123` doivent être traitées comme si seul `book_id=123` était pertinent, c'est-à-dire ignorer toutes les valeurs `ref=` (Source: yandex.com). Yandex n'indexera alors qu'une seule URL canonique (`get_book.pl?book_id=123`) au lieu de doublons. Cette directive est propre à Yandex (Google, Bing, etc. ne prennent pas en charge `Clean-param`), et sa syntaxe peut accepter plusieurs paramètres et même des caractères génériques de chemin (Source: yandex.com).

- **`Noindex` et `Nofollow` (dans robots.txt)** : Au début, certains propriétaires de sites (et même certaines discussions Google) envisageaient d'ajouter `Noindex: /somepage` dans robots.txt pour empêcher l'indexation. **Cependant**, Google a longtemps refusé d'honorer `Noindex` *dans robots.txt*. Dans un blog de 2019, Gary Illyes (Google) a découragé l'utilisation de toute règle `noindex`, `nofollow` ou `crawl-delay` dans robots.txt, déclarant : « nous retirons tout code qui gère les règles non prises en charge et non publiées (telles que `noindex`) » (Source: developers.google.com). En fait, Google dit explicitement qu'il **ne garantit pas** que le blocage d'une URL dans robots.txt la maintiendra hors des résultats de recherche (Source: developers.google.com) (Source: searchengineland.com). (Les pages interdites par robots peuvent toujours être classées si elles sont liées ailleurs ; Google peut simplement afficher un extrait de texte de remplacement.) De même, l'équipe de Bing a noté que la « directive `noindex` non documentée n'a jamais fonctionné pour Bing » (Source: www.seroundtable.com). En résumé, **aucun moteur de recherche moderne majeur ne prend en charge une règle `<Noindex>` dans robots.txt**. Pour supprimer des pages de Google, il faut utiliser une balise `<meta name="robots" content="noindex">` dans la page ou lui envoyer une réponse 404/410 (Source: developers.google.com) (Source: www.seroundtable.com). (Le seul « nofollow » pertinent pour robots.txt est `Disallow` , qui empêche le robot d'exploration de suivre les liens – mais cela n'arrête pas non plus l'indexation si d'autres sites y renvoient.)

- **Autres extensions proposées** : Au fil des ans, diverses autres directives ont été suggérées ou adoptées par des robots d'exploration de niche. Par exemple, le brouillon de Conman.org (années 2000) définissait `Request-rate:` et `Visit-time:` . SeznamBot (un moteur de recherche tchèque populaire) les implémente : par exemple, `Request-rate: 10/1m` limite l'exploration à 10 pages par minute, éventuellement avec des fenêtres horaires supplémentaires (comme `1500-0559`) (Source: blog.seznam.cz). Le `Visit-time` du brouillon pouvait suggérer des heures d'exploration préférées. Ceux-ci ne sont pas reconnus par Google, Bing, Yandex ou la plupart des sites en dehors de Seznam. Mat Cutts a plaisanté en disant que l'équipe d'IBM pourrait définir des lignes `unicorns: allowed` s'ils voulaient étendre le protocole (Source: developers.google.com). Le point clé est que **les robots d'exploration sont libres d'implémenter des directives propriétaires** – le protocole est extensible. Par exemple, l'analyseur open-source de Google a été présenté avec un gestionnaire `Sitemap:` pour valider le support de règles personnalisées (Source: developers.google.com). Le blog Google de 2025 « Future-proof REP » reconnaît explicitement que de telles règles personnalisées (comme `clean-param` , `crawl-delay`) sont en dehors du nouvel RFC mais toujours prises en charge par certains moteurs (bien que *pas* par Google Search pour ces spécifiques) (Source: developers.google.com).

Le tableau ci-dessous résume les directives robots.txt courantes (en haut) et moins courantes, et quels moteurs de recherche majeurs les prennent actuellement en charge :

DIRECTIVE	FONCTION	GOOGLE	BING	YANDEX	AUTRES (NOTABLES)
`User-agent:`	Spécifie le robot d'exploration cible (ou `*` pour tous).	✓	✓	✓	–
`Disallow:`	Bloque le préfixe de chemin spécifié.	✓	✓	✓	–
`Allow:`	Autorise explicitement un chemin (annule Disallow).	✓	✓	✓	–
`Sitemap:`	URL(s) des fichiers sitemap XML du site.	✓	✓	✓	–
`Crawl-delay:`	Secondes à attendre entre les récupérations (limitation).	Non (Source: www.askapache.com)	✓	✓	(Également pris en charge par Yandex, Archive.org et certains robots d'exploration) (Source: www.askapache.com) (Source: yandex.com)
`Host:`	(Yandex uniquement) Domaine préféré parmi les miroirs.	–	–	*Partiel* (pris en charge jusqu'en mars 2018) (Source: robotstxt.ru)	–
`Clean-param:`	(Yandex uniquement) Ignore les paramètres d'URL spécifiés.	–	–	✓	–
`Noindex:`	(Si cela fonctionnait) Bloque l'indexation (obsolète).	✗ (Source: developers.google.com)	✗ (Source: www.seroundtable.com)	¬ support (documenté)	–

DIRECTIVE	FONCTION	GOOGLE	BING	YANDEX	AUTRES (NOTABLES)
(caractères génériques `*`, `$`)	Correspondance de motifs pour les URL.	✓ (pris en charge)	✓ (pris en charge)	✓ (pris en charge)	Implémenté par Baidu, Yandex, etc (Source: www.baidu.com)
(Autres : `Auth-group:` etc)	Pas d'utilisation courante	–	–	–	(Voir les robots de niche)

Table : Directives `robots.txt` clés et leur prise en charge par les principaux moteurs de recherche. « ✓ » indique la prise en charge. Un tiret « – » signifie l'absence de prise en charge. Yandex acceptait historiquement `Host:` et `Clean-param:` ; Google/Bing ne le font pas. Google et Bing ignorent tous deux toute `Noindex:` dans robots.txt` (Source: developers.google.com) (Source: www.seroundtable.com). (Sources : documentation officielle de Google, Yandex, connaissances communautaires.)

Comportements spécifiques aux moteurs de recherche

Différents robots d'exploration interprètent les règles robots de manière légèrement différente. Cette section met en évidence les comportements des principaux moteurs de recherche (Google, Bing, Yandex, Baidu, etc.) et la façon dont ils traitent robots.txt.

- **Google (et Googlebot)** : Googlebot suit entièrement la partie « standard » du REP. Il reconnaît `User-agent`, `Disallow`, `Allow`, `Sitemap` et les caractères génériques. Google **n'implémente pas** `Crawl-delay` ou `Request-rate` ; il utilise plutôt des contrôles d'exploration centralisés. Google ignore également toutes les lignes non prises en charge comme `Noindex:` dans robots.txt (Source: developers.google.com). Il est important de noter que Google indexera toujours (sans contenu) les URL qui sont interdites. Comme le note un guide SEO, « Aucune URL n'est entièrement bloquée des moteurs de recherche si vous l'interdisez dans robots.txt » (Source: searchengineland.com). La documentation de Google indique également qu'il « ne garantit pas » que les pages interdites ne finiront pas par être indexées. En pratique, Google peut afficher une entrée de résultat pour une URL interdite (souvent étiquetée « Non vérifiée » ou sans extrait) s'il trouve des liens vers celle-ci (Source: searchengineland.com). Les nouvelles fonctionnalités de Google permettent même que les pages interdites soient citées dans les aperçus d'IA avec des extraits (Source: searchengineland.com). Après 2019, Google a formellement désactivé l'analyse de `noindex` dans robots (et de toute règle non publiée comme `nofollow`) (Source: developers.google.com), s'alignant sur la position de Bing (Source: www.seroundtable.com). En 2019, Google a publié le code source de son analyseur robots.txt et a publié un Internet-Draft (proposition pré-RFC) montrant comment de nouvelles règles pourraient être ajoutées (Source: developers.google.com). Le blog officiel de Google (« Future-proof REP ») note qu'en plus de 25 ans, le seul changement universellement adopté a été l'ajout de `Allow` (Source: developers.google.com) ; d'autres extensions (comme `sitemap:`) sont devenues courantes en dehors du RFC.

- **Bing et Yahoo** : Étant donné que Yahoo Search utilise désormais le robot d'exploration de Bing (« Bingbot »), leur utilisation de robots est identique. Bing prend en charge `User-agent`, `Disallow`, `Allow`, `Sitemap` et (officieusement) `Crawl-delay`. Bing *exige* que si vous spécifiez une section nommée pour `Bingbot:`, vous devez y répéter toutes les règles générales. Comme l'a rapporté SearchLand, « Si vous créez une section spécifiquement pour Bingbot, toutes les directives par défaut seront ignorées... Vous *devez* copier-coller les directives que vous souhaitez que Bingbot suive sous sa propre section » (Source: searchengineland.com). Le développeur senior de Bing, Frédéric Dubut, a confirmé qu'il n'avait jamais reconnu `noindex` dans robots.txt, de sorte que les pages doivent utiliser des balises meta ou des en-têtes pour être supprimées de l'index de Bing (Source: www.seroundtable.com). Sinon, le comportement de Bing est similaire à celui de Google : les pages interdites peuvent toujours être indexées si elles sont liées, et Bing réserve ses propres contrôles de mise en cache dans les Outils pour les webmasters.

- **Yandex** : Yandexbot respecte le REP standard ainsi que ses extensions propriétaires. Sa documentation liste `Allow`, `Disallow`, `Crawl-delay`, ainsi que `Sitemap` et `Clean-param` (Source: yandex.com). Yandex utilise `Clean-param:` pour optimiser l'exploration des URL dynamiques (voir l'exemple ci-dessus (Source: yandex.com). Son `Crawl-delay` est exprimé en secondes (par exemple, `Crawl-delay: 10`) (Source: yandex.com). Jusqu'en 2018, Yandex lisait également une directive `Host:` pour le domaine canonique, mais celle-ci est maintenant abandonnée (Source: robotstxt.ru). Notamment, Yandex traite les pages interdites de manière similaire à Google : elles peuvent toujours être indexées, mais Yandex ne peut pas voir leur contenu et ne peut donc pas respecter

les balises HTML `noindex` qu'elles contiennent. (Yandex avertit donc les webmasters d'utiliser la balise meta `noindex` au lieu de robots pour masquer le contenu (Source: yandex.com).) Comme Google, Yandex considère les règles robots comme sensibles à la casse.

- **Baidu (principal moteur de recherche chinois)** : Les robots de Baidu (par exemple, « Baiduspider ») prennent en charge `User-agent`, `Disallow`, `Allow` et `Sitemap`. Baidu prend explicitement en charge les motifs de caractères génériques `*` et de fin de ligne `$` (Source: www.baidu.com) (sa propre documentation indique « Baiduspider supports wildcard characters `*` and `$` »). Baidu n'a **pas** d'argument `Crawl-delay` public dans robots.txt ; au lieu de cela, les webmasters chinois ajustent le taux d'exploration via les Outils pour les webmasters de Baidu. Baidu note également que les pages bloquées par robots peuvent toujours apparaître dans les résultats de recherche via des liens provenant d'autres sites (Source: www.baidu.com) ; donc encore une fois, `Disallow` est une directive pour contrôler l'exploration, *pas* une méthode infaillible de non-indexation. En pratique, `<User-agent: Baiduspider>` apparaît dans environ 1,9 % des sites (selon Crawling Stats (Source: almanac.httparchive.org).

- **Autres robots d'exploration** : Il existe d'innombrables robots moins importants (Majestic `mj12bot`, Ahrefs, etc.) qui obéissent simplement au REP comme Google. Le rapport SEO de l'HTTP Archive 2021 a noté que les user-agents spécifiques les plus courants rencontrés (après Google, Bing, Baidu, Yandex) incluaient Majestic (`mj12bot`, 3,3 % sur ordinateur) et Ahrefs (`ahrefsbot`, 3,3 % sur ordinateur) (Source: almanac.httparchive.org). Aucun de ces robots n'introduit de nouvelles directives uniques au-delà de ce que font les principaux moteurs.

Tendances et statistiques des données

Pour comprendre l'utilisation réelle de `robots.txt`, nous pouvons nous appuyer sur les données d'exploration web. Selon le chapitre SEO 2021 de l'HTTP Archive (Source: almanac.httparchive.org), **81,9 % des sites web utilisent un fichier `robots.txt` sur leur domaine principal** (une légère augmentation par rapport à environ 72 % en 2019). Inversement, environ 16,5 % des sites n'ont *pas* de fichier robots.txt, auquel cas Google traite *toutes* les pages comme explorables (Source: almanac.httparchive.org). Les ~1,6 % restants ont soit renvoyé des erreurs, soit n'étaient pas accessibles. Il est important de noter que si la récupération d'un fichier `robots.txt` échoue avec une erreur HTTP 5xx (erreur de serveur), la politique de Google (selon la RFC 9309) est de considérer le site comme « inaccessible » et de suspendre temporairement l'exploration (Source: www.rfc-editor.org) (Source: almanac.httparchive.org). S'il échoue avec une erreur 4xx ou 403, Google peut considérer le fichier comme « indisponible » et autoriser l'exploration par défaut (Source: www.rfc-editor.org). En pratique, l'Archive a constaté qu'environ 0,3 % des sites renvoyaient des erreurs 403/5xx pour robots.txt, et l'équipe de Google a estimé que jusqu'à 5 % présentaient des erreurs 5xx transitoires et que 26 % étaient parfois inaccessibles (Source: almanac.httparchive.org). Même des problèmes temporaires avec robots.txt peuvent arrêter un robot d'exploration : lors d'une enquête, Google a déclaré qu'il cesserait d'explorer un site pendant un certain temps si son fichier robots.txt renvoyait des erreurs, car il « ne sait pas si une page donnée peut ou non être explorée » (Source: almanac.httparchive.org).

Concernant la taille des fichiers, la plupart des fichiers `robots.txt` sont assez petits (<100 KiB). L'analyse de l'HTTP Archive montre que seulement ~3 000 domaines ont dépassé 500 KiB – le maximum documenté par Google – ce qui signifie que sur ces fichiers extra-larges, toute règle au-delà de 500 KiB serait simplement ignorée (Source: almanac.httparchive.org). Outre la taille, il y a aussi des considérations d'encodage de fichier (la RFC 9309 exige l'UTF-8) et une limite de 500 Ko pour l'analyseur afin d'éviter la surcharge (Source: yandex.com) (Source: www.rfc-editor.org). Les fichiers très volumineux ou mal formés risquent donc de ne pas être entièrement analysés.

Une autre statistique utile : la fréquence à laquelle des user-agents spécifiques sont mentionnés. La Figure 8.6 de l'Almanach du Web montre que « Googlebot » apparaît dans environ 3,3 à 3,4 % des règles robots.txt, Bingbot dans ~2,5 à 3,4 %, Baiduspider ~1,9 %, Yandexbot ~0,5 % (Source: almanac.httparchive.org). (Ces pourcentages concernent tous les fichiers robots.txt explorés.) Cela indique que Google et Bing sont explicitement ciblés par quelques pour cent des sites, tandis que Majestic et Ahrefs apparaissent également (environ 3 % chacun). Cela fait écho à la pratique des outils SEO qui placent leurs propres instructions d'exploration.

Enfin, l'utilisation de directives étendues est relativement rare sur le web. Par exemple, comparez `Clean-param` de Yandex avec son utilisation générale : pratiquement **100 %** des règles robots.txt destinées à Yandex l'utilisent lorsqu'il est présent, mais il n'apparaît que sur quelques pour cent de tous les sites à l'échelle mondiale (puisque seuls les sites indexés par Yandex l'utiliseraient de toute façon). De même, très peu de sites listent `Host:` maintenant (depuis que Yandex l'a abandonné) ou `Request-rate` de Seznam. Ce rapport s'est concentré sur l'exhaustivité plutôt que sur la prévalence, nous couvrons donc entièrement même ces cas plus rares.

Études de cas et exemples concrets

Erreurs SEO : Une illustration classique de l'impact de robots.txt est l'étude de cas de Glenn Gabe sur Search Engine Land (Source: searchengineland.com). Un client a réalisé que des pages de catégories clés disparaissaient mystérieusement de Google. Après enquête, Gabe a trouvé deux coupables : (1) le fournisseur du CMS avait ajouté de nouvelles directives `robots.txt` de manière programmatique au fil du temps à l'insu du propriétaire du site, et (2) certaines interdictions utilisaient la mauvaise casse (par exemple, `/CATEGORY/` au lieu de `/Category/`). Comme la correspondance robots.txt est sensible à la casse, ces directives ont accidentellement bloqué des pages. Le résultat a été une « fuite lente » d'URL importantes de l'index de Google (Source: searchengineland.com). L'analyse de Gabe souligne le danger des moindres modifications de robots.txt. Elle souligne que les webmasters devraient surveiller les modifications de robots.txt (certains utilisent des alertes ou le contrôle de version) et vérifier régulièrement quelles URL importantes pourraient être bloquées (des outils comme Screaming Frog ou le testeur robots.txt de la Search Console peuvent aider) (Source: searchengineland.com) (Source: searchengineland.com). L'utilisation de la Wayback Machine de l'Internet Archive pour vérifier les versions historiques de robots.txt peut également identifier le moment où une directive nuisible a été ajoutée (Source: searchengineland.com).

Risque de sécurité/Pot de miel : Au-delà du SEO, les fichiers robots.txt ont attiré l'attention des chercheurs en sécurité et même des hackers. Un testeur d'intrusion a analysé en 2015 des centaines de milliers de fichiers `robots.txt` sur le web et a constaté qu'ils exposent souvent des « cartes au trésor » aux attaquants (Source: www.theregister.com). Si un fichier robots.txt interdit des répertoires comme `/admin/`, `/staging/` ou `/backup/`, il **annonce** essentiellement l'existence de ces zones sensibles. Par exemple, Weksteen (un chercheur en sécurité) a signalé avoir trouvé de nombreux portails d'administration et de connexion simplement en explorant les chemins interdits dans robots.txt (Source: www.theregister.com). Ses découvertes incluent des cas réels : des milliers de sites gouvernementaux et universitaires avaient des entrées « /disallow » pointant vers des archives PDF confidentielles et des données personnelles, auxquelles les attaquants ont ensuite accédé via la recherche. Comme le résume The Register, *« la mention d'un répertoire dans un fichier robots.txt crie que le propriétaire a quelque chose à cacher »* (Source: www.theregister.com). Même les sites bien connus ne sont pas immunisés : il cite des cas où les noms de dossiers de victimes de traite ont été involontairement exposés via des descriptions d'images dans des listes d'interdiction.

De même, les experts en sécurité conseillent largement : **ne vous fiez pas à robots.txt pour protéger le contenu secret**. Les guides de hacking éthique soulignent que la publication de noms de fichiers ou de répertoires sensibles dans robots.txt est contre-productive ; elle crée une « surface d'attaque involontaire » (Source: nemocyberworld.github.io). En fait, certains administrateurs mettent en place des pots de miel en listant de faux chemins interdits et alléchants (par exemple, `/admin/please_dont_hack/`) et en surveillant ensuite toute tentative d'accès à ces chemins. En fin de compte, robots.txt est public : tout humain et tout bot malveillant peut le lire. Une section restreignant un chemin signifie que ce chemin existe et est important ; les attaquants sonderont en conséquence (Source: nemocyberworld.github.io) (Source: www.theregister.com).

Blocage vs Indexation : Une autre préoccupation pratique concerne les différents comportements de « bloqué » versus « indexé ». La Search Console a introduit de nouveaux messages d'état comme « Indexée, bien que bloquée par robots.txt », ce qui déroute de nombreux professionnels du SEO. Comme l'explique un article de SearchEngineLand de février 2025 (Source: searchengineland.com), « Bloquée par robots.txt » ne signifie *pas* « n'apparaîtra jamais dans les résultats de recherche ». Google déclare explicitement qu'une page interdite *peut toujours être indexée* (souvent en utilisant son URL et le texte des liens externes), même si Googlebot n'en récupérera pas le contenu (Source: searchengineland.com). En fait, les pages peuvent même apparaître dans des fonctionnalités spéciales ; Lily Ray a observé une URL Goodreads listée dans les aperçus IA de Google bien qu'elle soit bloquée par robots.txt (Source: searchengineland.com). Le consensus de la communauté se résume ainsi : **« Aucune URL n'est entièrement bloquée des moteurs de recherche si vous l'interdisez dans robots.txt »** (Source: searchengineland.com). Corriger une interdiction accidentelle implique généralement de supprimer la directive et de demander une nouvelle indexation via des outils (ou simplement d'attendre une nouvelle exploration) (Source: searchengineland.com).

Cas : Opérations d'exploration à grande échelle : Des projets publics comme l'Internet Archive s'appuient sur robots.txt pour respecter les exclusions de sites. Les robots d'exploration de l'Archive analysent `robots.txt` et respectent `Disallow` (comme le font pratiquement tous les bons robots). Cependant, différentes organisations ont choisi d'interpréter certains codes d'état différemment. Par exemple, les notes d'ingénierie de l'Internet Archive indiquent que par défaut, un fichier `robots.txt` manquant est traité comme « tout autoriser », tandis qu'un certain schéma de redirections ou de codes 401/403 pourrait être traité comme « autorisation complète » (c'est-à-dire, indexable) (Source: www.rfc-editor.org). Google, en revanche, traite les codes 401/403 différemment (il les considère comme analysables comme « tout autoriser ») (Source: www.rfc-editor.org). De telles nuances impliquent que les résultats de l'exploration peuvent varier légèrement entre les institutions.

Analyse et discussion

Profondeur des extensions vs. utilisation pratique. Au cours des plus de 25 ans d'existence de robots.txt, très peu de nouvelles règles ont atteint l'adoption universelle des directives fondamentales. Les ingénieurs de Google notent qu'à part `Allow`, la seule autre « extension » que presque tous les principaux robots comprennent est `Sitemap:` (Source: developers.google.com). Toutes les autres fonctionnalités restent soit spécifiques aux moteurs, soit facultatives. Par exemple, Google et Yandex ignorent discrètement toute directive `noindex` ou `nofollow` placée dans robots.txt (Source: developers.google.com) (Source: www.seroundtable.com) – de telles lignes n'ont tout simplement aucun effet. De même, bien que `Crawl-delay` soit largement reconnu par Bing et Yandex, Google choisit intentionnellement de ne pas le prendre en charge (Source: www.askapache.com).

Certaines extensions proposées par le passé subsistent dans des usages de niche. Les directives « Request-rate » et « Visit-time » de Seznam montrent qu'il est possible de planifier des calendriers d'exploration complexes si le robot et le webmaster sont d'accord. L'équipe de robotique de Google encourage la contribution de la communauté : l'article « REP à l'épreuve du futur » invite explicitement les webmasters à proposer de nouvelles directives (avec consensus) via des canaux ouverts (Source: developers.google.com) (Source: developers.google.com). Cela nous rappelle que la simplicité et l'omniprésence de robots.txt en font un candidat pour de nouvelles règles, mais seulement si elles sont largement bénéfiques. L'histoire de l'adoption de `sitemap:` comme règle (autrefois issue d'une collaboration entre SEOs et moteurs de recherche) est présentée comme un modèle (Source: developers.google.com). Inversement, ils avertissent que les changements unilatéraux ne deviendront pas la norme – la collaboration est nécessaire.

Implications pour les webmasters et le SEO. Pour les praticiens, les « secrets » de robots.txt consistent principalement à comprendre le comportement de chaque robot et à tester correctement. Partez toujours du principe que Google (et Bing) ignorera toute signification privée ou cachée dans votre fichier robots.txt. Ne mettez jamais de vrais mots de passe, de clés ou de points d'accès hautement secrets dans `robots.txt`. Utilisez-le uniquement pour bloquer les chemins d'exploration de faible valeur (pages dupliquées, zones de staging/test, etc.), et non pour masquer du contenu. Testez vos règles dans des outils : le testeur robots.txt de Google Search Console (si vérifié) et les validateurs tiers, pour vous assurer que la syntaxe est correcte. Surveillez les changements d'utilisation de votre robots.txt (la Wayback Machine ou des alertes) – comme le montre le cas Gabe, des changements inattendus peuvent discrètement ruiner le SEO. Gardez le fichier léger pour éviter les limites de taille ; compressez plusieurs interdictions en une seule spécification de chemin lorsque c'est possible.

Perspectives d'avenir. Avec la REP désormais officiellement standardisée, la plupart des « lacunes du protocole » sont connues. Les robots d'exploration montrent un intérêt pour l'évolution de robots.txt (par exemple, le brouillon de l'IETF, les analyseurs open-source), mais tout changement sera lent étant donné la nécessité d'une compatibilité ascendante et d'un large soutien. La perspective de Google pour 2025 est que robots.txt pourrait véhiculer de *nouvelles préférences d'exploration*, mais seulement par un consensus communautaire prudent (Source: developers.google.com). À mesure que l'IA et de nouvelles modalités de recherche émergent, contrôler ce qu'un robot peut voir reste crucial (robots.txt est la première ligne de communication). Pourtant, ironiquement, les spécifications soulignent que le contrôle sensible devrait être déplacé vers des mécanismes plus sécurisés (par exemple, les balises meta, les configurations de serveur) (Source: www.rfc-editor.org). La REP restera probablement un élément important, bien que limité, de l'écosystème d'indexation. Les indices de l'avenir incluent une meilleure analyse (Google a mis son analyseur en open source (Source: developers.google.com) et potentiellement de nouvelles directives flexibles – mais pour l'instant, les webmasters devraient maîtriser celles qui existent, sachant qu'il n'y a « pas d'autres secrets concernant robots.txt » au-delà de ces règles (Source: www.askapache.com).

Tableaux

MOTEUR DE RECHERCHE / ROBOT	PREND EN CHARGE `ALLOW`	PREND EN CHARGE LES CARACTÈRES GÉNÉRIQUES (`*`, `$`)	PREND EN CHARGE `CRAWL-DELAY`	PREND EN CHARGE `CLEAN-PARAM`	PREND EN CHARGE `HOST`	NOTES SUR `NOINDEX`
To understand real-world usage of `robots.txt`, we can draw on web crawl data. According to the 2021 HTTP Archive SEO chapter (Source: almanac.httparchive.org), **81.9% of websites use a `robots.txt` file on their main domain** (a slight increase from ~72% in 2019). Conversely, about 16.5% of sites have *no* robots.txt, in which case Google treats *all* pages as crawlable (Source: almanac.httparchive.org). The remaining ~1.6% either returned errors or were not reachable. Importantly, if a `robots.txt` fetch fails with HTTP 5xx (server error), Google's policy (per RFC 9309) is to treat the site as "unreachable" and temporarily suspend crawling (Source: www.rfc-editor.org) (Source: almanac.httparchive.org). If it fails with a 4xx or 403, Google may treat the file as "unavailable" and default to allowing crawling (Source: www.rfc-editor.org). In practice, the Archive found ~0.3% of sites returned 403/5xx for robots.txt, and Google's team estimated up to 5% had transient 5xx and 26% were unreachable at times (Source: almanac.httparchive.org). Even temporary issues with robots.txt can halt a crawler: in one survey Google said it will stop crawling a site for a while if its robots.txt returns errors, since it "is unsure if a given page can or cannot be crawled" (Source: almanac.httparchive.org).						

Regarding file size, most `robots.txt` are quite small (<100 KiB). The HTTP Archive analysis shows only ~3,000 domains exceeded 500 KiB – Google's documented maximum – meaning on those extra-large files any rules beyond 500 KiB would simply be ignored (Source: almanac.httparchive.org). Besides size, there are also file encoding considerations (RFC 9309 requires UTF-8) and a 500 KB parser limit to avoid overload (Source: yandex.com) (Source: www.rfc-editor.org). Very large or malformed files thus risk not being parsed fully.

Another useful statistic: how often specific user-agents are mentioned. Figure 8.6 of the Web Almanac shows that "Googlebot" appears in about 3.3–3.4% of robots.txt rules, Bingbot in ~2.5–3.4%, Baiduspider ~1.9%, Yandexbot ~0.5% (Source: almanac.httparchive.org). (These percentages are of all robots.txt files crawled.) That indicates Google and Bing are explicitly targeted by a few percent of sites, whereas Majestic and Ahrefs also appear (~3% each). This echoes the practice of SEO tools placing their own crawl instructions.

Finally, usage of extended directives is relatively rare on the web. For example, contrast Yandex's `Clean-param` with broad usage: virtually **100%** of Yandex-directed robots rules use it when present, but it appears on only a few percent of all sites globally (since only sites indexed by Yandex would use it at all). Similarly, very few sites list `Host:` now (since Yandex dropped it) or Seznam's `Request-rate`. This report has focused on comprehensiveness rather than prevalence, so we cover even these rarer cases fully.

Études de cas et exemples concrets

Erreurs SEO : Une illustration classique de l'impact de robots.txt est l'étude de cas de Glenn Gabe sur Search Engine Land (Source: searchengineland.com). Un client a réalisé que des pages de catégories clés disparaissaient mystérieusement de Google. Après enquête, Gabe a trouvé deux coupables : (1) le fournisseur du CMS avait ajouté de nouvelles directives `robots.txt` de manière programmatique au fil du temps à l'insu du propriétaire du site, et (2) certaines interdictions utilisaient la mauvaise casse (par exemple, `/CATEGORY/` au lieu de `/Category/`). Comme la correspondance robots.txt est sensible à la casse, ces directives ont accidentellement bloqué des pages. Le résultat a été une « fuite lente » d'URL importantes de l'index de Google (Source: searchengineland.com). L'analyse de Gabe souligne le danger des moindres modifications de robots.txt. Elle souligne que les webmasters devraient surveiller les modifications de robots.txt (certains utilisent des alertes ou le contrôle de version) et vérifier régulièrement quelles URL importantes pourraient être bloquées (des outils comme Screaming Frog ou le testeur robots.txt de la Search Console peuvent aider) (Source: searchengineland.com) (Source: searchengineland.com). L'utilisation de la Wayback Machine de l'Internet Archive pour vérifier les versions historiques de robots.txt peut également identifier le moment où une directive nuisible a été ajoutée (Source: searchengineland.com).

Risque de sécurité/Pot de miel : Au-delà du SEO, les fichiers robots.txt ont attiré l'attention des chercheurs en sécurité et même des hackers. Un testeur d'intrusion a analysé en 2015 des centaines de milliers de fichiers `robots.txt` sur le web et a constaté qu'ils exposent souvent des « cartes au trésor » aux attaquants (Source: www.theregister.com). Si un fichier robots.txt interdit des répertoires comme `/admin/`, `/staging/` ou `/backup/`, il **annonce** essentiellement l'existence de ces zones sensibles. Par exemple, Weksteen (un chercheur en sécurité) a signalé avoir trouvé de nombreux portails d'administration et de connexion simplement en explorant les chemins interdits dans robots.txt (Source: www.theregister.com). Ses découvertes incluent des cas réels : des milliers de sites gouvernementaux et universitaires avaient des entrées « /disallow » pointant vers des archives PDF confidentielles et des données personnelles, auxquelles les attaquants ont ensuite accédé via la recherche. Comme le résume The Register, *« la mention d'un répertoire dans un fichier robots.txt crie que le propriétaire a quelque chose à cacher »* (Source: www.theregister.com). Même les sites bien connus ne sont pas immunisés : il cite des cas où les noms de dossiers de victimes de traite ont été involontairement exposés via des descriptions d'images dans des listes d'interdiction.

De même, les experts en sécurité conseillent largement : **ne vous fiez pas à robots.txt pour protéger le contenu secret**. Les guides de hacking éthique soulignent que la publication de noms de fichiers ou de répertoires sensibles dans robots.txt est contre-productive ; elle crée une « surface d'attaque involontaire » (Source: nemocyberworld.github.io). En fait, certains administrateurs mettent en place des pots de miel en listant de faux chemins interdits et alléchants (par exemple, `/admin/please_dont_hack/`) et en surveillant ensuite toute tentative d'accès à ces chemins. En fin de compte, robots.txt est public : tout humain et tout bot malveillant peut le lire. Une section restreignant un chemin signifie que ce chemin existe et est important ; les attaquants sonderont en conséquence (Source: nemocyberworld.github.io) (Source: www.theregister.com).

Blocage vs Indexation : Une autre préoccupation pratique concerne les différents comportements de « bloqué » versus « indexé ». La Search Console a introduit de nouveaux messages d'état comme « Indexée, bien que bloquée par robots.txt », ce qui déroute de nombreux professionnels du SEO. Comme l'explique un article de SearchEngineLand de février 2025 (Source: searchengineland.com), « Bloquée par robots.txt » ne signifie *pas* « n'apparaîtra jamais dans les résultats de recherche ». Google déclare explicitement qu'une page interdite *peut toujours être indexée* (souvent en utilisant son URL et le texte des liens externes), même si Googlebot n'en récupérera pas le contenu (Source: searchengineland.com). En fait, les pages peuvent même apparaître dans des fonctionnalités spéciales ; Lily Ray a observé une URL Goodreads listée dans les aperçus IA de Google bien qu'elle soit bloquée par robots.txt (Source: searchengineland.com). Le consensus de la communauté se résume ainsi : **« Aucune URL n'est entièrement bloquée des moteurs de recherche si vous l'interdisez dans robots.txt »** (Source: searchengineland.com). Corriger une interdiction accidentelle implique généralement de supprimer la directive et de demander une nouvelle indexation via des outils (ou simplement d'attendre une nouvelle exploration) (Source: searchengineland.com).

Cas : Opérations d'exploration à grande échelle : Des projets publics comme l'Internet Archive s'appuient sur robots.txt pour respecter les exclusions de sites. Les robots d'exploration de l'Archive analysent `robots.txt` et respectent `Disallow` (comme le font pratiquement tous les bons robots). Cependant, différentes organisations ont choisi d'interpréter certains codes d'état différemment. Par exemple, les notes d'ingénierie de l'Internet Archive indiquent que par défaut, un fichier `robots.txt` manquant est traité comme « tout autoriser », tandis qu'un certain schéma de redirections ou de codes 401/403 pourrait être traité comme « autorisation complète » (c'est-à-dire, indexable) (Source: www.rfc-editor.org). Google, en revanche, traite les codes 401/403 différemment (il les considère comme analysables comme « tout autoriser ») (Source: www.rfc-editor.org). De telles nuances impliquent que les résultats de l'exploration peuvent varier légèrement entre les institutions.

Analyse et discussion

Profondeur des extensions vs. utilisation pratique. Au cours des plus de 25 ans d'existence de robots.txt, très peu de nouvelles règles ont atteint l'adoption universelle des directives fondamentales. Les ingénieurs de Google notent qu'à part `Allow`, la seule autre « extension » que presque tous les principaux robots comprennent est `Sitemap:` (Source: developers.google.com). Toutes les autres fonctionnalités restent soit spécifiques aux moteurs, soit facultatives. Par exemple, Google et Yandex ignorent discrètement toute directive `noindex` ou `nofollow` placée dans robots.txt (Source: developers.google.com) (Source: www.seroundtable.com) – de telles lignes n'ont tout simplement aucun effet. De même, bien que `Crawl-delay` soit largement reconnu par Bing et Yandex, Google choisit intentionnellement de ne pas le prendre en charge (Source: www.askapache.com).

Certaines extensions proposées par le passé subsistent dans des usages de niche. Les directives « Request-rate » et « Visit-time » de Seznam montrent qu'il est possible de planifier des calendriers d'exploration complexes si le robot et le webmaster sont d'accord. L'équipe de robotique de Google encourage la contribution de la communauté : l'article « REP à l'épreuve du futur » invite explicitement les webmasters à proposer de nouvelles directives (avec consensus) via des canaux ouverts (Source: developers.google.com) (Source: developers.google.com). Cela nous rappelle que la simplicité et l'omniprésence de robots.txt en font un candidat pour de nouvelles règles, mais seulement si elles sont largement bénéfiques. L'histoire de l'adoption de `sitemap:` comme règle (autrefois issue d'une collaboration entre SEOs et moteurs de recherche) est présentée comme un modèle (Source: developers.google.com). Inversement, ils avertissent que les changements unilatéraux ne deviendront pas la norme – la collaboration est nécessaire.

Implications pour les webmasters et le SEO. Pour les praticiens, les « secrets » de robots.txt consistent principalement à comprendre le comportement de chaque robot et à tester correctement. Partez toujours du principe que Google (et Bing) ignorera toute signification privée ou cachée dans votre fichier robots.txt. Ne mettez jamais de vrais mots de passe, de clés ou de points d'accès hautement secrets dans `robots.txt`. Utilisez-le uniquement pour bloquer les chemins d'exploration de faible valeur (pages dupliquées, zones de staging/test, etc.), et non pour masquer du contenu. Testez vos règles dans des outils : le testeur robots.txt de Google Search Console (si vérifié) et les validateurs tiers, pour vous assurer que la syntaxe est correcte. Surveillez les changements d'utilisation de votre robots.txt (la Wayback Machine ou des alertes) – comme le montre le cas Gabe, des changements inattendus peuvent discrètement ruiner le SEO. Gardez le fichier léger pour éviter les limites de taille ; compressez plusieurs interdictions en une seule spécification de chemin lorsque c'est possible.

Perspectives d'avenir. Avec la REP désormais officiellement standardisée, la plupart des « lacunes du protocole » sont connues. Les robots d'exploration montrent un intérêt pour l'évolution de robots.txt (par exemple, le brouillon de l'IETF, les analyseurs open-source), mais tout changement sera lent étant donné la nécessité d'une compatibilité ascendante et d'un large soutien. La perspective de Google pour 2025 est que robots.txt pourrait véhiculer de *nouvelles préférences d'exploration*, mais seulement par un consensus communautaire prudent (Source: developers.google.com). À mesure que l'IA et de nouvelles modalités de recherche émergent, contrôler ce qu'un robot peut voir reste crucial (robots.txt est la première ligne de communication). Pourtant, ironiquement, les spécifications soulignent que le contrôle sensible devrait être déplacé vers des mécanismes plus sécurisés (par exemple, les balises meta, les configurations de serveur) (Source: www.rfc-editor.org). La REP restera probablement un élément important, bien que limité, de l'écosystème d'indexation. Les indices de l'avenir incluent une meilleure analyse (Google a mis son analyseur en open source (Source: developers.google.com) et potentiellement de nouvelles directives flexibles – mais pour l'instant, les webmasters devraient maîtriser celles qui existent, sachant qu'il n'y a « pas d'autres secrets concernant robots.txt » au-delà de ces règles (Source: www.askapache.com).

Tableaux

MOTEUR DE RECHERCHE / ROBOT	PREND EN CHARGE `ALLOW`	PREND EN CHARGE LES CARACTÈRES GÉNÉRIQUES (`*`, `$`)	PREND EN CHARGE `CRAWL-DELAY`	PREND EN CHARGE `CLEAN-PARAM`	PREND EN CHARGE `HOST`	NOTES SUR `NOINDEX`
Pour comprendre l'utilisation réelle de `robots.txt`, nous pouvons nous appuyer sur les données d'exploration web. Selon le chapitre SEO 2021 de l'HTTP Archive (Source: almanac.httparchive.org), **81,9 % des sites web utilisent un fichier `robots.txt` sur leur domaine principal** (une légère augmentation par rapport à environ 72 % en 2019). Inversement, environ 16,5 % des sites n'ont *pas* de fichier robots.txt, auquel cas Google traite *toutes* les pages comme explorables (Source: almanac.httparchive.org). Les ~1,6 % restants ont soit renvoyé des erreurs, soit n'étaient pas accessibles. Il est important de noter que si la récupération d'un fichier `robots.txt` échoue avec une erreur HTTP 5xx (erreur de serveur), la politique de Google (selon la RFC 9309) est de considérer le site comme « inaccessible » et de suspendre temporairement l'exploration (Source: www.rfc-editor.org) (Source: almanac.httparchive.org). S'il échoue avec une erreur 4xx ou 403, Google peut considérer le fichier comme « indisponible » et autoriser l'exploration par défaut (Source: www.rfc-editor.org). En pratique, l'Archive a constaté qu'environ 0,3 % des sites renvoyaient des erreurs 403/5xx pour robots.txt, et l'équipe de Google a estimé que jusqu'à 5 % présentaient des erreurs 5xx transitoires et que 26 % étaient parfois inaccessibles (Source: almanac.httparchive.org). Même des problèmes temporaires avec robots.txt peuvent arrêter un robot d'exploration : lors d'une enquête, Google a déclaré qu'il cesserait d'explorer un site pendant un certain temps si son fichier robots.txt renvoyait des erreurs, car il « ne sait pas si une page donnée peut ou non être explorée » (Source: almanac.httparchive.org).						

Concernant la taille des fichiers, la plupart des fichiers `robots.txt` sont assez petits (<100 KiB). L'analyse de l'HTTP Archive montre que seulement ~3 000 domaines ont dépassé 500 KiB – le maximum documenté par Google – ce qui signifie que sur ces fichiers extra-larges, toute règle au-delà de 500 KiB serait simplement ignorée (Source: almanac.httparchive.org). Outre la taille, il y a aussi des considérations d'encodage de fichier (la RFC 9309 exige l'UTF-8) et une limite de 500 Ko pour l'analyseur afin d'éviter la surcharge (Source: yandex.com) (Source: www.rfc-editor.org). Les fichiers très volumineux ou mal formés risquent donc de ne pas être entièrement analysés.

Une autre statistique utile : la fréquence à laquelle des user-agents spécifiques sont mentionnés. La Figure 8.6 de l'Almanach du Web montre que « Googlebot » apparaît dans environ 3,3 à 3,4 % des règles robots.txt, Bingbot dans ~2,5 à 3,4 %, Baiduspider ~1,9 %, Yandexbot ~0,5 % (Source: almanac.httparchive.org). (Ces pourcentages concernent tous les fichiers robots.txt explorés.) Cela indique

que Google et Bing sont explicitement ciblés par quelques pour cent des sites, tandis que Majestic et Ahrefs apparaissent également (environ 3 % chacun). Cela fait écho à la pratique des outils SEO qui placent leurs propres instructions d'exploration.

Enfin, l'utilisation de directives étendues est relativement rare sur le web. Par exemple, comparez `Clean-param` de Yandex avec son utilisation générale : pratiquement **100 %** des règles robots.txt destinées à Yandex l'utilisent lorsqu'il est présent, mais il n'apparaît que sur quelques pour cent de tous les sites à l'échelle mondiale (puisque seuls les sites indexés par Yandex l'utiliseraient de toute façon). De même, très peu de sites listent `Host:` maintenant (depuis que Yandex l'a abandonné) ou `Request-rate` de Seznam. Ce rapport s'est concentré sur l'exhaustivité plutôt que sur la prévalence, nous couvrons donc entièrement même ces cas plus rares.

Études de cas et exemples concrets

Erreurs SEO : Une illustration classique de l'impact de robots.txt est l'étude de cas de Glenn Gabe sur Search Engine Land (Source: searchengineland.com). Un client a réalisé que des pages de catégories clés disparaissaient mystérieusement de Google. Après enquête, Gabe a trouvé deux coupables : (1) le fournisseur du CMS avait ajouté de nouvelles directives `robots.txt` de manière programmatique au fil du temps à l'insu du propriétaire du site, et (2) certaines interdictions utilisaient la mauvaise casse (par exemple, `/CATEGORY/` au lieu de `/Category/`). Comme la correspondance robots.txt est sensible à la casse, ces directives ont accidentellement bloqué des pages. Le résultat a été une « fuite lente » d'URL importantes de l'index de Google (Source: searchengineland.com). L'analyse de Gabe souligne le danger des moindres modifications de robots.txt. Elle souligne que les webmasters devraient surveiller les modifications de robots.txt (certains utilisent des alertes ou le contrôle de version) et vérifier régulièrement quelles URL importantes pourraient être bloquées (des outils comme Screaming Frog ou le testeur robots.txt de la Search Console peuvent aider) (Source: searchengineland.com) (Source: searchengineland.com). L'utilisation de la Wayback Machine de l'Internet Archive pour vérifier les versions historiques de robots.txt peut également identifier le moment où une directive nuisible a été ajoutée (Source: searchengineland.com).

Risque de sécurité/Pot de miel : Au-delà du SEO, les fichiers robots.txt ont attiré l'attention des chercheurs en sécurité et même des hackers. Un testeur d'intrusion a analysé en 2015 des centaines de milliers de fichiers `robots.txt` sur le web et a constaté qu'ils exposent souvent des « cartes au trésor » aux attaquants (Source: www.theregister.com). Si un fichier robots.txt interdit des répertoires comme `/admin/`, `/staging/` ou `/backup/`, il **annonce** essentiellement l'existence de ces zones sensibles. Par exemple, Weksteen (un chercheur en sécurité) a signalé avoir trouvé de nombreux portails d'administration et de connexion simplement en explorant les chemins interdits dans robots.txt (Source: www.theregister.com). Ses découvertes incluent des cas réels : des milliers de sites gouvernementaux et universitaires avaient des entrées « /disallow » pointant vers des archives PDF confidentielles et des données personnelles, auxquelles les attaquants ont ensuite accédé via la recherche. Comme le résume The Register, *« la mention d'un répertoire dans un fichier robots.txt crie que le propriétaire a quelque chose à cacher »* (Source: www.theregister.com). Même les sites bien connus ne sont pas immunisés : il cite des cas où les noms de dossiers de victimes de traite ont été involontairement exposés via des descriptions d'images dans des listes d'interdiction.

De même, les experts en sécurité conseillent largement : **ne vous fiez pas à robots.txt pour protéger le contenu secret**. Les guides de hacking éthique soulignent que la publication de noms de fichiers ou de répertoires sensibles dans robots.txt est contre-productive ; elle crée une « surface d'attaque involontaire » (Source: nemocyberworld.github.io). En fait, certains administrateurs mettent en place des pots de miel en listant de faux chemins interdits et alléchants (par exemple, `/admin/please_dont_hack/`) et en surveillant ensuite toute tentative d'accès à ces chemins. En fin de compte, robots.txt est public : tout humain et tout bot malveillant peut le lire. Une section restreignant un chemin signifie que ce chemin existe et est important ; les attaquants sonderont en conséquence (Source: nemocyberworld.github.io) (Source: www.theregister.com).

Blocage vs Indexation : Une autre préoccupation pratique concerne les différents comportements de « bloqué » versus « indexé ». La Search Console a introduit de nouveaux messages d'état comme « Indexée, bien que bloquée par robots.txt », ce qui déroute de nombreux professionnels du SEO. Comme l'explique un article de SearchEngineLand de février 2025 (Source: searchengineland.com), « Bloquée par robots.txt » ne signifie *pas* « n'apparaîtra jamais dans les résultats de recherche ». Google déclare explicitement qu'une page interdite *peut toujours être indexée* (souvent en utilisant son URL et le texte des liens externes), même si Googlebot n'en récupérera pas le contenu (Source: searchengineland.com). En fait, les pages peuvent même apparaître dans des fonctionnalités spéciales ; Lily Ray a observé une URL Goodreads listée dans les aperçus IA de Google bien qu'elle soit bloquée par robots.txt (Source: searchengineland.com). Le consensus de la communauté se résume ainsi : **« Aucune URL n'est entièrement bloquée des moteurs de recherche si vous l'interdisez dans robots.txt »** (Source: searchengineland.com). Corriger une interdiction accidentelle implique généralement de supprimer la directive et de demander une nouvelle indexation via des outils (ou simplement d'attendre une nouvelle exploration) (Source: searchengineland.com).

Cas : Opérations d'exploration à grande échelle : Des projets publics comme l'Internet Archive s'appuient sur robots.txt pour respecter les exclusions de sites. Les robots d'exploration de l'Archive analysent `robots.txt` et respectent `Disallow` (comme le font pratiquement tous les bons robots). Cependant, différentes organisations ont choisi d'interpréter certains codes d'état différemment. Par exemple, les notes d'ingénierie de l'Internet Archive indiquent que par défaut, un fichier `robots.txt` manquant est traité comme « tout autoriser », tandis qu'un certain schéma de redirections ou de codes 401/403 pourrait être traité comme « autorisation complète » (c'est-à-dire, indexable) (Source: www.rfc-editor.org). Google, en revanche, traite les codes 401/403 différemment (il les considère comme analysables comme « tout autoriser ») (Source: www.rfc-editor.org). De telles nuances impliquent que les résultats de l'exploration peuvent varier légèrement entre les institutions.

Analyse et discussion

Profondeur des extensions vs. utilisation pratique. Au cours des plus de 25 ans d'existence de robots.txt, très peu de nouvelles règles ont atteint l'adoption universelle des directives fondamentales. Les ingénieurs de Google notent qu'à part `Allow`, la seule autre « extension » que presque tous les principaux robots comprennent est `Sitemap:` (Source: developers.google.com). Toutes les autres fonctionnalités restent soit spécifiques aux moteurs, soit facultatives. Par exemple, Google et Yandex ignorent discrètement toute directive `noindex` ou `nofollow` placée dans robots.txt (Source: developers.google.com) (Source: www.seroundtable.com) – de telles lignes n'ont tout simplement aucun effet. De même, bien que `Crawl-delay` soit largement reconnu par Bing et Yandex, Google choisit intentionnellement de ne pas le prendre en charge (Source: www.askapache.com).

Certaines extensions proposées par le passé subsistent dans des usages de niche. Les directives « Request-rate » et « Visit-time » de Seznam montrent qu'il est possible de planifier des calendriers d'exploration complexes si le robot et le webmaster sont d'accord. L'équipe de robotique de Google encourage la contribution de la communauté : l'article « REP à l'épreuve du futur » invite explicitement les webmasters à proposer de nouvelles directives (avec consensus) via des canaux ouverts (Source: developers.google.com) (Source: developers.google.com). Cela nous rappelle que la simplicité et l'omniprésence de robots.txt en font un candidat pour de nouvelles règles, mais seulement si elles sont largement bénéfiques. L'histoire de l'adoption de `sitemap:` comme règle (autrefois issue d'une collaboration entre SEOs et moteurs de recherche) est présentée comme un modèle (Source: developers.google.com). Inversement, ils avertissent que les changements unilatéraux ne deviendront pas la norme – la collaboration est nécessaire.

Implications pour les webmasters et le SEO. Pour les praticiens, les « secrets » de robots.txt consistent principalement à comprendre le comportement de chaque robot et à tester correctement. Partez toujours du principe que Google (et Bing) ignorera toute signification privée ou cachée dans votre fichier robots.txt. Ne mettez jamais de vrais mots de passe, de clés ou de points d'accès hautement secrets dans `robots.txt`. Utilisez-le uniquement pour bloquer les chemins d'exploration de faible valeur (pages dupliquées, zones de staging/test, etc.), et non pour masquer du contenu. Testez vos règles dans des outils : le testeur robots.txt de Google Search Console (si vérifié) et les validateurs tiers, pour vous assurer que la syntaxe est correcte. Surveillez les changements d'utilisation de votre robots.txt (la Wayback Machine ou des alertes) – comme le montre le cas Gabe, des changements inattendus peuvent discrètement ruiner le SEO. Gardez le fichier léger pour éviter les limites de taille ; compressez plusieurs interdictions en une seule spécification de chemin lorsque c'est possible.

Perspectives d'avenir. Avec la REP désormais officiellement standardisée, la plupart des « lacunes du protocole » sont connues. Les robots d'exploration montrent un intérêt pour l'évolution de robots.txt (par exemple, le brouillon de l'IETF, les analyseurs open-source), mais tout changement sera lent étant donné la nécessité d'une compatibilité ascendante et d'un large soutien. La perspective de Google pour 2025 est que robots.txt pourrait véhiculer de *nouvelles préférences d'exploration*, mais seulement par un consensus communautaire prudent (Source: developers.google.com). À mesure que l'IA et de nouvelles modalités de recherche émergent, contrôler ce qu'un robot peut voir reste crucial (robots.txt est la première ligne de communication). Pourtant, ironiquement, les spécifications soulignent que le contrôle sensible devrait être déplacé vers des mécanismes plus sécurisés (par exemple, les balises meta, les configurations de serveur) (Source: www.rfc-editor.org). La REP restera probablement un élément important, bien que limité, de l'écosystème d'indexation. Les indices de l'avenir incluent une meilleure analyse (Google a mis son analyseur en open source (Source: developers.google.com) et potentiellement de nouvelles directives flexibles – mais pour l'instant, les webmasters devraient maîtriser celles qui existent, sachant qu'il n'y a « pas d'autres secrets concernant robots.txt » au-delà de ces règles (Source: www.askapache.com).

Tableaux

MOTEUR DE RECHERCHE / ROBOT	PREND EN CHARGE `ALLOW`	PREND EN CHARGE LES CARACTÈRES GÉNÉRIQUES (`*`, `$`)	PREND EN CHARGE `CRAWL-DELAY`	PREND EN CHARGE `CLEAN-PARAM`	PREND EN CHARGE `HOST`	NOTES SUR `NOINDEX`
Pour comprendre l'utilisation réelle de `robots.txt`, nous pouvons nous appuyer sur les données d'exploration web. Selon le chapitre SEO 2021 de l'HTTP Archive (Source: almanac.httparchive.org), **81,9 % des sites web utilisent un fichier `robots.txt` sur leur domaine principal** (une légère augmentation par rapport à environ 72 % en 2019). Inversement, environ 16,5 % des sites n'ont *pas* de fichier robots.txt, auquel cas Google traite *toutes* les pages comme explorables (Source: almanac.httparchive.org). Les ~1,6 % restants ont soit renvoyé des erreurs, soit n'étaient pas accessibles. Il est important de noter que si la récupération d'un fichier `robots.txt` échoue avec une erreur HTTP 5xx (erreur de serveur), la politique de Google (selon la RFC 9309) est de considérer le site comme « inaccessible » et de suspendre temporairement l'exploration (Source: www.rfc-editor.org) (Source: almanac.httparchive.org). S'il échoue avec une erreur 4xx ou 403, Google peut considérer le fichier comme « indisponible » et autoriser l'exploration par défaut (Source: www.rfc-editor.org). En pratique, l'Archive a constaté qu'environ 0,3 % des sites renvoyaient des erreurs 403/5xx pour robots.txt, et l'équipe de Google a estimé que jusqu'à 5 % présentaient des erreurs 5xx transitoires et que 26 % étaient parfois inaccessibles (Source: almanac.httparchive.org). Même des problèmes temporaires avec robots.txt peuvent arrêter un robot d'exploration : lors d'une enquête, Google a déclaré qu'il cesserait d'explorer un site pendant un certain temps si son fichier robots.txt renvoyait des erreurs, car il « ne sait pas si une page donnée peut ou non être explorée » (Source: almanac.httparchive.org).						

Concernant la taille des fichiers, la plupart des fichiers `robots.txt` sont assez petits (<100 KiB). L'analyse de l'HTTP Archive montre que seulement ~3 000 domaines ont dépassé 500 KiB – le maximum documenté par Google – ce qui signifie que sur ces fichiers extra-larges, toute règle au-delà de 500 KiB serait simplement ignorée (Source: almanac.httparchive.org). Outre la taille, il y a aussi des considérations d'encodage de fichier (la RFC 9309 exige l'UTF-8) et une limite de 500 Ko pour l'analyseur afin d'éviter la surcharge (Source: yandex.com) (Source: www.rfc-editor.org). Les fichiers très volumineux ou mal formés risquent donc de ne pas être entièrement analysés.

Une autre statistique utile : la fréquence à laquelle des user-agents spécifiques sont mentionnés. La Figure 8.6 de l'Almanach du Web montre que « Googlebot » apparaît dans environ 3,3 à 3,4 % des règles robots.txt, Bingbot dans ~2,5 à 3,4 %, Baiduspider ~1,9 %, Yandexbot ~0,5 % (Source: almanac.httparchive.org). (Ces pourcentages concernent tous les fichiers robots.txt explorés.) Cela indique

que Google et Bing sont explicitement ciblés par quelques pour cent des sites, tandis que Majestic et Ahrefs apparaissent également (environ 3 % chacun). Cela fait écho à la pratique des outils SEO qui placent leurs propres instructions d'exploration.

Enfin, l'utilisation de directives étendues est relativement rare sur le web. Par exemple, comparez `Clean-param` de Yandex avec son utilisation générale : pratiquement **100 %** des règles robots.txt destinées à Yandex l'utilisent lorsqu'il est présent, mais il n'apparaît que sur quelques pour cent de tous les sites à l'échelle mondiale (puisque seuls les sites indexés par Yandex l'utiliseraient de toute façon). De même, très peu de sites listent `Host:` maintenant (depuis que Yandex l'a abandonné) ou `Request-rate` de Seznam. Ce rapport s'est concentré sur l'exhaustivité plutôt que sur la prévalence, nous couvrons donc entièrement même ces cas plus rares.

Études de cas et exemples concrets

Erreurs SEO : Une illustration classique de l'impact de robots.txt est l'étude de cas de Glenn Gabe sur Search Engine Land (Source: searchengineland.com). Un client a réalisé que des pages de catégories clés disparaissaient mystérieusement de Google. Après enquête, Gabe a trouvé deux coupables : (1) le fournisseur du CMS avait ajouté de nouvelles directives `robots.txt` de manière programmatique au fil du temps à l'insu du propriétaire du site, et (2) certaines interdictions utilisaient la mauvaise casse (par exemple, `/CATEGORY/` au lieu de `/Category/`). Comme la correspondance robots.txt est sensible à la casse, ces directives ont accidentellement bloqué des pages. Le résultat a été une « fuite lente » d'URL importantes de l'index de Google (Source: searchengineland.com). L'analyse de Gabe souligne le danger des moindres modifications de robots.txt. Elle souligne que les webmasters devraient surveiller les modifications de robots.txt (certains utilisent des alertes ou le contrôle de version) et vérifier régulièrement quelles URL importantes pourraient être bloquées (des outils comme Screaming Frog ou le testeur robots.txt de la Search Console peuvent aider) (Source: searchengineland.com) (Source: searchengineland.com). L'utilisation de la Wayback Machine de l'Internet Archive pour vérifier les versions historiques de robots.txt peut également identifier le moment où une directive nuisible a été ajoutée (Source: searchengineland.com).

Risque de sécurité/Pot de miel : Au-delà du SEO, les fichiers robots.txt ont attiré l'attention des chercheurs en sécurité et même des hackers. Un testeur d'intrusion a analysé en 2015 des centaines de milliers de fichiers `robots.txt` sur le web et a constaté qu'ils exposent souvent des « cartes au trésor » aux attaquants (Source: www.theregister.com). Si un fichier robots.txt interdit des répertoires comme `/admin/`, `/staging/` ou `/backup/`, il **annonce** essentiellement l'existence de ces zones sensibles. Par exemple, Weksteen (un chercheur en sécurité) a signalé avoir trouvé de nombreux portails d'administration et de connexion simplement en explorant les chemins interdits dans robots.txt (Source: www.theregister.com). Ses découvertes incluent des cas réels : des milliers de sites gouvernementaux et universitaires avaient des entrées « /disallow » pointant vers des archives PDF confidentielles et des données personnelles, auxquelles les attaquants ont ensuite accédé via la recherche. Comme le résume The Register, *« la mention d'un répertoire dans un fichier robots.txt crie que le propriétaire a quelque chose à cacher »* (Source: www.theregister.com). Même les sites bien connus ne sont pas immunisés : il cite des cas où les noms de dossiers de victimes de traite ont été involontairement exposés via des descriptions d'images dans des listes d'interdiction.

De même, les experts en sécurité conseillent largement : **ne vous fiez pas à robots.txt pour protéger le contenu secret**. Les guides de hacking éthique soulignent que la publication de noms de fichiers ou de répertoires sensibles dans robots.txt est contre-productive ; elle crée une « surface d'attaque involontaire » (Source: nemocyberworld.github.io). En fait, certains administrateurs mettent en place des pots de miel en listant de faux chemins interdits et alléchants (par exemple, `/admin/please_dont_hack/`) et en surveillant ensuite toute tentative d'accès à ces chemins. En fin de compte, robots.txt est public : tout humain et tout bot malveillant peut le lire. Une section restreignant un chemin signifie que ce chemin existe et est important ; les attaquants sonderont en conséquence (Source: nemocyberworld.github.io) (Source: www.theregister.com).

Blocage vs Indexation : Une autre préoccupation pratique concerne les différents comportements de « bloqué » versus « indexé ». La Search Console a introduit de nouveaux messages d'état comme « Indexée, bien que bloquée par robots.txt », ce qui déroute de nombreux professionnels du SEO. Comme l'explique un article de SearchEngineLand de février 2025 (Source: searchengineland.com), « Bloquée par robots.txt » ne signifie *pas* « n'apparaîtra jamais dans les résultats de recherche ». Google déclare explicitement qu'une page interdite *peut toujours être indexée* (souvent en utilisant son URL et le texte des liens externes), même si Googlebot n'en récupérera pas le contenu (Source: searchengineland.com). En fait, les pages peuvent même apparaître dans des fonctionnalités spéciales ; Lily Ray a observé une URL Goodreads listée dans les aperçus IA de Google bien qu'elle soit bloquée par robots.txt (Source: searchengineland.com). Le consensus de la communauté se résume ainsi : **« Aucune URL n'est entièrement bloquée des moteurs de recherche si vous l'interdisez dans robots.txt »** (Source: searchengineland.com). Corriger une interdiction accidentelle implique généralement de supprimer la directive et de demander une nouvelle indexation via des outils (ou simplement d'attendre une nouvelle exploration) (Source: searchengineland.com).

Cas : Opérations d'exploration à grande échelle : Des projets publics comme l'Internet Archive s'appuient sur robots.txt pour respecter les exclusions de sites. Les robots d'exploration de l'Archive analysent `robots.txt` et respectent `Disallow` (comme le font pratiquement tous les bons robots). Cependant, différentes organisations ont choisi d'interpréter certains codes d'état différemment. Par exemple, les notes d'ingénierie de l'Internet Archive indiquent que par défaut, un fichier `robots.txt` manquant est traité comme « tout autoriser », tandis qu'un certain schéma de redirections ou de codes 401/403 pourrait être traité comme « autorisation complète » (c'est-à-dire, indexable) (Source: www.rfc-editor.org). Google, en revanche, traite les codes 401/403 différemment (il les considère comme analysables comme « tout autoriser ») (Source: www.rfc-editor.org). De telles nuances impliquent que les résultats de l'exploration peuvent varier légèrement entre les institutions.

Analyse et discussion

Profondeur des extensions vs. utilisation pratique. Au cours des plus de 25 ans d'existence de robots.txt, très peu de nouvelles règles ont atteint l'adoption universelle des directives fondamentales. Les ingénieurs de Google notent qu'à part `Allow`, la seule autre « extension » que presque tous les principaux robots comprennent est `Sitemap:` (Source: developers.google.com). Toutes les autres fonctionnalités restent soit spécifiques aux moteurs, soit facultatives. Par exemple, Google et Yandex ignorent discrètement toute directive `noindex` ou `nofollow` placée dans robots.txt (Source: developers.google.com) (Source: www.seroundtable.com) – de telles lignes n'ont tout simplement aucun effet. De même, bien que `Crawl-delay` soit largement reconnu par Bing et Yandex, Google choisit intentionnellement de ne pas le prendre en charge (Source: www.askapache.com).

Certaines extensions proposées par le passé subsistent dans des usages de niche. Les directives « Request-rate » et « Visit-time » de Seznam montrent qu'il est possible de planifier des calendriers d'exploration complexes si le robot et le webmaster sont d'accord. L'équipe de robotique de Google encourage la contribution de la communauté : l'article « REP à l'épreuve du futur » invite explicitement les webmasters à proposer de nouvelles directives (avec consensus) via des canaux ouverts (Source: developers.google.com) (Source: developers.google.com). Cela nous rappelle que la simplicité et l'omniprésence de robots.txt en font un candidat pour de nouvelles règles, mais seulement si elles sont largement bénéfiques. L'histoire de l'adoption de `sitemap:` comme règle (autrefois issue d'une collaboration entre SEOs et moteurs de recherche) est présentée comme un modèle (Source: developers.google.com). Inversement, ils avertissent que les changements unilatéraux ne deviendront pas la norme – la collaboration est nécessaire.

Implications pour les webmasters et le SEO. Pour les praticiens, les « secrets » de robots.txt consistent principalement à comprendre le comportement de chaque robot et à tester correctement. Partez toujours du principe que Google (et Bing) ignorera toute signification privée ou cachée dans votre fichier robots.txt. Ne mettez jamais de vrais mots de passe, de clés ou de points d'accès hautement secrets dans `robots.txt`. Utilisez-le uniquement pour bloquer les chemins d'exploration de faible valeur (pages dupliquées, zones de staging/test, etc.), et non pour masquer du contenu. Testez vos règles dans des outils : le testeur robots.txt de Google Search Console (si vérifié) et les validateurs tiers, pour vous assurer que la syntaxe est correcte. Surveillez les changements d'utilisation de votre robots.txt (la Wayback Machine ou des alertes) – comme le montre le cas Gabe, des changements inattendus peuvent discrètement ruiner le SEO. Gardez le fichier léger pour éviter les limites de taille ; compressez plusieurs interdictions en une seule spécification de chemin lorsque c'est possible.

Perspectives d'avenir. Avec la REP désormais officiellement standardisée, la plupart des « lacunes du protocole » sont connues. Les robots d'exploration montrent un intérêt pour l'évolution de robots.txt (par exemple, le brouillon de l'IETF, les analyseurs open-source), mais tout changement sera lent étant donné la nécessité d'une compatibilité ascendante et d'un large soutien. La perspective de Google pour 2025 est que robots.txt pourrait véhiculer de *nouvelles préférences d'exploration*, mais seulement par un consensus communautaire prudent (Source: developers.google.com). À mesure que l'IA et de nouvelles modalités de recherche émergent, contrôler ce qu'un robot peut voir reste crucial (robots.txt est la première ligne de communication). Pourtant, ironiquement, les spécifications soulignent que le contrôle sensible devrait être déplacé vers des mécanismes plus sécurisés (par exemple, les balises meta, les configurations de serveur) (Source: www.rfc-editor.org). La REP restera probablement un élément important, bien que limité, de l'écosystème d'indexation. Les indices de l'avenir incluent une meilleure analyse (Google a mis son analyseur en open source (Source: developers.google.com) et potentiellement de nouvelles directives flexibles – mais pour l'instant, les webmasters devraient maîtriser celles qui existent, sachant qu'il n'y a « pas d'autres secrets concernant robots.txt » au-delà de ces règles (Source: www.askapache.com).

Tableaux

MOTEUR DE RECHERCHE / ROBOT	PREND EN CHARGE `ALLOW`	PREND EN CHARGE LES CARACTÈRES GÉNÉRIQUES (`*`, `$`)	PREND EN CHARGE `CRAWL-DELAY`	PREND EN CHARGE `CLEAN-PARAM`	PREND EN CHARGE `HOST`	NOTES SUR `NOINDEX`

*Tableau : Prise en charge des fonctionnalités par les principaux crawlers de moteurs de recherche. Les coches proviennent des documents officiels et des blogs mentionnés ci-dessus. Google et Bing ne reconnaissent **pas** la directive `noindex` dans robots.txt (Source: developers.google.com) (Source: www.seroundtable.com) (utilisez plutôt les balises meta ou les en-têtes HTTP).*

USER-AGENT DANS `ROBOTS.TXT`	FRÉQUENCE DANS LES ÉTUDES DE CRAWL	CAS D'UTILISATION TYPIQUE
`User-agent: *` (tous les bots)	~100% (tous les sites avec robots.txt)	Règles par défaut pour tous les crawlers
`Googlebot` ou `Googlebot-News`	3.3–3.4% (Source: almanac.httparchive.org)	Règles explicites pour le crawler de Google
`Bingbot` ou `Slurp`	2.5–3.4% (Source: almanac.httparchive.org)	Règles explicites pour les crawlers de Bing/Yahoo
`Baiduspider`	~1.9% (Source: almanac.httparchive.org)	Règles spécifiques pour Baidu (moteur de recherche chinois)
`Yandexbot`	~0.5% (Source: almanac.httparchive.org)	Règles spécifiques pour Yandex (moteur de recherche russe)
`MJ12bot`, `AhrefsBot`, etc.	~3–4% chacun (par les outils SEO alternatifs)	Ciblé par les outils SEO pour guider leurs crawlers

Tableau : Répartition de la fréquence à laquelle des bots spécifiques sont nommés dans robots.txt (L'utilisation sur ordinateur de bureau et mobile est similaire) (Source: almanac.httparchive.org). Mis à part `*` (utilisé dans tous les fichiers), les bots de Google et Bing ne sont explicitement référencés que par quelques pour cent des sites. Il est à noter que certains bots d'outils SEO (Majestic, Ahrefs) apparaissent aussi fréquemment que ceux des grands moteurs de recherche.

Conclusion

Le fichier `robots.txt` peut sembler trivial, mais il recèle de nombreuses subtilités. Notre étude montre qu'outre les règles bien connues `User-agent` / `Disallow`, il existe un riche éventail de directives avec une adoption variable. Certains « secrets » de `robots.txt` ont trait à l'*absence* : par exemple, savoir que Google ne respectera pas `Crawl-delay` ou `Noindex` dans ce fichier (Source: www.askapache.com) (Source: developers.google.com). D'autres nuances impliquent des interactions et des cas limites : les URL interdites peuvent toujours être partiellement indexées (Source: searchengineland.com), ou les échecs de récupération de robots.txt peuvent **geler** le crawl jusqu'à ce qu'ils soient corrigés (Source: www.rfc-editor.org) (Source: almanac.httparchive.org). Nous avons également découvert des outils moins connus comme `Clean-param` de Yandex pour la fusion de chaînes de requête (Source: yandex.com) et les règles de limitation de débit de Seznam (Source: blog.seznam.cz). Chacun a sa place dans des écosystèmes spécifiques.

Historiquement, robots.txt s'est avéré remarquablement durable et extensible. Il n'a connu que des modifications mineures en 30 ans (par exemple, l'ajout de `Allow` et la prise en charge des caractères génériques) (Source: developers.google.com). Le RFC de 2022 a officiellement figé une grande partie de sa syntaxe, bien qu'il autorise de nouveaux enregistrements via des « autres enregistrements » (comme avec `Sitemap:`) (Source: www.rfc-editor.org). À l'avenir, les changements viendront très lentement, voire pas du tout. Google encourage les idées issues de la communauté (l'exemple d'une règle `sitemap:` existante montre comment le consensus peut favoriser l'adoption (Source: developers.google.com). Mais pour l'instant, les webmasters doivent être experts des règles et interprétations actuelles : une mauvaise utilisation peut nuire au SEO ou à la sécurité.

Recommandations : Les webmasters doivent maintenir `robots.txt` simple et bien testé. Ne listez que ce qui est véritablement non public ou de faible priorité (par exemple, les pages de connexion, les chemins de recherche en double). Vérifiez toujours la syntaxe (utilisez des outils ou le testeur de la Search Console) et rappelez-vous que le fichier lui-même est public. Consultez la documentation officielle de chaque moteur de recherche en cas de doute : pour Google, voir Google Search Central ; pour Yandex, voir les directives Yandex.Webmaster (Source: yandex.com) ; pour Baidu ou Bing, leurs centres d'aide. Lorsque vous dépannez des problèmes d'indexation, vérifiez toujours que vous n'avez pas accidentellement interdit des URL nécessaires (les outils courants incluent le rapport robots de la Google Search Console (Source: searchengineland.com) et l'historique des versions archivées (Source: searchengineland.com)).

En résumé, `robots.txt` reste un outil essentiel, bien que discret, pour le contrôle du crawl. Comprendre toute son étendue – jusqu'aux « secrets » des paramètres cachés ou uniques – permet aux propriétaires de sites de mieux gérer leur présence en ligne. Toutes les affirmations et recommandations ci-dessus sont étayées par des sources faisant autorité et des exemples concrets. Utilisez ce rapport comme une liste de contrôle de référence pour vous assurer que votre `robots.txt` est à la fois correct et sécurisé, et pour rester informé de tout développement futur du Protocole d'Exclusion des Robots (Source: developers.google.com) (Source: www.rfc-editor.org).

Étiquettes: robotstxt, seo, explorateurs-web, protocole-exclusion-robots, rfc-9309, crawl-delay, user-agent, directive-disallow, directive-sitemap

---

AVERTISSEMENT

Ce document est fourni à titre informatif uniquement. Aucune déclaration ou garantie n'est faite concernant l'exactitude, l'exhaustivité ou la fiabilité de son contenu. Toute utilisation de ces informations est à vos propres risques. Unknown ne sera pas responsable des dommages découlant de l'utilisation de ce document. Ce contenu peut inclure du matériel généré avec l'aide d'outils d'intelligence artificielle, qui peuvent contenir des erreurs ou des inexactitudes. Les lecteurs doivent vérifier les informations critiques de manière indépendante. Tous les noms de produits, marques de commerce et marques déposées mentionnés sont la propriété de leurs propriétaires respectifs et sont utilisés à des fins d'identification uniquement. L'utilisation de ces noms n'implique pas l'approbation. Ce document ne constitue pas un conseil professionnel ou juridique. Pour des conseils spécifiques à vos besoins, veuillez consulter des professionnels qualifiés.