

# NETCURLv6.1

## Code

Currently, and still in a one-man-army mode, the code for next version (6.1) could be found [hereere](#). There's a big difference between 6.0 and 6.1 - it's more standardized and PSR-4 compliant. However, the goal is to keep compatibility to 6.0 so that developers can upgrade to proper code without ripping their hair off their heads. 6.1 will be merged into the master branch as soon as it has been properly tested which is primarily done with [Bamboo](#), [Pipelines](#) and [pipelines with regular ecommerce](#) systems.

## Requirements and recommended addons

- curl - not necessary, package will fail over to streams without curl.
- ssl - not necessary, but you will proceed without https support.
- soap/xml - Only necessary if you tend to use XML and SoapCalls.
- DOMDocument - Only necessary if you tend to use reading of plain html.
- laminas - Only necessary if you tend to use extended RSS reading or read DOMDocuments with an extension. Package will fail over to xml without laminas.

## Live information

There's a [Mailinglist](#) put up for everything concerning netcurl. That's also where you can find release information (for now). You can subscribe to the list [here](#).

## Supported Versions

Branch	Development started	Initial Release	Active Support Until	Maintenance	PHP Support
5.0	2016-12-16	Never	2017-08-17	2017-08-17	5.4 - 7.4 <i>Follows 6.0 branch.</i>
6.0	2017-08-17	2017-09-01 Commit <a href="#">397c036fc61</a>	2020-04-01	2020-11-25	(5.3) 5.4 - 7.4 <b>NOT PHP 8.0</b>
6.1	2019-09-01 According to commit <a href="#">b4dea50fd24</a>	2020-05-24	-	-	5.6 - 8.0 (From >6.1.1)

## Inline documentation

Autogenerated docs can be found, daily, at <https://gitreport.tornevall.net/tornelib-php-netcurl-6.1/>.

## Compatibility

To keep compatibility with v6.0 the plan is to keep the primary class MODULE\_CURL callable from a root position. It will probably be recommended to switch over to a PSR friendly structure from there, but the base will remain in 6.1 and the best way to instantiate the module in future is to call for the same wrapper as the main MODULE\_CURL will use - NetWrapper (TorneLIB\Module\Network\NetWrapper) as it is planned to be the primary driver handler.

## Project status

### v6.1.5

Key Summary T Updated Status

No issues found

### v6.1.4

Key Summary T Updated Status

NETCURL-332 [Walk through new inspections.](#)  Jun 22, 2021 **DONE**

NETCURL-329	Multicurl do not support several identical requests	<input checked="" type="checkbox"/>	May 27, 2021	DONE
NETCURL-328	Look up PATCH support.	<input checked="" type="checkbox"/>	May 27, 2021	DONE

3 issues

## Older releases

Key	Summary	T	Updated	Status
NETCURL-323	Timeout defaults should be flaggable and support millisec, also should native support for timeouts, visually, be passed through the configuration	<input checked="" type="checkbox"/>	Dec 09, 2020	DONE
NETCURL-324	Avoid using flaggables for timeouts in tests	<input checked="" type="checkbox"/>	Dec 09, 2020	DONE
NETCURL-320	setSignature overwriting	<input checked="" type="checkbox"/>	Dec 09, 2020	DONE
NETCURL-311	Simplify setCurlHeader again	<input checked="" type="checkbox"/>	Oct 22, 2020	DONE
NETCURL-305	Verify emptyness only	<input checked="" type="checkbox"/>	Oct 22, 2020	DONE
NETCURL-313	Bitbucket pipelines for PHP 8	<input checked="" type="checkbox"/>	Oct 22, 2020	DONE
NETCURL-312	resetCurlRequest() should not reset custom headers on demand	<input checked="" type="checkbox"/>	Oct 21, 2020	DONE
NETCURL-314	Default internal timeout must be higher than 8 as 4 for connection timeouts is too low	<input checked="" type="checkbox"/>	Oct 14, 2020	DONE
NETCURL-308	Change the way netcurl returns package version information	<input checked="" type="checkbox"/>	Sep 25, 2020	DONE
NETCURL-301	Use centralized version checker as most of the libraries are only compatible with 5.6 and above	<input checked="" type="checkbox"/>	Jul 15, 2020	DONE
NETCURL-263	Add errorhandler for multicurl	<input checked="" type="checkbox"/>	May 27, 2020	DONE
NETCURL-258	NetWrapper MultiRequest	<input checked="" type="checkbox"/>	May 27, 2020	DONE
NETCURL-285	Reinstate Environment but in ConfigWrapper to make wsdl transfers go non-cache vs cache, etc	<input checked="" type="checkbox"/>	May 20, 2020	DONE
NETCURL-295	MultiNetwrapper (+Soap)	<input checked="" type="checkbox"/>	May 12, 2020	DONE
NETCURL-278	setChain in 6.1 should throw errors when requested true	<input checked="" type="checkbox"/>	May 12, 2020	DONE
NETCURL-293	Try fix proper rss parsing without garbage	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-280	proxy support for curlwrapper and wrappers that is not stream wrappers	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-294	Make it possible to initialize an empty curlwrapper (without url)	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-292	Support basic rss+xml via GenericParser	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-261	Make sure setAuthentication is a required standard in the wrapper interface	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-260	Current curl implementation is only using GET and no advantage of Config	<input checked="" type="checkbox"/>	May 09, 2020	DONE

NETCURL-288	Output support for XML in simpler wrappers	<input checked="" type="checkbox"/>	May 09, 2020	DONE
NETCURL-291	SoapClient must be reinitialized each time it is called	<input checked="" type="checkbox"/>	Apr 27, 2020	DONE
NETCURL-231	Move out MODULE_NETWORK to own repo	<input checked="" type="checkbox"/>	Apr 27, 2020	DONE
NETCURL-287	Initialize simplified streamSupport	<input checked="" type="checkbox"/>	Apr 25, 2020	DONE
NETCURL-283	Use setSignature (?) to make requesting clients set internal clientname/version as userAgent automatically instead of Mozilla	<input checked="" type="checkbox"/>	Apr 25, 2020	DONE
NETCURL-277	Netwrapper Compatibility Service	<input checked="" type="checkbox"/>	Apr 25, 2020	DONE
NETCURL-289	Open for third party identification rather than standard browser agent	<input checked="" type="checkbox"/>	Apr 25, 2020	DONE
NETCURL-286	Move driver handler into own class	<input checked="" type="checkbox"/>	Apr 25, 2020	DONE
NETCURL-279	Make sure setOption is useful in NetWrapper and MODULE_CURL or work has been useless	<input checked="" type="checkbox"/>	Apr 22, 2020	DONE
NETCURL-234	Support immediate inclusions of network libraries	<input checked="" type="checkbox"/>	Apr 22, 2020	DONE
NETCURL-271	Synchronize with netcurl 6.0 test suites	<input checked="" type="checkbox"/>	Apr 22, 2020	DONE
NETCURL-247	The way we set user agent in the SSL module must be able to set in parents	<input checked="" type="checkbox"/>	Apr 22, 2020	DONE
NETCURL-209	Support stream, when curl is not an option	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-238	Make setTimeout support ms (curl_opt_timeout_ms)	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-259	High focus on curl (rebuild from 6.0)	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-268	Add Timeouts	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-245	Reimport soapclient	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-244	Reimport curl module	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-243	Reimport SSL helper	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-276	Use a natural soapcall with call_user_func_array	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-242	Disengage from constructor usage	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-264	Avoid static constants inside core functions	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-273	Support driverless environment	<input checked="" type="checkbox"/>	Apr 20, 2020	DONE
NETCURL-200	Confirm (by driver) that a driver is really available (update interface with requirements)	<input checked="" type="checkbox"/>	Apr 19, 2020	DONE
NETCURL-227	Migrate: getHttpHost()	<input checked="" type="checkbox"/>	Apr 19, 2020	DONE
NETCURL-265	getCurlException(\$curlHandle, \$httpCode) - httpCode is unused. Throw on >400	<input checked="" type="checkbox"/>	Apr 17, 2020	DONE
NETCURL-249	setAuth for curl	<input checked="" type="checkbox"/>	Apr 17, 2020	DONE

NETCURL-251	setAuth for soap	✓	Apr 17, 2020	DONE
NETCURL-269	Proxy support for stream_context	✓	Apr 10, 2020	DONE
NETCURL-267	On http head errors (>400) and non empty bodies	✓	Apr 08, 2020	DONE
NETCURL-255	Add static list of browsers for user-agent	✓	Mar 06, 2020	DONE
NETCURL-232	Make exceptions global	✓	Feb 18, 2020	DONE
NETCURL-225	NetCURL 6.1	⚡	Oct 11, 2021	DONE
NETCURL-325	Prevent objects to be pushed through multi-instance requests as urls	✖	Dec 09, 2020	DONE
NETCURL-319	setSignature may crash setUserAgent if array is sent into merger	✖	Dec 09, 2020	DONE
NETCURL-317	Version data falsely reports 6.1.0 when composer.json is missing	✖	Oct 25, 2020	DONE
NETCURL-316	Fetching version falsely returns wrong version	✖	Oct 21, 2020	DONE
NETCURL-302	getParsedResponse is used in rare cases (t-auth)	✖	Oct 14, 2020	DONE
NETCURL-309	Deprecated curldriver is missing proper PHP8 support	✖	Sep 25, 2020	DONE
NETCURL-310	Invisible exit code with unit70 for wsdlcache test	✖	Sep 13, 2020	DONE
NETCURL-304	memory exhaustion on lowest php memory limit (resources are no longer resources).	✖	Jul 24, 2020	DONE
NETCURL-246	Pipeline errors for PHP 7.3-7.4	✖	Apr 20, 2020	DONE
NETCURL-274	Cached wsdl requests and unauthorized exceptions	✖	Apr 20, 2020	DONE
NETCURL-272	getSoapEmbeddedRequest() - PHP 5.6+PHP 7.0	✖	Apr 20, 2020	DONE
NETCURL-226	PSR4 NetCURL+Network (Phase 1)	✖	Apr 19, 2020	DONE
NETCURL-230	Wordpress driver in prior netcurl is lacking authentication mechanisms	✖	Apr 03, 2020	DONE

67 issues

## Backstory

### History

NETCURL is a network library with a backstory in the simplicity to find and extract lists of ip-addresses from websites, utilize them and register them as proxies of different types. In short, NETCURL was once built for the [DNSBL API](#).

### Present time

NetCURL has changed purpose over time. What was planned to be scraping tool became, thanks to the self selective engine that was added, a tool to automatically fetch and parse data and transform it to standard output arrays/objects. The implementation supported common communications including SOAP and partially PHP built in streams though extra drivers - and it was autoselective. If the primary driver wasn't present, it jumped to next one and so on.

The intentions with NETCURL was probably never to rebuild the wheel. Many solutions already did this, but backfired somewhere when it came to setting the utilities up. Very often developers had to make all configurations manually (Both Guzzle and Zend does this), while NETCURL was firing up a high-verbose-level-fetching communications driver automatically. In our case, we wanted to utilize as much packages as possible and choose the best preferred method without asking question. The goal? Standard output, ready to be fetched by the endpoint developer.