

ZeroMQ on a Windows System 1.4V

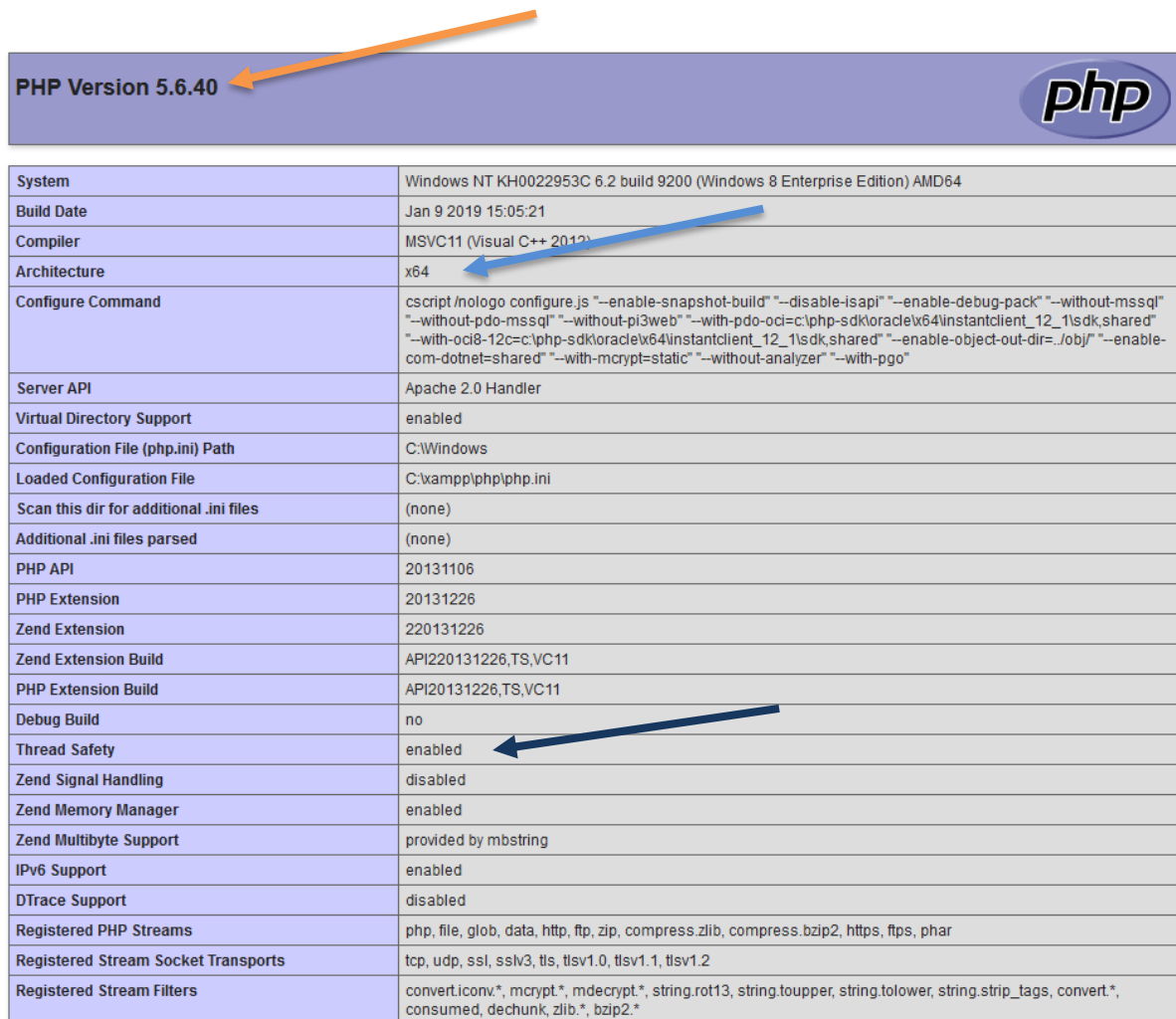
for all the guys who have trouble to install ZeroMQ on an Windows System.

ZeroMQ is not working with PHP-Versions above 7.2.x !!!

1: Server-Environment:

1.1 You have to check what version of PHP is installed on your Webserver:

You can find that information, when you open your PHP-Info-File:



The screenshot shows the PHP Info page. At the top left, it displays "PHP Version 5.6.40" with an orange arrow pointing to it. The PHP logo is on the top right. Below is a table of system information. A blue arrow points to the "Architecture" row, which shows "x64". Another blue arrow points to the "Thread Safety" row, which shows "enabled".

| System | Windows NT KH0022953C 6.2 build 9200 (Windows 8 Enterprise Edition) AMD64 |
|---|---|
| Build Date | Jan 9 2019 15:05:21 |
| Compiler | MSVC11 (Visual C++ 2012) |
| Architecture | x64 |
| Configure Command | cmd /c "cd /d %~dp0 & phpinfo.php" --enable-snapshot-build --disable-isapi --enable-debug-pack --without-mssql --without-pdo-mssql --without-pi3web --with-pdo-oci=c:\php-sdk\oracle\64\instantclient_12_1\sdk\shared --with-oci8-12c=c:\php-sdk\oracle\64\instantclient_12_1\sdk\shared --enable-object-out-dir=.obj --enable-com-dotnet-shared --with-mcrypt=static --without-analyzer --with-pgo |
| Server API | Apache 2.0 Handler |
| Virtual Directory Support | enabled |
| Configuration File (php.ini) Path | C:\Windows |
| Loaded Configuration File | C:\xampp\php\php.ini |
| Scan this dir for additional .ini files | (none) |
| Additional .ini files parsed | (none) |
| PHP API | 20131106 |
| PHP Extension | 20131226 |
| Zend Extension | 220131226 |
| Zend Extension Build | API220131226,TS,VC11 |
| PHP Extension Build | API20131226,TS,VC11 |
| Debug Build | no |
| Thread Safety | enabled |
| Zend Signal Handling | disabled |
| Zend Memory Manager | enabled |
| Zend Multibyte Support | provided by mbstring |
| IPv6 Support | enabled |
| DTrace Support | disabled |
| Registered PHP Streams | php, file, glob, data, http, ftp, zip, compress.zlib, compress.bzip2, https, ftps, phar |
| Registered Stream Socket Transports | tcp, udp, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2 |
| Registered Stream Filters | convert.iconv.*, mdecrypt.*, mdecrypt.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, zlib.*, bzip2.* |

In my case the PHP-Version: 5.6.40

1.2 Under which Architecture the Server is running:

In my case the Server is running in the x64 Architecture (64bit)

1.3 Is the Server running under Thread Safety Conditions:

In my case: enable

2. Microsoft Visual C++ Version

ZeroMQ needs Microsoft Visual C++ Runtimes to run correctly.

If you have a x64 Server running, it's necessary to have both Runtimeversions (x32 and x64) installed on the Server!!!

These are:

- Microsoft Visual C++ 2008 (VC9)
- Microsoft Visual C++ 2010 (VC10)
- Microsoft Visual C++ 2012 (VC11)
- Microsoft Visual C++ 2013 (VC12)
- Microsoft Visual C++ 2015 (VC14)
- Microsoft Visual C++ 2017 (VC15)

I made it easy and installed all. After that, don't forget to reboot.

3. What ZeroMQ version I need:

Here you can download the ZeroMQ Package:

<https://windows.php.net/downloads/pecl/releases/zmq/>

- 1.1.1 = PHP Version 5.3.x – 5.5.x
- 1.1.2 = PHP Version 5.3.x – 5.6.x
- 1.1.3 = PHP Version 7.0.x – 7.2.x

In the name of the file TS = Thread Save
 NTS = None Thread Save

- VC9 means: Microsoft Visual C++ 2008 (VC9)
- VC11 means: Microsoft Visual C++ 2012 (VC11)
- VC14 means: Microsoft Visual C++ 2015 (VC14)
- VC15 means: Microsoft Visual C++ 2017 (VC15)

derived therefrom I need the following package:

- PHP- Version = 5.6.40
- Architecture = X64
- Thread Save = Enable

[php_zmq-1.1.2-5.6-ts-vc11-x64.zip](#)

I must have installed Microsoft Visual C++ 2012 (VC11) in x32 and x64 Version

4. Installing

Download the right zip Archiv from the link above.

Unpack the following files to the folders:

File libzmq.dll must copied to your PHP folder

File php_zmq.dll must copied to your PHP-folder/ext

after that you must go to your PHP.ini and edit the following line:

Under the chapter: „Dynamic Extensions“ you must write the line:

extension=php_zmq.dll

```
php.ini - Editor
Datei Bearbeiten Format Ansicht ?
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; If you wish to have an extension loaded automatically, use the following
; syntax:
;
;   extension=modulename.extension
;
; For example, on Windows:
;
;   extension=mysql.dll
;
; ... or under UNIX:
;
;   extension=mysql.so
;
; ... or with a path:
;
;   extension=/path/to/extension/mysql.so
;
; If you only provide the name of the extension, PHP will look for it in its
; default extension directory.
;
; Windows Extensions
; Note that ODBC support is built in, so no dll is needed for it.
; Note that many DLL files are located in the extensions/ (PHP 4) ext/ (PHP 5)
; extension folders as well as the separate PECL DLL download (PHP 5).
; Be sure to appropriately set the extension_dir directive.
;
extension=php_zmq.dll
extension=php_bz2.dll
extension=php_curl.dll
extension=php_fileinfo.dll
extension=php_gd2.dll
extension=php_gettext.dll
;extension=php_gmp.dll
;extension=php_intl.dll
;extension=php_imap.dll
;extension=php_interbase.dll
;extension=php_ldap.dll
extension=php_mbstring.dll
extension=php_exif.dll ; Must be after mbstring as it depends on it
extension=php_mysql.dll
extension=php_mysqli.dll
```

Sometimes it would be necessary, that you write the following line in your http.conf to load the .dll in the Apache Webserver.

LoadFile "C:\server-path\php\libzmq.dll"



```
httpd.conf - Editor
Datei Bearbeiten Format Ansicht ?
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80

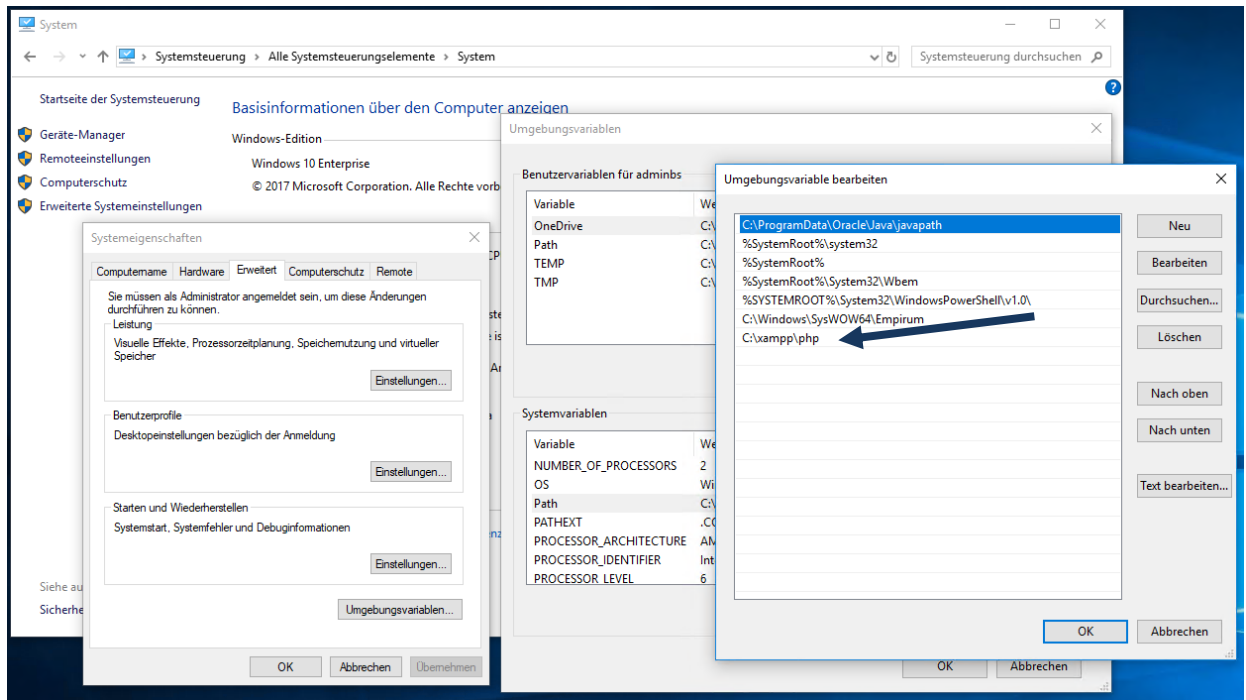
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadFile "C:\xampp\php\libzmq.dll"
LoadModule access_compat_module modules/mod_access_compat.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule allowmethods_module modules/mod_allowmethods.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
#LoadModule auth_digest_module modules/mod_auth_digest.so
#LoadModule auth_form_module modules/mod_auth_form.so
#LoadModule authn_anon_module modules/mod_authn_anon.so
LoadModule authn_core_module modules/mod_authn_core.so
#LoadModule authn_dbd_module modules/mod_authn_dbd.so
#LoadModule authn_dbm_module modules/mod_authn_dbm.so
LoadModule authn_file_module modules/mod_authn_file.so
#LoadModule authn_socache_module modules/mod_authn_socache.so
#LoadModule authnz_fcgi_module modules/mod_authnz_fcgi.so
#LoadModule authnz_ldap_module modules/mod_authnz_ldap.so
LoadModule authz_core_module modules/mod_authz_core.so
#LoadModule authz_dbd_module modules/mod_authz_dbd.so
#LoadModule authz_dbm_module modules/mod_authz_dbm.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_host_module modules/mod_authz_host.so
#LoadModule authz_owner_module modules/mod_authz_owner.so
```

After the last step, you must go to your

Control Panel
System
System Properties
Environment Variables
System Variables

Under the Key: „Path“ you must write down the path to your PHP-Directory

In my case: c:\server-path\php



Finally you must reboot your Windows Server and do the Xibo-System-Test again.

Hopefully the ZeroMQ Extension is successful installed.

5. Configure the XMR Service under a Windows system

Please go to:

In your Webroot-Folder\xibo\vendor\xibosignage\xibo-xmr

In my case: C:\xampp\htdocs\xibo\vendor\xibosignage\xibo-xmr

and create an empty textfile

write in that textfile:

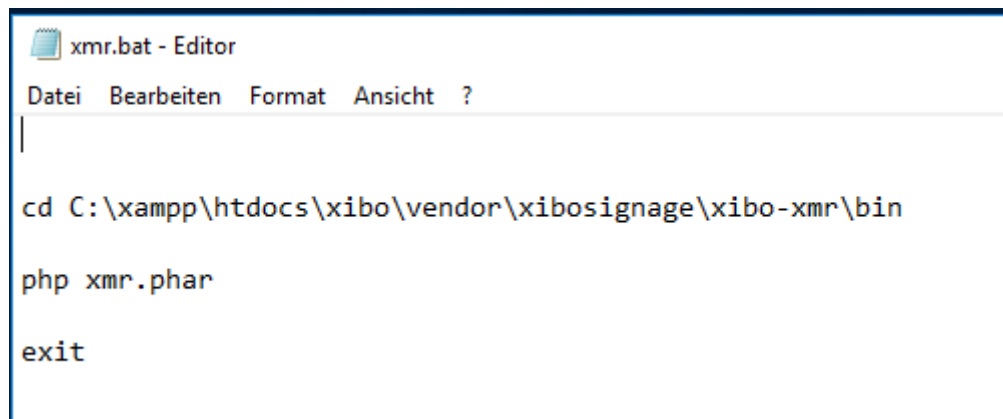
```
{  
  "listenOn": "tcp://127.0.0.1:50001",  
  "pubOn": ["tcp://Webserver_IP:9505"],  
  "debug": true  
}
```

Save and close it.

Rename it to: config.json

Now you must call the xmr.phar to start the XMR-Service.

In my case I have a Batch-Script to start the XMR-Service on Systemstart:



```
xmr.bat - Editor  
Datei Bearbeiten Format Ansicht ?  
  
cd C:\xampp\htdocs\xibo\vendor\xibosignage\xibo-xmr\bin  
  
php xmr.phar  
  
exit
```

Its important that the script is always running. If it don't run, XMR don't work!!!

6. Configure XMR in Xibo

Login in Xibo and go to : Settings -> Displays:

under XMR Private Address : tcp://127.0.0.1:50001

Under XMR Public Address: tcp://Webserver_IP:9505

Save

Go to: Display Settings: Chose your Display-Profile that you use and chose "edit" on the Menu

under XMR Public Address you write: tcp://Webserver_IP:9505

save

Go to: Displays and push the Button "Column visibility" an chose "Xmr-Registered" to display that your Xibo-Player received XMR-Commandos.

In Displays in menu chose: "Edit" and than got to "Advanced" and make a check on "reconfigure XMR" and save it.

After that restart your Player.

Hopefully you get a hook at "XMR-Registerd" in Displays an so you can send commands and recieved screenshots from the player.

Greetings from Germany

Torsten