Engineer

robots.txt

файл, що вказує роботам, куди не заходити

Що це таке

Будь-який хост у мережі може опублікувати за фіксованим шляхом `/robots.txt` простий текстовий файл із проханнями до програм, які його обходять. Мартейн Костер описав цю домовленість у 1994 році, а у вересні 2022 року IETF опублікувала її як RFC 9309, Robots Exclusion Protocol, з точним формулюванням правил та інструкціями щодо помилок і кешування.

Файл — це перелік груп. Група починається з одного чи кількох рядків `user-agent`, кожен із яких називає краулер за його токеном продукту, наприклад `ExampleBot`, або `*` для всіх краулерів. Під ними йдуть рядки `allow` і `disallow`, кожен зі шляхом. Краулер шукає групу, що називає його власний токен, без огляду на регістр літер, і виконує саме її; якщо його називають кілька груп, їхні правила зливаються в одну. Лише краулер без власної групи переходить до групи `*`, а якщо немає ні тієї, ні іншої, правил для нього немає.

Усередині групи правило діє, коли його шлях збігається з початком адреси, і перемагає найконкретніший збіг, тобто той, що має найбільше октетів. Коли `allow` і `disallow` рівні, перемагає `allow`, а адреса поза всіма правилами дозволена. Два символи розширюють шлях: `*` означає будь-яку послідовність символів, а `$` закріплює кінець адреси. `#` починає коментар, а власний шлях файлу дозволений завжди.

RFC також визначає, як файл отримують. Він лежить саме за цим шляхом, написаним малими літерами, на верхньому рівні сервісу, тож кожен хост відповідає сам за себе, і файл на одному піддомені нічого не каже про інший. Це простий текст у UTF-8. Краулерові рекомендовано підтримувати

ланцюжок щонайменше з п’яти переспрямувань, не повинен користуватися кешованою копією довше за 24 години і може обмежити обсяг, який розбирає, але не нижче за 500 КіБ.

Рядки поза протоколом допускаються. RFC називає їх іншими записами, наводить як приклад `Sitemap` і дозволяє краулеру тлумачити їх чи ні за однієї умови: такий рядок ніколи не повинен завершувати групу.

Застосуйте поняття до рішення

Застосуйте «Що таке robots.txt?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Сайт чи застосунок, який ШІ-асистенти ваших клієнтів можуть знайти, прочитати й використати, і це перевіряється з кожним випуском, а не обіцяється один раз.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Побудувала сайт на Hugo 3 мовами — 108 файлів шаблонів обслуговують три мови й чотири формати

Побудувала сайт на Hugo 3 мовами Зв'язатися з нами →

Чому це важливо

RFC прямо називає межу у вступі: правила не є формою контролю доступу. Файл — це прохання. Краулер, який його поважає, тримається осторонь, а той, що не поважає, читає все, що віддає сервер. Розділ про безпеку додає, що шлях, згаданий у файлі, тим самим стає публічним і його легко знайти, тож усе, що має лишатися приватним, потребує справжнього захисту, наприклад HTTP-автентифікації.

Документація Google звужує призначення ще більше. За нею, файл використовують передусім для того, щоб не перевантажувати сайт запитами, і він не є способом прибрати сторінку з Google. Заборонену адресу все одно можуть проіндексувати, якщо на неї посилаються інші сторінки, і тоді вона з’являється в результатах без опису, бо краулер так і не прочитав сторінку. Щоб тримати сторінку поза результатами, Google радить правило `noindex` або пароль, а правило `noindex` діє лише на сторінці, яку краулерові дозволено завантажити.

Добре файл робить ту роботу, для якої його створили: відводить краулери від частин сайту, які коштують серверу запитів і мало дають пошуковій системі, наприклад від неважливих чи майже однакових сторінок. Його

охоплення — це водночас і його ризик. Один рядок під `*` діє на кожну адресу й кожен краулер, що поважає файл, а зміна може набути чинності лише за добу, тож `Disallow: /`, вписаний не в ту групу, закриває весь сайт.

Як це виглядає на практиці

Файл цього сайту довгий, і майже весь він складається з коментарів. Частина, на яку краулер реагує, — це чотири рядки:

```
User-agent: *
Content-Signal: search=yes, ai-input=yes, ai-train=yes
Allow: /

Sitemap: https://engineer.company/sitemap.xml
```

Одна група охоплює всі краулери, а `Allow: /` збігається з кожною адресою. Група зовсім без правил дозволила б те саме, адже адреса поза всіма правилами дозволена, але рядок записано явно, бо відкритість — це вибір. Коментар над ним називає цю політику ухваленою, а не успадкованою, а сторінка про готовність до ШІ-агентів описує політику сканування сайту як одне з його свідомих рішень.

`Content-Signal` не належить до записів, які визначає RFC 9309. Він походить зі словника Content Signals і тут дає згоду на збирання для пошуку, для відповіді на запитання в ту мить, коли його ставлять, і для навчання. За RFC це один з інших записів: краулер, що знає цю домовленість, може його прочитати, старіший краулер пропускає рядок, як каже власний коментар файлу, і обидва мають вести групу далі після нього. Рядок `Sitemap:` — запис того самого роду, він вказує на індекс карт сайту. Сторінка подяк підсумовує файл так само: він дозволяє все, «і пошуковим краулерам, і ШІ-системам однаково», і називає карту сайту.

Те, чого у файлі немає, теж вирішено свідомо. Серед п’яти рекомендацій, від яких цей сайт відмовився, сторінка про готовність до ШІ-агентів називає окремі правила для кожного агента і лишає одне правило robots для всіх, «бо правило, написане для одного краулера, мовчки замінює для нього загальне». Якби файл дав одному ШІ-краулеру власну групу, той читав би лише її, і `Disallow`, доданий згодом під `*`, ніколи б до нього не дійшов.

Решта — коментарі для того, хто відкриє файл: обґрунтування політики, а далі адреси `llms.txt`, `agents.txt`, `humans.txt`, `security.txt` та інших машиночитних файлів, записані коментарями, бо robots.txt не має для них поля. Через це коментарі стають частиною того, що публікується. Рядок із `#` у шаблоні віддається кожному краулеру байт у байт, і двічі такий рядок ніс нотатку для супровідника, що вказувала на плановий документ у непублічному репозиторії. Тепер перевірка зібраного сайту падає, коли опублікований файл містить шлях усередину репозиторію, а нотатки для супровідників у шаблоні стоять у коментарях шаблону, які генератор вилучає.

Де виникають проблеми

Правило для всіх перестає діяти для одного краулера. Хтось додає групу для окремого бота, скажімо, щоб не пускати його в одну теку, і відтоді цей бот ігнорує все під `*`, включно з кожним `Disallow`, що там стоїть. RFC 9309 відсилає краулер до групи `*` лише тоді, коли для нього немає власної групи, а документація Google каже, що специфічні групи й група `*` не поєднуються. Повторіть кожне загальне правило в кожній іменованій групі або тримайтеся однієї групи, як цей сайт.

Заблокована сторінка все одно з'являється в результатах пошуку. Адреса показується без опису, можливо, зі словами, якими інші сайти підписали посилання на неї. Google так і не завантажив сторінку, тож знає лише, що адреса існує і як її називають інші. `Disallow` керує скануванням, а не індексуванням. Приберіть правило й віддавайте `noindex` на самій сторінці або закрийте сторінку паролем; `noindex`, схований за `Disallow`, ніхто ніколи не прочитає.

Файл рекламує те, що мав приховати. Рядок на кшталт `Disallow: /admin/` підказує кожному читачеві, чемному краулеру чи ні, де саме шукати, а файл публічний за своєю природою. Розділ RFC про безпеку каже, що шляхи у файлі стає легко знайти, і вимагає натомість контролю доступу. Тримайте приватне за автентифікацією і взагалі не згадуйте його у файлі.

Помилка сервера зупиняє сканування всього сайту. Сторінки відкриваються, але запит до `/robots.txt` повертає статус 5xx або перевищує час очікування, і краулер, що дотримується RFC, має вважати це повною забороною. Google припиняє сканування на перші 12 годин, а потім до 30 днів працює з останньою вдалою копією. Протилежна помилка тихіша: будь-

яку відповідь 4xx, крім 429, зокрема 401 і 403, Google вважає відсутністю файлу, тож кожна адреса відкрита. Віддавайте файл як статичний текст зі статусом 200 і стежте за самим `/robots.txt`, а не лише за головною сторінкою.

Що почитати далі

- RFC 9309: Robots Exclusion Protocol — сам стандарт: групи, зіставлення, обробка помилок, кешування й розділ про безпеку.
- Introduction to robots.txt — для чого, за словами Google, потрібен файл і чому він не ховає сторінку з результатів.
- How Google interprets the robots.txt specification — коди статусу, кешування, поля, які Google ігнорує, і як він обирає групу.
- Block Search indexing with noindex — правило, яке справді тримає сторінку поза результатами, і чому сторінка має бути доступна для сканування.



Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке robots.txt?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/robots-txt/