

Windows 端编译 tonyenc 扩展：以 Win10+PHP7.4.27 为例

一、准备工作

1、安装 Visual Studio

php7.0 7.1 版本依赖 vc14 需要提前安装 visual studio 2015

php7.2 7.3 7.4 版本依赖 vc15 需要提前安装 visual studio 2017

php8.0 8.1 版本依赖 vs2016 需要提前安装 visual studio 2019

各版本 Visual Studio 需要安装的模块，见文章第三部分，如果已经安装的 vs 但是缺少需要的模块或者漏选了模块，不要着急，重新运行对应版本 vs 的安装程序，选择修改，继续添加所需模块进行安装即可。

注意：本人用的 win10 操作系统所以 sdk 选的是 win10 的，如果是其他版本的 windows，选择对应的 sdk 安装即可

2、下载 php 的 windows 源码包

windows 源码包下载地址：<https://windows.php.net/downloads/releases/archives/>

以 php7.4.27 为例（php-7.4.27-src.zip）

3、下载 windows 端构建 PHP 的工具包：

①使用 git：git clone <https://gitee.com/lfveeker/php-sdk-binary-tool.git>

②浏览器打开链接：<https://gitee.com/lfveeker/php-sdk-binary-tool> 选择 zip 压缩包下载

二、开始构建 Windows 端的 tonyenc 扩展

1、运行 CMD 并将目录切换到工具包的根目录

```
Microsoft Windows [版本 10.0.19043.1415]
(c) Microsoft Corporation。保留所有权利。

D:\DevTools\php-sdk-binary-tool>
```

2、运行对应版本的 bat 文件，此处为 php7.4.27 编译 64 位版本扩展，需要运行 phpsdk-vc15-x64.bat，出现下图所示结果说明正常否则说明对应版本所需的 Visual Studio 未正确安装。

```
D:\DevTools\php-sdk-binary-tool>phpsdk-vc15-x64.bat
[vcvarsall.bat] Environment initialized for: 'x64'

PHP SDK 2.2.0

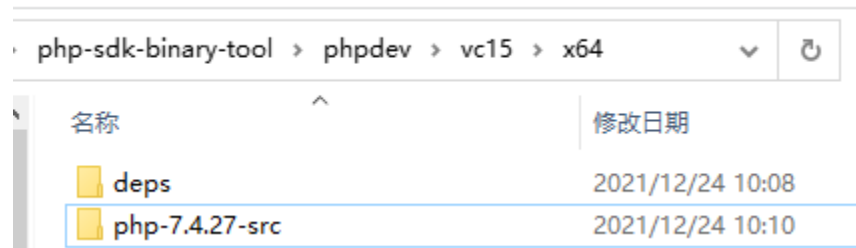
OS architecture:      64-bit
Build architecture:   64-bit
Visual C++:           14.16.27045.0
PHP-SDK path:          D:\DevTools\php-sdk-binary-tool

D:\DevTools\php-sdk-binary-tool
$
```

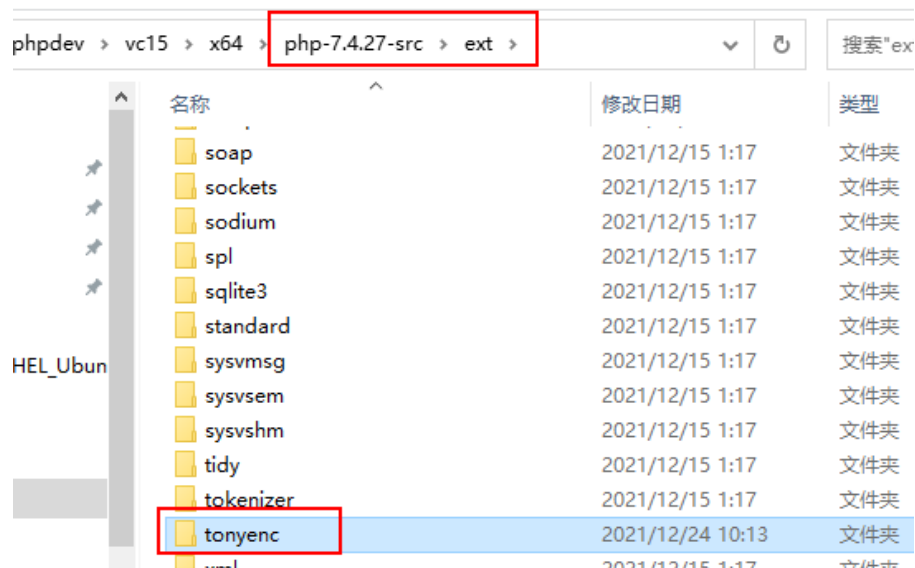
- 3、运行 `phpsdk_buildtree phpdev`，成功后会在工具包的根目录创建如下（`phpdev\vc15\x64`）结构的目录并自动切换到该目录下

```
D:\DevTools\php-sdk-binary-tool\phpdev\vc15\x64
$
```

- 4、将 `php` 源码包复制到上一步操作生成的目录下并解压，与 `deps` 目录平级。



- 5、将 `tonyenc` 扩展源码复制到 `php` 源码包目录下的 `ext` 目录，结构如下：



- 6、回到 `cmd` 窗口，将路径切换到 `php` 源码包目录下并执行 `buildconf`

```
D:\DevTools\php-sdk-binary-tool\phpdev\vc15\x64
$ cd php-7.4.27-src
D:\DevTools\php-sdk-binary-tool\phpdev\vc15\x64\php-7.4.27-src
$ buildconf
Rebuilding configure.js
Now run 'configure --help'
D:\DevTools\php-sdk-binary-tool\phpdev\vc15\x64\php-7.4.27-src
$
```

7、执行 `configure --help`，如果出现如下提示，再次执行一遍 `configure --help` 即可

```
Options that enable extensions and SAPI will accept 'yes' or 'no' as a
parameter. They also accept 'shared' as a synonym for 'yes' and request a
shared build of that module. Not all modules can be built as shared modules;
configure will display [shared] after the module name if can be built that
way.
```

8、`configure --help` 的结果中如果出现下图信息，则说明 `tonyenc` 扩展可用

```
--disable-session          session support
--enable-shmop              shmop support
--with-snmp                 SNMP support
--enable-sockets            SOCKETS support
--with-sodium               for libsodium support
--with-sqlite3              SQLite 3 support
--with-password-argon2      Argon2 support
--with-config-file-scan-dir Dir to check for additional php ini files
--enable-sysvshm            SysV Shared Memory support
--with-tidy                 TIDY support
--disable-tokenizer         tokenizer support
--enable-tonyenc             tonyenc support
--enable-zend-test          enable zend-test extension
--disable-zip               ZIP support
--disable-zlib              ZLIB support
--without-libxml            LibXML support
--without-dom               DOM support
--enable-exif               Exchangeable image information (EXIF)
                           Support
--with-mysqli               MySQLi support
```

9、配置并编译生成 `dll` 文件

NTS 版本扩展配置命令：`configure --disable-all --enable-cli --enable-tonyenc=shared --disable-zts`

TS 版本扩展配置命令：`configure --disable-all --enable-cli --enable-tonyenc=shared`

①构建 NTS 版 `dll`：

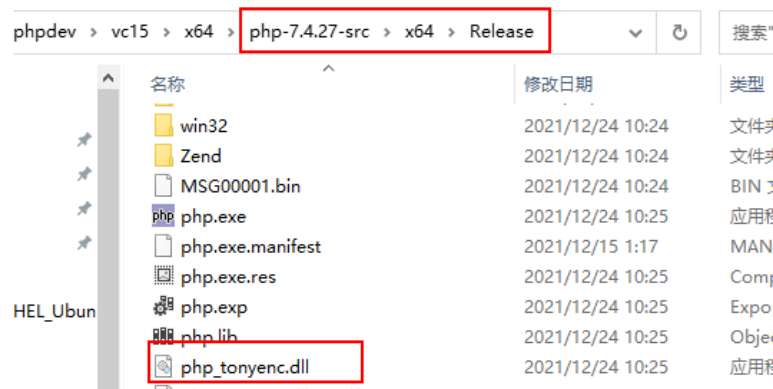
运行：`configure --disable-all --enable-cli --enable-tonyenc=shared --disable-zts`

```
Enabled extensions:
+-----+-----+
| Extension | Mode |
+-----+-----+
| date      | static |
| hash      | static |
| pcre      | static |
| reflection | static |
| spl       | static |
| standard  | static |
| tonyenc   | shared |
+-----+-----+

Enabled SAPI:
+-----+
| Sapi Name |
+-----+
| cli       |
+-----+

+-----+-----+
| Build type | Release |
+-----+-----+
| Thread Safety | No |
| Compiler      | Visual C++ 2017 |
| Architecture  | x64 |
| Optimization  | PGO disabled |
| Native intrinsics | SSE2 |
| Static analyzer | disabled |
+-----+-----+
```

执行 nmake，成功后会在源码包如下位置生成 dll 文件

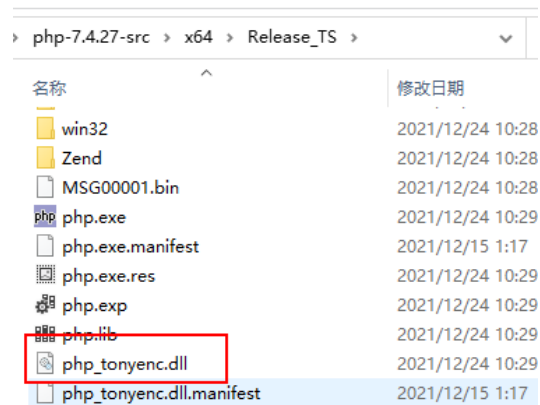


②构建 TS 版 dll，依次执行：

`configure --disable-all --enable-cli --enable-tonyenc=shared`

`nmake`

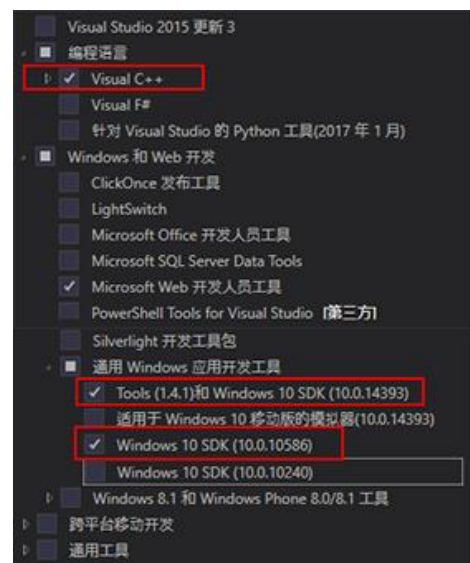
会在如下位置生成 TS 版的 dll 文件



如果编译 32 位的，则生成目录就为少了 x64 这层目录，直接在源码包根目录下产生 Release 与 Release_TS 目录

三、各版本 Visual Studio 所需模块

VS2015 所需模块：



VS2017 所需模块:

▼ 使用 C++ 的桌面开发

▼ 已包含

- ✓ Visual C++ 核心桌面功能

▼ 可选

- ☒ 实时调试器
- ☒ VC++ 2017 version 15.9 v14.16 latest v141...
- ☒ C++ 分析工具
- ☒ Windows 10 SDK (10.0.17763.0)
- ☒ 用于 CMake 的 Visual C++ 工具
- ☒ 用于 x86 和 x64 的 Visual C++ ATL
- ☒ 用于 Boost.Test 的测试适配器
- ☒ Google Test 测试适配器
- ☒ IntelliTrace
- ☐ Windows 8.1 SDK 和 UCRT SDK
- ☐ 对 C++ 的 Windows XP 支持
- ☐ 用于 x86 和 x64 的 Visual C++ MFC
- ☒ C++/CLI 支持
- ☐ 标准库模块(实验性)
- ☐ IncrediBuild - 生成加速
- ☐ Windows 10 SDK (10.0.17134.0)
- ☐ Windows 10 SDK (10.0.16299.0)
- ☐ Windows 10 SDK (10.0.15063.0)
- ☐ Windows 10 SDK (10.0.14393.0)
- ☐ Windows 10 SDK (10.0.10586.0)
- ☐ Windows 10 SDK (10.0.10240.0)
- ☐ 适用于桌面的 VC++ 2015.3 v14.00 (v140)...

▼ 通用 Windows 平台开发

▼ 已包含

- ✓ Blend for Visual Studio
- ✓ .NET Native 和 .NET Standard
- ✓ NuGet 包管理器
- ✓ 通用 Windows 平台工具
- ✓ Windows 10 SDK (10.0.17763.0)

▼ 可选

- ☒ IntelliTrace
- ☐ .NET Core 2.1 SDK (out of support)
- ☐ USB 设备连接性
- ☐ C++ 通用 Windows 平台工具
- ☒ 用于 DirectX 的图形调试器和 GPU 探查器
- ☐ Windows 10 移动仿真程序(Fall Creators U...
- ☐ Windows 10 SDK (10.0.17134.0)
- ☐ Windows 10 SDK (10.0.16299.0)
- ☐ Windows 10 SDK (10.0.15063.0)
- ☐ Windows 10 SDK (10.0.14393.0)
- ☐ Windows 10 SDK (10.0.10586.0)
- ☐ Windows 10 SDK (10.0.10240.0)
- ☐ 体系结构和分析工具

VS2019 所需模块:

