

## CURL 常量表

选项	类型	可选 VALUE 值
CURLOPT_AUTOREFERER	bool	当根据 Location:重定向时, 自动设置 header 中的 Referer:信息。
CURLOPT_BINARYTRANSFER	bool	在启用 CURLOPT_RETURNTRANSFER 的时候, 返回原生的 (Raw) 输出。
CURLOPT_COOKIESESSION	bool	启用时 curl 会仅仅传递一个 session cookie, 忽略其他的 cookie, 默认状况下 cURL 会将所有的 cookie 返回给服务端。 session cookie 是指那些用来判断服务器端的 session 是否有效而存在的 cookie。
CURLOPT_CRLF	bool	启用时将 Unix 的换行符转换成回车换行符。
CURLOPT_DNS_USE_GLOBAL_CACHE	bool	启用时会启用一个全局的 DNS 缓存, 此项为线程安全的, 并且默认启用。
CURLOPT_FAILONERROR	bool	显示 HTTP 状态码, 默认行为是忽略编号小于等于 400 的 HTTP 信息。
CURLOPT_FILETIME	bool	启用时会尝试修改远程文档中的信息。 结果信息会通过 curl_getinfo()函数的 CURLINFO_FILETIME 选项返回。
CURLOPT_FOLLOWLOCATION	bool	启用时会将服务器服务器返回的"Location: "放在 header 中递归的返回给服务器。 使用 CURLOPT_MAXREDIRS 可以限定递归返回的数量。
CURLOPT_FORBID_REUSE	bool	在完成交互以后强迫断开连接, 不能重用。
CURLOPT_FRESH_CONNECT	bool	强制获取一个新的连接, 替代缓存中的连接。
CURLOPT_FTP_USE_EPRT	bool	启用时当 FTP 下载时, 使用 EPRT (或 LPRT)命令。 设置为 FALSE 时禁用 EPRT 和 LPRT, 仅使用 PORT 命令。

选项	类型	可选 VALUE 值
CURLOPT_FTP_USE_EPSV	bool	启用时，在 FTP 传输过程中回复到 PASV 模式前首先尝试 EPSV 命令。 设置为 FALSE 时禁用 EPSV 命令。
CURLOPT_FTPAPPEND	bool	启用时追加写入文件而不是覆盖它。
CURLOPT_FTPASCII	bool	CURLOPT_TRANSFERTEXT 的别名。
CURLOPT_FTPLISTONLY	bool	启用时只列出 FTP 目录的名字。
CURLOPT_HEADER	bool	启用时会将头文件的信息作为数据流输出。
CURLINFO_HEADER_OUT	bool	启用时追踪句柄的请求字符串。
CURLOPT_HTTPGET	bool	启用时会设置 HTTP 的 method 为 GET，因为 GET 是默认是，所以只在被修改的情况下使用。
CURLOPT_HTTPPROXYTUNNEL	bool	启用时会通过 HTTP 代理来传输。
CURLOPT_MUTE	bool	启用时将 cURL 函数中所有修改过的参数恢复默认值。
CURLOPT_NETRC	bool	在连接建立以后，访问 ~/.netrc 文件获取用户名和密码信息连接远程站点。
CURLOPT_NOBODY	bool	启用时将不对 HTML 中的 BODY 部分进行输出。
CURLOPT_NOPROGRESS	bool	启用时关闭 curl 传输的进度条，此项的默认设置为启用。 PHP 自动地设置这个选项为 TRUE，这个选项仅仅应当在以调试为目的时被改变。
CURLOPT_NOSIGNAL	bool	启用时忽略所有的 curl 传递给 php 进行的信号。在 SAPI 多线程传输时此项被默认启用。
CURLOPT_POST	bool	启用时会发送一个常规的 POST 请求，类型为：application/x-www-form-urlencoded，就像表单提交的一样。
CURLOPT_PUT	bool	启用时允许 HTTP 发送文件，必须同时设置 CURLOPT_INFILE 和 CURLOPT_INFILESIZE。

选项	类型	可选 VALUE 值
CURLOPT_RETURNTRANSFER	bool	将 curl_exec()获取的信息以文件流的形式返回，而不是直接输出。
CURLOPT_SSL_VERIFYPEER	bool	禁用后 cURL 将终止从服务端进行验证。 使用 CURLOPT_CAINFO 选项设置证书使用 CURLOPT_CAPATH 选项设置证书目录， 如果 CURLOPT_SSL_VERIFYPEER(默认值为 2)被启用，CURLOPT_SSL_VERIFYHOST 需要被设置成 TRUE，否则设置为 FALSE。
CURLOPT_TRANSFERTEXT	bool	启用后对 FTP 传输使用 ASCII 模式。 对于 LDAP，它检索纯文本信息而非 HTML。 在 Windows 系统上，系统不会把 STDOUT 设置成 binary 模式。
CURLOPT_UNRESTRICTED_AUTH	bool	在使用 CURLOPT_FOLLOWLOCATION 产生的 header 中的多个 locations 中持续追加用户名和密码信息，即使域名已发生改变。
CURLOPT_UPLOAD	bool	启用后允许文件上传。
CURLOPT_VERBOSE	bool	启用时会汇报所有的信息，存放在 STDERR 或指定的 CURLOPT_STDERR 中。
CURLOPT_BUFFERSIZE	Integer	每次获取的数据中读入缓存的大小，但是不保证这个值每次都会被填满。
CURLOPT_CLOSEPOLICY	Integer	不是 CURLCLOSEPOLICY_LEAST_RECENTLY_USED 就是 CURLCLOSEPOLICY_OLDEST，还存在另外三个 CURLCLOSEPOLICY_，但是 cURL 暂时还不支持。
CURLOPT_CONNECTTIMEOUT	Integer	在发起连接前等待的时间，如果设置为 0，则无限等待。
CURLOPT_CONNECTTIMEOUT_MS	Integer	尝试连接等待的时间，以毫秒为单位。如果设置为 0，则无限等待。
CURLOPT_DNS_CACHE_TIMEOUT	Integer	设置在内存中保存 DNS 信息的时间，默认为 120 秒。

选项	类型	可选 VALUE 值
CURLOPT_FTPSSLAUTH	Integer	FTP 验证方式: CURLFTPAUTH_SSL (首先尝试 SSL) CURLFTPAUTH_TLS (首先尝试 TLS) CURLFTPAUTH_DEFAULT (让 cURL 自动决定)。
CURLOPT_HTTP_VERSION	Integer	CURL_HTTP_VERSION_NONE (默认值, 让 cURL 自己判断使用哪个版本) CURL_HTTP_VERSION_1_0 (强制使用 HTTP/1.0) CURL_HTTP_VERSION_1_1 (强制使用 HTTP/1.1)。
CURLOPT_HTTPAUTH	Integer	使用的 HTTP 验证方法, 可选的值有: CURLAUTH_BASIC CURLAUTH_DIGEST CURLAUTH_GSSNEGOTIATE CURLAUTH_NTLM CURLAUTH_ANY CURLAUTH_ANYSAFE。 可以使用 位域(或)操作符分隔多个值, cURL 让服务器选择一个支持最好的值。 CURLAUTH_ANY 等价于 CURLAUTH_BASIC   CURLAUTH_DIGEST   CURLAUTH_GSSNEGOTIATE   CURLAUTH_NTLM. CURLAUTH_ANYSAFE 等价于

选项	类型	可选 VALUE 值
		CURLAUTH_DIGEST   CURLAUTH_GSSNEGOTIATE   CURLAUTH_NTLM.
CURLOPT_INFILESIZE	Integer	设定上传文件的大小限制，字节(byte)为单位。
CURLOPT_LOW_SPEED_LIMIT	Integer	当传输速度小于 CURLOPT_LOW_SPEED_LIMIT 时(bytes/sec)， PHP 会根据 CURLOPT_LOW_SPEED_TIME 来判断是否因太慢而取消传输。
CURLOPT_LOW_SPEED_TIME	Integer	当传输速度小于 CURLOPT_LOW_SPEED_LIMIT 时(bytes/sec)， PHP 会根据 CURLOPT_LOW_SPEED_TIME 来判断是否因太慢而取消传输。
CURLOPT_MAXCONNECTS	Integer	允许的最大连接数量，超过是会通过 CURLOPT_CLOSEPOLICY 决定应该停止哪些连接。
CURLOPT_MAXREDIRS	Integer	指定最多的 HTTP 重定向的数量，这个选项是和 CURLOPT_FOLLOWLOCATION 一起使用的。
CURLOPT_PORT	Integer	用来指定连接端口。（可选项）
CURLOPT_PROTOCOLS	Integer	<p>CURLPROTO_*的位域指。如果被启用，位域值会限定 libcurl 在传输过程中有哪些可使用的协议。这将允许你在编译 libcurl 时支持众多协议，但是限制只是用它们中被允许使用的一个子集。默认 libcurl 将会使用全部它支持的协议。参见 CURLOPT_REDIR_PROTOCOLS.</p> <p>可用的协议选项为：</p> <p>CURLPROTO_HTTP</p> <p>CURLPROTO_HTTPS</p> <p>CURLPROTO_FTP</p> <p>CURLPROTO_FTPS</p>

选项	类型	可选 VALUE 值
		CURLPROTO_SCP CURLPROTO_SFTP CURLPROTO_TELNET CURLPROTO_LDAP CURLPROTO_LDAPS CURLPROTO_DICT CURLPROTO_FILE CURLPROTO_TFTP CURLPROTO_ALL
CURLOPT_PROXYAUTH	Integer	HTTP 代理连接的验证方式。 使用在 CURLOPT_HTTPAUTH 中的位域标志来设置相应选项。 对于代理验证只有 CURLAUTH_BASIC 和 CURLAUTH_NTLM 当前被支持。
CURLOPT_PROXYPORT	Integer	代理服务器的端口。端口也可以在 CURLOPT_PROXY 中进行设置。
CURLOPT_PROXYTYPE	Integer	不是 CURLPROXY_HTTP (默认值) 就是 CURLPROXY_SOCKS5。
CURLOPT_REDIR_PROTOCOLS	Integer	CURLPROTO_* 中的位域值。如果被启用，位域值将会限制传输线程在 CURLOPT_FOLLOWLOCATION 开启时跟随某个重定向时可使用的协议。 这将使你对重定向时限制传输线程使用被允许的协议子集默认 libcurl 将会允许除 FILE 和 SCP 之外的全部协议。

选项	类型	可选 VALUE 值
		这个和 7.19.4 预发布版本种无条件地跟随所有支持的协议有一些不同。关于协议常量，请参照 CURLOPT_PROTOCOLS。
CURLOPT_RESUME_FROM	Integer	在恢复传输时传递一个字节偏移量（用来断点续传）。
CURLOPT_SSL_VERIFYHOST	Integer	1 检查服务器 SSL 证书中是否存在一个公用名(common name)。 译者注：公用名(Common Name)一般来讲就是填写你将要申请 SSL 证书的域名 (domain)或子域名(sub domain)。 2 检查公用名是否存在，并且是否与提供的主机名匹配。
CURLOPT_SSLVERSION	Integer	使用的 SSL 版本(2 或 3)。默认情况下 PHP 会自己检测这个值，尽管有些情况下需要手动地进行设置。
CURLOPT_TIMECONDITION	Integer	如果在 CURLOPT_TIMEVALUE 指定的某个时间以后被编辑过， 则使用 CURL_TIMECOND_IFMODSINCE 返回页面； 如果没有被修改过，并且 CURLOPT_HEADER 为 true，则返回一个"304 Not Modified"的 header； CURLOPT_HEADER 为 false，则使用 CURL_TIMECOND_IFUNMODSINCE； 默认值为 CURL_TIMECOND_IFUNMODSINCE。
CURLOPT_TIMEOUT	Integer	设置 cURL 允许执行的最长秒数。
CURLOPT_TIMEOUT_MS	Integer	设置 cURL 允许执行的最长毫秒数。
CURLOPT_TIMEVALUE	Integer	设置一个 CURLOPT_TIMECONDITION 使用的时间戳， 在默认状态下使用的是 CURL_TIMECOND_IFMODSINCE。

选项	类型	可选 VALUE 值
CURLOPT_CAINFO	string	一个保存着 1 个或多个用来让服务端验证的证书的文件名。 这个参数仅仅在和 CURLOPT_SSL_VERIFYPEER 一起使用时才有意义。
CURLOPT_CAPATH	string	一个保存着多个 CA 证书的目录。这个选项是和 CURLOPT_SSL_VERIFYPEER 一起使用的。
CURLOPT_COOKIE	string	设定 HTTP 请求中"Cookie: "部分的内容。 多个 cookie 用分号分隔，分号后带一个空格(例如， "fruit=apple; colour=red")。
CURLOPT_COOKIEFILE	string	包含 cookie 数据的文件名 cookie 文件的格式可以是 Netscape 格式，或者只是纯 HTTP 头部信息存入文件。
CURLOPT_COOKIEJAR	string	连接结束后保存 cookie 信息的文件。
CURLOPT_CUSTOMREQUEST	string	使用一个自定义的请求信息来代替"GET"或"HEAD"作为 HTTP 请求。 这对于执行"DELETE"或者其他更隐蔽的 HTTP 请求。 有效值如"GET", "POST", "CONNECT"等等。 也就是说，不要在这里输入整个 HTTP 请求。例如输入"GET /index.html HTTP/1.0\r\n\r\n"是不正确的。 Note: 在确定服务器支持这个自定义请求的方法前不要使用。
CURLOPT_EGDSOCKET	string	类似 CURLOPT_RANDOM_FILE，除了一个 Entropy Gathering Daemon 套接字。
CURLOPT_ENCODING	string	HTTP 请求头中"Accept-Encoding: "的值。 支持的编码有"identity", "deflate"和"gzip"。 如果为空字符串""，请求头会发送所有支持的编码类型。



选项	类型	可选 VALUE 值
CURLOPT_FTPPORT	string	<p>这个值将被用来获取供 FTP"POST"指令所需要的 IP 地址。</p> <p>"POST"指令告诉远程服务器连接到我们指定的 IP 地址。</p> <p>这个字符串可以是纯文本的 IP 地址、主机名、一个网络接口名（UNIX 下）或者只是一个 '-' 来使用默认的 IP 地址。</p>
CURLOPT_INTERFACE	string	网络发送接口名，可以是一个接口名、IP 地址或者是一个主机名。
CURLOPT_KRB4LEVEL	string	<p>KRB4 (Kerberos 4) 安全级别。</p> <p>下面的任何值都是有效的(从低到高的顺序): "clear"、"safe"、"confidential"、"private"。</p> <p>如果字符串和这些都不匹配，将使用"private"。</p> <p>这个选项设置为 NULL 时将禁用 KRB4 安全认证。目前 KRB4 安全认证只能用于 FTP 传输。</p>
CURLOPT_POSTFIELDS	string	<p>全部数据使用 HTTP 协议中的"POST"操作来发送。</p> <p>要发送文件，在文件名前面加上@前缀并使用完整路径。</p> <p>这个参数可以通过 urlencoded 后的字符串类似'para1=val1&amp;para2=val2&amp;...'，或使用一个以字段名为键值，字段数据为值的数组。</p> <p>如果 value 是一个数组，Content-Type 头将会被设置成 multipart/form-data。</p>
CURLOPT_PROXY	string	HTTP 代理通道。
CURLOPT_PROXYUSERPWD	string	一个用来连接到代理的"[username]:[password]"格式的字符串。
CURLOPT_RANDOM_FILE	string	一个被用来生成 SSL 随机数种子的文件名。
CURLOPT_RANGE	string	以"X-Y"的形式，其中 X 和 Y 都是可选项获取数据的范围，以字节计。HTTP 传输线程也支持几个这样

选项	类型	可选 VALUE 值
		的重复项中间用逗号分隔如"X-Y,N-M"。
CURLOPT_REFERER	string	在 HTTP 请求头中"Referer: "的内容。
CURLOPT_SSL_CIPHER_LIST	string	一个 SSL 的加密算法列表。例如 RC4-SHA 和 TLSv1 都是可用的加密列表。
CURLOPT_SSLCERT	string	一个包含 PEM 格式证书的文件名。
CURLOPT_SSLCERTPASSWD	string	使用 CURLOPT_SSLCERT 证书需要的密码。
CURLOPT_SSLCERTTYPE	string	证书的类型。支持的格式有"PEM" (默认值), "DER"和"ENG"。
CURLOPT_SSLENGINE	string	用来在 CURLOPT_SSLKEY 中指定的 SSL 私钥的加密引擎变量。
CURLOPT_SSLENGINE_DEFAULT	string	用来做非对称加密操作的变量。
CURLOPT_SSLKEY	string	包含 SSL 私钥的文件名。
CURLOPT_SSLKEYPASSWD	string	在 CURLOPT_SSLKEY 中指定了的 SSL 私钥的密码。 Note: 由于这个选项包含了敏感密码信息, 记得保证这个 PHP 脚本的安全。
CURLOPT_SSLKEYTYPE	string	CURLOPT_SSLKEY 中规定的私钥的加密类型, 支持的密钥类型为"PEM"(默认值)、"DER"和"ENG"。
CURLOPT_URL	string	需要获取的 URL 地址, 也可以在 curl_init()函数中设置。
CURLOPT_USERAGENT	string	在 HTTP 请求中包含一个"User-Agent: "头的字符串。
CURLOPT_USERPWD	string	传递一个连接中需要的用户名和密码, 格式为: "[username]:[password]"。
CURLOPT_FILE	stream	设置输出文件的位置, 值是一个资源类型, 默认为 STDOUT (浏览器)。
CURLOPT_INFILE	stream	在上传文件的时候需要读取的文件地址, 值是一个资源类型。

选项	类型	可选 VALUE 值
CURLOPT_STDERR	stream	设置一个错误输出地址，值是一个资源类型，取代默认的 STDERR。
CURLOPT_WRITEHEADER	stream	设置 header 部分内容的写入的文件地址，值是一个资源类型。
CURLOPT_HEADERFUNCTION	callback	设置一个回调函数，这个函数有两个参数，第一个是 cURL 的资源句柄，第二个是输出的 header 数据。 header 数据的输出必须依赖这个函数，返回已写入的数据大小。
CURLOPT_PASSWDFUNCTION	callback	设置一个回调函数，有三个参数，第一个是 cURL 的资源句柄，第二个是一个密码提示符，第三个参数是密码长度允许的最大值。返回密码的值。
CURLOPT_PROGRESSFUNCTION	callback	设置一个回调函数，有三个参数，第一个是 cURL 的资源句柄，第二个是一个文件描述符资源，第三个是长度。返回包含的数据。
CURLOPT_READFUNCTION	callback	拥有两个参数的回调函数，第一个是参数是会话句柄，第二是 HTTP 响应头信息的字符串。 使用此函数，将自行处理返回的数据。返回值为数据大小，以字节计。返回 0 代表 EOF 信号。
CURLOPT_WRITEFUNCTION	callback	拥有两个参数的回调函数，第一个是参数是会话句柄，第二是 HTTP 响应头信息的字符串。 使用此回调函数，将自行处理响应头信息。响应头信息是整个字符串。 设置返回值为精确的已写入字符串长度。发生错误时传输线程终止。