**Learn about robots.txt files**

A **robots.txt** file is a file at the root of your site that indicates those parts of your site you don’t want accessed by search engine crawlers. The file uses the [Robots Exclusion Standard](http://en.wikipedia.org/wiki/Robots_exclusion_standard#About_the_standard), which is a protocol with a small set of commands that can be used to indicate access to your site by section and by specific kinds of web crawlers (such as mobile crawlers vs desktop crawlers).

You only need a robots.txt file if your site includes content that you don't want Google or other search engines to index.

**De wikipedia**

L'**estàndard d'exclusió de robots**, també conegut com a **protocol d'exclusió de robots** o siplement **protocol robots.txt**, és una convenció per a evitar que les [aranyes web](http://ca.wikipedia.org/wiki/Aranya_web) que el compleixin accedeixin a la totalitat o part d'un contingut, però que alhora és públic. Els robots són utilitzats pels cercadors per a poder categoritzar i arxivar llocs web, però també els empren alguns desenvolupadors web per a poder revisar-ne el codi que es recull. El complement d'aquest estàndard, per a la inclusió en comptes de l'exclusió de continguts, es coneix com a [sitemaps](http://ca.wikipedia.org/w/index.php?title=Sitemaps&action=edit&redlink=1).

Un fitxer robots.txt en un lloc web funcionarà com una sol·licitud perquè els robots ignorin els [fitxers](http://ca.wikipedia.org/wiki/Fitxer) o [directoris](http://ca.wikipedia.org/wiki/Directori) que hi són especificats en llurs cerques. Això pot estar motivat perquè els cercadors no retornin continguts de caràcter més privat, perquè els continguts que s'exclouen es considerin irrellevants o fora de lloc, o pel desig que una aplicació externa només pugui operar en un cert tipus de dades.

Per a aquells llocs web amb múltiples subdominis, cada un ha de contenir el seu propi fitxer robots.txt. Per exemple, si example.com té un fitxer robots.txt, però no el tingués a.example.com, les regles que s'apliquessin a example.com no es transmetrien a.example.com

Cal dir, per això, que es tracta d'una recomanació. Tot depèn de la cooperació del robot web; i per tant, marcar una àrea d'un lloc web com a exclosa amb el robots.txt no en garanteix la [privadesa](http://ca.wikipedia.org/wiki/Privadesa). Alguns administradors han provat d'utilitzar el fitxer per a fer privades parts invisibles a la resta del món, però així precisament el fitxer està disponible públicament i aquests continguts poden ser llavors determinats fàcilment des del [navegador](http://ca.wikipedia.org/wiki/Navegador).

No hi ha cap organisme oficial que reguli aquests [estàndards](http://ca.wikipedia.org/wiki/Est%C3%A0ndard_t%C3%A8cnic) o cap [document RFC](http://ca.wikipedia.org/wiki/Document_RFC) per al protocol de robots.txt. Aquest sorgeix fruit del consens el juny de 1994 entre els membres de llista de correu de robots ([robots-request@nexor.co.uk](mailto:robots-request@nexor.co.uk))

Aquest exemple **permet a tots els robots** visitar **tots els fitxers** perquè el comodí "\*" es refereix a tots els robots:

User-agent: \* Disallow:

Aquest altre exemple manté **a tots els robots fora**:

User-agent: \* Disallow: /

El següent és un exemple que diu a **totes les aranyes** que no entrin en 4 directoris:

User-agent: \* Disallow: /cgi-bin/ Disallow: /images/ Disallow: /tmp/ Disallow: /private/

A continuació es demana a **una aranya específica** que no entri en un directori concret:

User-agent: BadBot Disallow: /private/

Un altre exemple que demana **a totes les aranyes** no considerar un fitxer concret:

User-agent: \* Disallow: /directory/file.html

Tingueu en compte que la resta de fitxers del directori sí que es processaran.

Un exemple que demostra com poden utilitzar-se els comentaris

# Aquests apareixen darrere d'un símbol "#" a l'inici d'una línia, o després d'una directiva User-agent: \* # match all bots Disallow: / # keep them out

a majoria d'aranyes també reconeixen el paràmetre Sitemap (incloent-hi Google, Yahoo, MSN, Ask). Els [Sitemaps](http://ca.wikipedia.org/w/index.php?title=Sitemaps&action=edit&redlink=1) especifiquen la ubicació de la llista d'URL del lloc web. El paràmetre és independent del parèmtre User-agent i per tant pot col·locar-se allà on es prefereixi.

Sitemap: {{format ref}} http://www.example.com/sitemap.xml.gz

Les [etiquetes meta](http://ca.wikipedia.org/w/index.php?title=Etiquetes_meta&action=edit&redlink=1) d'[HTML](http://ca.wikipedia.org/wiki/HTML) poden utilitzar-se per a excloure els robots d'acord amb el contingut de les pàgines. De totes maneres, com en el cas anterior, també depèn de la cooperació dels programes robot. Per exemple,

<meta name="robots" content="noindex,nofollow" />

Dins de la secció HEAD d'un document [HTML](http://ca.wikipedia.org/wiki/HTML) diu als motors de cerca com ara [Google](http://ca.wikipedia.org/wiki/Google_search), [Yahoo!](http://ca.wikipedia.org/w/index.php?title=Yahoo%21_Search&action=edit&redlink=1), o [MSN](http://ca.wikipedia.org/wiki/MSN_Search) que exclogui la pàgina del seu índex i que no segueixi cap enllaç que hi pugui haver per a cap possible ulterior indexació.

## Understand the limitations of robots.txt

To test which URLs Google can and cannot access on your website, try using the **[robots.txt Tester](https://support.google.com/webmasters/answer/6062598" \t "_blank).**

Before you build your robots.txt, you should know the risks of this URL blocking method. At times, you might want to consider other mechanisms to ensure your URLs are not findable on the web.

### Robots.txt instructions are directives only

The instructions in robots.txt files cannot enforce crawler behavior to your site; instead, these instructions act as directives to the crawlers accessing your site. While Googlebot and other respectable web crawlers obey the instructions in a robots.txt file, other crawlers might not. Therefore, if you want to keep information secure from web crawlers, it’s better to use other blocking methods, such as [password-protecting private files on your server](https://support.google.com/webmasters/answer/93708).

### Different crawlers interpret syntax differently

Although respectable web crawlers follow the directives in a robots.txt file, each crawler might interpret the directives differently. You should know the proper syntax for addressing different web crawlers as some might not understand certain instructions.

### Your robots.txt directives can’t prevent references to your URLs from other sites

While Google won't crawl or index the content blocked by robots.txt, we might still find and index a disallowed URL from other places on the web. As a result, the URL address and, potentially, other publicly available information such as anchor text in links to the site can still appear in Google search results. You can stop your URL from appearing in Google Search results completely by using your robots.txt in combination with other URL blocking methods, such as [password-protecting the files on your server](https://support.google.com/webmasters/answer/93708), or [inserting indexing directive meta tags into your HTML](https://support.google.com/webmasters/answer/93710).

Note: Combining multiple crawling and indexing directives might cause some directives to counteract other directives. Learn how to configure these directives properly by reading the [Combining crawling with indexing / serving directives section](https://developers.google.com/webmasters/control-crawl-index/docs/robots_meta_tag) of the Google Developers documentation.

#### [Block URLs with robots.txt](https://support.google.com/webmasters/topic/6061961?hl=en&ref_topic=4598466)

* [Learn about robots.txt files](javascript:;)
* [Create a robots.txt file](https://support.google.com/webmasters/answer/6062596?hl=en&ref_topic=6061961)
* [Test your robots.txt with the robots.txt Tester](https://support.google.com/webmasters/answer/6062598?hl=en&ref_topic=6061961)
* [Submit your updated robots.txt to Google](https://support.google.com/webmasters/answer/6078399?hl=en&ref_topic=6061961)

Create a robots.txt file

In order to make a robots.txt file, you need access to the root of your domain. If you're unsure about how to access the root, you can contact your web hosting service provider. Also, if you know you can't access to the root of the domain, you can use alternative blocking methods, such [as password-protecting the files on your server](https://support.google.com/webmasters/answer/93708" \t "_blank), and [inserting meta tags into your HTML](https://support.google.com/webmasters/answer/93710" \t "_blank).

You can make or edit an existing robots.txt file using the **[robots.txt Tester](https://support.google.com/webmasters/answer/6062598" \t "_blank)** tool. This allows you to test your changes as you adjust your robots.txt.

## Learn robots.txt syntax

The simplest robots.txt file uses two key words, **User-agent** and **Disallow**. User-agents are search engine robots (or web crawler software); most user-agents are listed in the [Web Robots Database](http://www.robotstxt.org/db.html" \t "_blank). Disallow is a command for the user-agent that tells it not to access a particular URL. On the other hand, to give Google access to a particular URL that is a child directory in a disallowed parent directory, then you can use a third key word, **Allow.**

Google uses several user-agents, such as **Googlebot** for Google Search and **Googlebot-Image** for Google Image Search. Most Google user-agents follow the rules you set up for Googlebot, but you can override this option and make specific rules for only certain Google user-agents as well.

The syntax for using the keywords is as follows:

**User-agent**: [the name of the robot the following rule applies to]

**Disallow**: [the URL path you want to block]

**Allow:** [the URL path in of a subdirectory, within a blocked parent directory, that you want to unblock]

These two lines are together considered a single entry in the file, where the Disallow rule only applies to the user-agent(s) specified above it. You can include as many entries as you want, and multiple Disallow lines can apply to multiple user-agents, all in one entry.You can set the User-agent command to apply to all web crawlers by listing an asterisk (\*) as in the example below:

**User-agent:** \*

|  |  |
| --- | --- |
| ****Block...**** | ****Sample**** |
| **The entire site**with aforward slash (/): | Disallow: / |
| **A directory and its contents** by following the directory name with a forward slash: | Disallow: /sample-directory/ |
| **A webpage** by listing the page after the slash: | Disallow: /private\_file.html |
| **A specific image from Google Images:** | User-agent: Googlebot-Image  Disallow: /images/dogs.jpg |
| **All images on your site from Google Images:** | User-agent: Googlebot-Image  Disallow: / |
| **Files of a specific file type** (for example, .gif): | User-agent: Googlebot  Disallow: /\*.gif$ |
| **Pages on your site, but show AdSense ads on those pages**, disallow all web crawlers other than Mediapartners-Google. This implementation hides your pages from search results, but the **Mediapartners-Google** web crawler can still analyze them to decide what ads to show visitors to your site. | User-agent: \*  Disallow: /  User-agent: Mediapartners-Google  Allow: / |

Note that directives are **case-sensitive**. For instance, Disallow: /file.asp would block http://www.example.com/file.asp, but would allow http://www.example.com/File.asp. Googlebot also ignores white-space, and unknown directives in the robots.txt.

|  |  |
| --- | --- |
| ****Pattern-matching rule**** | ****Sample**** |
| **To block any sequence of characters**, use an asterisk (\*). For instance, the sample code blocks access to all subdirectories that begin with the word "private": | User-agent: Googlebot  Disallow: /private\*/ |
| **To block access to all URLs that include question marks (?).** For example, the sample code blocks URLs that begin with your domain name, followed by any string, followed by a question mark, and ending with any string: | User-agent: Googlebot  Disallow: /\*? |
| **To block any URLs that end in a specific way**, use $. For instance, the sample code blocks any URLs that end with .xls: | User-agent: Googlebot  Disallow: /\*.xls$ |
| **To block patterns with the Allow and Disallow directives**, see the sample to the right. In this example, a ? indicates a session ID. URLs that contain these IDs should typically be blocked from Google to prevent web crawlers from crawling duplicate pages. Meanwhile, if some URLs ending with ? are versions of the page that you want to include, you can use the following approach of combining Allow and Disallow directives:   1. The **Allow: /\*?$** directive allows any URL that ends in a ? (more specifically, it allows a URL that begins with your domain name, followed by a string, followed by a ?, with no characters after the ?). 2. The **Disallow: / \*?** directive blocks any URL that includes a ? (more specifically, it blocks a URL that begins with your domain name, followed by a string, followed by a question mark, followed by a string). | User-agent: \*  Allow: /\*?$  Disallow: /\*? |

### Save your robots.txt file

You must apply the following saving conventions so that Googlebot and other web crawlers can find and identify your robots.txt file:

* You must save your robots.txt code as a text file,
* You must place the file in the highest-level directory of your site (or the root of your domain), and
* The robots.txt file must be named robots.txt.

As an example, a robots.txt file saved at the root of example.com, at the URL address http://www.example.com/robots.txt, can be discovered by web crawlers, but a robots.txt file at http://www.example.com/not\_root/robots.txt cannot be found by any web crawler.

Test your robots.txt file with robots.txt tester

The **[robots.txt Tester](https://www.google.com/webmasters/tools/robots-testing-tool" \t "_blank)** tool shows you whether your robots.txt file blocks Google web crawlers from specific URLs on your site. For example, you can use this tool to test whether the **Googlebot-Image** crawler can crawl the URL of an image you wish to block from Google Image Search.

You can submit a URL to the **robots.txt Tester** tool. The tool operates as Googlebot would to check your robots.txt file and verifies that your URL has been blocked properly.

### Test your robots.txt

1. From the [Webmaster Tools Home page](http://google.com/webmasters/tools/home?hl=en" \t "_blank), choose the site whose robots.txt file you want to test.
2. Expand the **Crawl** heading on the left dashboard, and select the **robots.txt Tester** tool.
3. Make changes to your live robots.txt file in the **text editor**.
4. Scroll through the robots.txt code to locate the highlighted **syntax warnings** and **logic errors**. The number of syntax warnings and logic errors is shown immediately below the editor.
5. Type in an extension of the URL or path in the text box at the bottom of the page.
6. Select the **user-agent** you want to simulate in the dropdown list to the right of the text box.
7. Click the **TEST** next to the dropdown user-agent list to run the simulation.
8. Check to see if **TEST** button now reads **ACCEPTED** or **BLOCKED** to find out if the URL you entered is blocked from Google web crawlers.

### Please take note of the following limitations of the robots.txt Tester tool:

* Changes you make in the tool editor are not automatically saved to your web server. You need to copy and paste the content from the editor into the robots.txt file stored on your server.
* The **robots.txt Tester** tool only tests your robots.txt with Google user-agents or web crawlers, like Googlebot. We cannot predict how other web crawlers interpret your robots.txt file.

Submit your updated robots.txt to google

The **Submit** function of the **robots.txt Tester** tool allows you to easily put in place and ask Google to more quickly crawl and index a new robots.txt file for your site. Update and notify Google of changes to your robots.txt file by following the steps below.

1. Click **Submit** in the bottom-right corner of the robots.txt editor. This action opens up a Submit dialog.

2. Download your edited robots.txt code from the **robots.txt Tester** page by clicking **Download** in the Submit dialog.

3. Upload your new robots.txt file to the root of your domain as a text file named robots.txt (the URL for your robots.txt file should be /robots.txt).

**If you do not have permission to upload files to the root of your domain, you should contact your domain manager to make changes.**

For example, if your site home page resides under subdomain.example.com/site/example/, you likely cannot update the robots file subdomain.example.com/robots.txt. In this case, you should contact the owner of example.com/ to make any necessary changes to the robots.txt file.

4. Click **Verify live version** to see that your live robots.txt is the version that you want Google to crawl.

5. Click **Submit live version** to notify Google that changes have been made to your robots.txt file and request that Google crawl it.

6. Check that your newest version was successfully crawled by Google by refreshing the page in your browser to update the tool’s editor and see your live robots.txt code. After you refresh the page, you can also click the dropdown above the text editor to view the timestamp of when Google first saw the **latest version** of your robots.txt file.