Engineer

robots.txt

filen der fortæller crawlere hvor de ikke må gå

Hvad det er

Enhver vært på nettet kan på den faste sti `/robots.txt` udgive en ren tekstfil med anmodninger til de programmer, der crawler den. Martijn Koster definerede konventionen i 1994, og i september 2022 udgav IETF den som RFC 9309, Robots Exclusion Protocol, med et præcist sprog for reglerne og instrukser for fejl og caching.

Filen er en liste af grupper. En gruppe begynder med en eller flere `user-agent`-linjer, der hver nævner en crawler ved dens produkttoken, for eksempel `ExampleBot`, eller `*` for alle crawlere. Under dem kommer `allow`- og `disallow`-linjer, hver med en sti. En crawler leder efter den gruppe, der nævner dens eget token, uden hensyn til store og små bogstaver, og følger den gruppe; nævner flere grupper den, smelter deres regler sammen til én. Kun en crawler uden egen gruppe falder tilbage på `*`-gruppen, og en, der hverken finder det ene eller det andet, har ingen regler at følge.

Inden for en gruppe gælder en regel, når dens sti passer på begyndelsen af adressen, og det mest specifikke match vinder, altså det med flest oktetter. Står en `allow` og en `disallow` lige, vinder `allow`, og en adresse, som ikke rammes af en regel, er tilladt. To tegn udvider en sti: `*` står for en vilkårlig række tegn, og `$` låser enden af adressen. Et `#` indleder en kommentar, og filens egen sti er altid tilladt.

RFC'en fastlægger også, hvordan filen hentes. Den ligger på netop den sti med små bogstaver øverst i tjenesten, så hver vært står for sig selv, og en fil på ét subdomæne siger intet om et andet. Den er UTF-8-kodet ren tekst. En crawler bør følge mindst fem omdirigeringer for at nå den, bør ikke bruge en cachet kopi i mere end 24 timer og må gerne sætte et loft over, hvor meget den læser, men aldrig under 500 KiB.

Linjer uden for protokollen tåles. RFC'en kalder dem andre poster, nævner `Sitemap` som eksempel og lader en crawler fortolke dem eller lade være på én betingelse: sådan en linje må aldrig afslutte en gruppe.

Brug begrebet i en beslutning

Brug »Hvad er robots.txt?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Et site eller en app, som de AI-assistenter, jeres kunder bruger, kan finde, læse og handle på, kontrolleret ved hver udgivelse i stedet for lovet én gang.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har bygget et Hugo-site på 3 sprog — 108 skabelonfiler, der betjener tre sprog og fire formater

Har bygget et Hugo-site på 3 sprog Kontakt os →

Hvorfor det betyder noget

RFC'en siger grænsen ligeud i sin indledning: reglerne er ikke en form for adgangskontrol. Filen er en anmodning. En crawler, der respekterer den, holder sig væk, og en, der ikke gør, læser alt, hvad serveren udleverer. Afsnittet om sikkerhed tilføjer, at en sti, der står i filen, dermed bliver offentlig og til at finde, så alt, der skal forblive privat, kræver en egentlig kontrol, for eksempel HTTP-godkendelse.

Googles dokumentation indsnævrer formålet yderligere. Ifølge den bruges filen først og fremmest til at undgå at overbelaste et site med forespørgsler, og den er ikke en måde at holde en side ude af Google på. En spærret adresse kan stadig blive indekseret, når andre sider linker til den, og vises så i resultaterne uden beskrivelse, fordi crawleren aldrig læste siden. For at holde en side ude af resultaterne henviser Google til en `noindex` -regel eller en adgangskode, og en `noindex` -regel virker kun på en side, crawleren må hente.

Det, filen gør godt, er det arbejde, den blev skrevet til: at holde crawlere væk fra dele af et site, der koster serveren forespørgsler og giver en søgemaskine ringe værdi, for eksempel uvigtige eller næsten ens sider. Rækkevidden er også risikoen. Én linje under `*` gælder for hver adresse og hver crawler, der respekterer filen, og en ændring kan tage op til et døgn om at slå igennem, så `Disallow: /` skrevet i den forkerte gruppe lukker et helt site.

Sådan ser det ud i praksis

Dette sites fil er lang, og næsten det hele er kommentarer. Den del, en crawler handler på, er fire linjer:

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

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

Én gruppe dækker alle crawlere, og `Allow: /` passer på hver adresse. En gruppe helt uden regler ville tillade præcis det samme, fordi en adresse, som ikke rammes af en regel, er tilladt, men linjen er skrevet ud, fordi åbenheden er et valg. Kommentaren over den kalder politikken besluttet frem for arvet, og siden om at være klar til AI-agenter beskriver crawlerpolitikken som en af sitets bevidste beslutninger.

`Content-Signal` er ikke en post, RFC 9309 definerer. Den kommer fra Content Signals-ordforrådet, og her giver den samtykke til indsamling til søgning, til svar på et spørgsmål i det øjeblik, det stilles, og til træning. Efter RFC'en er den en af de andre poster: en crawler, der kender konventionen, må læse den, en, der er ældre end konventionen, springer linjen over, som filens egen kommentar siger, og begge skal lade gruppen fortsætte efter den. `Sitemap:` -linjen er en post af samme slags og peger på sitets sitemap-indeks. Kolofonen opsummerer filen på samme måde: den tillader alt, »søgemaskiner og AI-systemer på lige fod«, og peger på sitemappet.

Det, filen udelader, er også besluttet. Blandt de fem anbefalinger, dette site har fravalgt, nævner siden om AI-parathed regler pr. agent og beholder én robots-regel for alle agenter, »fordi en regel skrevet til én crawler i det stille erstatter den generelle for netop den«. Havde filen givet én AI-crawler sin egen gruppe, ville den crawler kun læse den gruppe, og en `Disallow` tilføjet senere under `*` ville aldrig nå den.

Resten er kommentarer, skrevet til den, der åbner filen: begrundelsen for politikken og derefter adresserne på `llms.txt`, `agents.txt`, `humans.txt`, `security.txt` og andre maskinlæsbare filer, holdt som kommentarer, fordi robots.txt ikke har et felt til dem. Det gør kommentarerne til en del af det, der udgives. En `#` -linje i skabelonen sendes byte for byte til hver crawler, og to

gange bar en af dem en note til en vedligeholder, der pegede på et planlægningsdokument i et repository, som ikke er offentligt. Et tjek af det byggede site fejler nu, når en udgivet fil indeholder en sti ind i repositoriet, og vedligeholdernoter i skabelonen står i skabelonkommentarer, som generatoren fjerner.

Hvor det går galt

En regel ment for alle holder op med at gælde for én crawler. Nogen tilføjer en gruppe for en enkelt bot, måske for at holde den ude af én mappe, og fra da af ignorerer den bot alt under `*`, også hver `Disallow`, der står der. RFC 9309 sender kun en crawler til `*`-gruppen, når ingen gruppe nævner den, og Googles dokumentation siger, at specifikke grupper og `*`-gruppen ikke lægges sammen. Gentag hver generel regel i hver navngiven gruppe, eller hold dig til én gruppe, som dette site gør.

En spærret side dukker stadig op i søgeresultaterne. Adressen vises uden beskrivelse, måske med de ord, andre sites brugte, da de linkede til den. Google hentede aldrig siden, så den ved kun, at adressen findes, og hvad andre kalder den. En `Disallow` styrer crawling, ikke indeksering. Fjern reglen og send `noindex` på siden, eller læg siden bag en adgangskode; en `noindex` gemt bag en `Disallow` bliver aldrig læst.

Filen reklamerer for det, den skulle skjule. En linje som `Disallow: /admin/` fortæller enhver læser, høflig crawler eller ej, præcis hvor der skal kigges, og filen er offentlig af natur. RFC'ens afsnit om sikkerhed siger, at stier i filen bliver til at finde, og kræver adgangskontrol i stedet. Læg privat materiale bag godkendelse, og hold det helt ude af filen.

En serverfejl standser crawlingen af et helt site. Siderne indlæses, men forespørgslen efter `/robots.txt` giver en 5xx-status eller en timeout, og en crawler, der følger RFC'en, skal behandle det som et fuldstændigt forbud. Google holder op med at crawle i de første 12 timer og arbejder derefter ud fra sin seneste gode kopi i op til 30 dage. Den modsatte fejl er mere stille: Google behandler et andet 4xx-svar end 429, også 401 og 403, som om der slet ikke var en fil, så hver adresse er åben. Server filen som statisk tekst med status 200, og overvåg selve `/robots.txt`, ikke kun forsiden.

Læs mere

- RFC 9309: Robots Exclusion Protocol — selve standarden: grupper, matching, fejlhåndtering, caching og sikkerhedsafsnittet.
- Introduction to robots.txt — hvad Google siger, filen er til, og hvorfor den ikke skjuler en side i resultaterne.
- How Google interprets the robots.txt specification — statuskoder, caching, de felter Google ignorerer, og hvordan den vælger en gruppe.
- Block Search indexing with noindex — reglen, der faktisk holder en side ude af resultaterne, og hvorfor siden skal kunne crawles.

Ordliste

Vælg ud fra jeres faktiske vilkår

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er robots.txt?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/robots-txt/