



链滴

# wget 下载百度网盘内容

作者: [gideal](#)

原文链接: <https://ld246.com/article/1385882048322>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>走了半年回来发现b3log更新了好多版本，于是开始更新可是发现国内一直是403无法部署不得用vps部署，在下载软件包更新时候发现b3log是用百度网盘的，由于可是vps是没有桌面的只能用wge下载，以下是下载方法</p>  
<p>首先用谷歌浏览器下载到本机然后再谷歌浏览器的输入网址的地方输入chrom e://downloads</p>  
<p>会出现下载的历史记录在里边复制你刚下载的软件包的下载路径之后登录vps</p>  
<p>在shell里输入</p>  
<p>wget -O "http://qd.cache.baidupcs.com/file/5aee851172421454d6f02e6f54c7fab?xcode=74464e71ab3ddeba1c03ef395b73dad483e5b54eb4fe109e&mp;fid=570729916-250528-687729370&time=1385874330&sign=FDTAXER-DCb74ccc5511e5e8fedcff06b081203-NhhhyfY%2BoAeKy1648IAfP0m8e5E%3D&to=qc&f=N,B,U,nc&expires=8h&rt=pr&r=875239027&logid=1053499758&uk=570729916&fn=b3log-solo-gae-0.6.5.zip" </p>  
<p>红色部分为你在谷歌浏览器里复制的那个原始地址 这样就下载成功了</p>  
<p>以下是wget的一些基本用法 你也可以通过在shell里输入wget --help来查看</p>  
<p><br />GNU Wget 1.12, 非交互式的网络文件下载工具。<br />用法: wget [选项]... [URL]...<p>  
<p>长选项所必须的参数在使用短选项时也是必须的。</p>  
<pre class="brush: bash">开始:  
-V, --version 显示 Wget 的版本信息并退出。  
-h, --help 打印此帮助。  
-b, --background 启动后转入后台。  
-e, --execute=COMMAND 运行一个 '.wgetrc' 风格的命令。

登入并输入文件:

-o, --output-file=FILE 将信息写入 FILE。  
-a, --append-output=FILE 将信息添加至 FILE。  
-d, --debug 打印大量调试信息。  
-q, --quiet 安静模式(无信息输出)。  
-v, --verbose 详尽的输出(此为默认值)。  
-nv, --no-verbose 关闭详尽输出，但不进入安静模式。  
-i, --input-file=FILE 下载本地或外部 FILE 中的 URLs。  
-F, --force-html 把输入文件当成 HTML 文件。  
-B, --base=URL 解析与 URL 相关的 HTML 输入文件 (由 -i -F 选项指定)。

下载:

-t, --tries=NUMBER 设置重试次数为 NUMBER (0 代表无限制)。  
--retry-connrefused 即使拒绝连接也是重试。  
-O, --output-document=FILE 将文档写入 FILE。  
-nc, --no-clobber 不要重复下载已存在的文件。  
-c, --continue 继续下载部分下载的文件。  
--progress=TYPE 选择进度条类型。  
-N, --timestamping 只获取比本地文件新的文件。

-S, --server-response 打印服务器响应。  
--spider 不下载任何文件。  
-T, --timeout=SECONDS 将所有超时设为 SECONDS 秒。  
--dns-timeout=SECS 设置 DNS 查寻超时为 SECS 秒。  
--connect-timeout=SECS 设置连接超时为 SECS 秒。  
--read-timeout=SECS 设置读取超时为 SECS 秒。  
-w, --wait=SECONDS 等待间隔为 SECONDS 秒。  
--waitretry=SECONDS 在取回文件的重试期间等待 1..SECONDS 秒。  
--random-wait 取回时等待 0...2\*WAIT 秒。  
--no-proxy 关闭代理。  
-Q, --quota=NUMBER 设置取回配额为 NUMBER 字节。  
--bind-address=ADDRESS 绑定至本地主机上的 ADDRESS (主机名或是 IP)。  
--limit-rate=RATE 限制下载速率为 RATE。  
--no-dns-cache 关闭 DNS 查寻缓存。  
--restrict-file-names=OS 限定文件名中的字符为 OS 允许的字符。  
--ignore-case 匹配文件/目录时忽略大小写。  
-4, --inet4-only 仅连接至 IPv4 地址。  
-6, --inet6-only 仅连接至 IPv6 地址。  
--prefer-family=FAMILY 首先连接至指定协议的地址  
FAMILY 为 IPv6, IPv4 或是 none。  
--user=USER 将 ftp 和 http 的用户名均设置为 USER。  
--password=PASS 将 ftp 和 http 的密码均设置为 PASS。  
--ask-password 提示输入密码。  
--no-iri 关闭 IRI 支持。  
--local-encoding=ENC IRI 使用 ENC 作为本地编码。  
--remote-encoding=ENC 使用 ENC 作为默认远程编码。

#### 目录:

-nd, --no-directories 不创建目录。  
-x, --force-directories 强制创建目录。  
-nH, --no-host-directories 不要创建主目录。  
--protocol-directories 在目录中使用协议名称。  
-P, --directory-prefix=PREFIX 以 PREFIX/... 保存文件  
--cut-dirs=NUMBER 忽略 NUMBER 个远程目录路径。

#### HTTP 选项:

--http-user=USER 设置 http 用户名为 USER。  
--http-password=PASS 设置 http 密码为 PASS。

--no-cache 不在服务器上缓存数据。  
--default-page=NAME 改变默认页  
(默认页通常是 "index.html" ) 。  
-E, --adjust-extension 以合适的扩展名保存 HTML/CSS 文档。  
--ignore-length 忽略头部的 'Content-Length' 区域。  
--header=STRING 在头部插入 STRING。  
--max-redirect 每页所允许的最大重定向。  
--proxy-user=USER 使用 USER 作为代理用户名。  
--proxy-password=PASS 使用 PASS 作为代理密码。  
--referer=URL 在 HTTP 请求头包含 'Referer: URL' 。  
--save-headers 将 HTTP 头保存至文件。  
-U, --user-agent=AGENT 标识为 AGENT 而不是 Wget/VERSION。  
--no-http-keep-alive 禁用 HTTP keep-alive(永久连接)。  
--no-cookies 不使用 cookies。  
--load-cookies=FILE 会话开始前从 FILE 中载入 cookies。  
--save-cookies=FILE 会话结束后保存 cookies 至 FILE。  
--keep-session-cookies 载入并保存会话(非永久) cookies。  
--post-data=STRING 使用 POST 方式; 把 STRING 作为数据发送。  
--post-file=FILE 使用 POST 方式; 发送 FILE 内容。  
--content-disposition 当选中本地文件名时  
允许 Content-Disposition 头部(尚在实验)。  
--auth-no-challenge send Basic HTTP authentication information  
without first waiting for the server's  
challenge.

HTTPS (SSL/TLS) 选项:

--secure-protocol=PR 选择安全协议, 可以是 auto、SSLv2、  
SSLv3 或是 TLSv1 中的一个。  
--no-check-certificate 不要验证服务器的证书。  
--certificate=FILE 客户端证书文件。  
--certificate-type=TYPE 客户端证书类型, PEM 或 DER。  
--private-key=FILE 私钥文件。  
--private-key-type=TYPE 私钥文件类型, PEM 或 DER。  
--ca-certificate=FILE 带有一组 CA 认证的文件。  
--ca-directory=DIR 保存 CA 认证的哈希列表的目录。  
--random-file=FILE 带有生成 SSL PRNG 的随机数据的文件。  
--egd-file=FILE 用于命名带有随机数据的 EGD 套接字的文件。

## FTP 选项:

- ftp-user=USER 设置 ftp 用户名为 USER。
- ftp-password=PASS 设置 ftp 密码为 PASS。
- no-remove-listing 不要删除 '.listing' 文件。
- no-glob 不在 FTP 文件名中使用通配符展开。
- no-passive-ftp 禁用 "passive" 传输模式。
- retr-symlinks 递归目录时, 获取链接的文件(而非目录)。

## 递归下载:

- r, --recursive 指定递归下载。
- l, --level=NUMBER 最大递归深度( inf 或 0 代表无限制, 即全部下载)。
- delete-after 下载完成后删除本地文件。
- k, --convert-links 让下载得到的 HTML 或 CSS 中的链接指向本地文件。
- K, --backup-converted 在转换文件 X 前先将它备份为 X.orig。
- m, --mirror -N -r -l inf --no-remove-listing 的缩写形式。
- p, --page-requisites 下载所有用于显示 HTML 页面的图片之类的元素。
- strict-comments 开启 HTML 注释的精确处理(SGML)。

## 递归接受/拒绝:

- A, --accept=LIST 逗号分隔的可接受的扩展名列表。
- R, --reject=LIST 逗号分隔的要拒绝的扩展名列表。
- D, --domains=LIST 逗号分隔的可接受的域列表。
- exclude-domains=LIST 逗号分隔的要拒绝的域列表。
- follow-ftp 跟踪 HTML 文档中的 FTP 链接。
- follow-tags=LIST 逗号分隔的跟踪的 HTML 标识列表。
- ignore-tags=LIST 逗号分隔的忽略的 HTML 标识列表。
- H, --span-hosts 递归时转向外部主机。
- L, --relative 只跟踪有关系的链接。
- I, --include-directories=LIST 允许目录的列表。
- X, --exclude-directories=LIST 排除目录的列表。
- np, --no-parent [</pre>](#)