

# ZoeeyGuard 用户手册

moxie(system128@gmail.com)

出版日期 2012-03-30

---

# 目录

项目介绍 .....	iii
1. 使用须知 .....	1
2. 快速入门 .....	2
3. 使用说明 .....	4
3.1. 函数说明 .....	4
zoeey_encode .....	4
zoeey_run .....	4
3.2. 常规定制 .....	5
加密定制 .....	5
设置使用期限 .....	6
其他定制 .....	6
4. 常见问题 .....	7

---

# 项目介绍

ZoeeyGuard 是一个PHP源码加密保护扩展。由商业收费代码简化而来。结构简单清晰，具有很强的可定制性。

---

# 第 1 章 使用须知

## 目录说明

目录	说明
/src	zoeeyguard扩展源码（c）
/example	加密脚本源码（php）
/doc	文档源码
/doc/build	已编译的文档

## 使用流程

1. 编译安装zoeeyguard扩展（/src）。
2. 使用加密函数对源码进行加密。可直接使用 /example 目录下的加密脚本。
3. 部署加密后的脚本，运行。

## 第 2 章 快速入门

### 扩展安装

```
root# cd /path/to/php/src/ext/zoeeyguard
root# /path/to/php/bin/phpize
root# ./configure --with-php-config=/path/to/php/bin/php-config
root# make install
root# make
root# cp modules/zoeeyguard.so /path/to/php/lib # (extension_dir)
root# echo 'extension=zoeeyguard.so' >> /path/to/php.ini
root# service httpd restart
```

#### zoeeyguard

zoeeyguard support	enabled
version	1.0
site	<a href="http://code.google.com/p/zoeeyguard/">http://code.google.com/p/zoeeyguard/</a>
build date	Mar 14 2012 14:30:55

图 2.1. phpinfo()

### 加密与运行

这里使用示例加密脚本。

```
<?php

/**
 * 这是一个示例
 */
echo '<pre>';
echo date('Y-m-d H:i:s'), PHP_EOL;
$names = get_extension_funcs('zoeeyguard');
foreach ($names as $name) {
    echo $name, PHP_EOL;
}
echo '</pre>';
?>
```

例 2.1. example.php 加密前

```
<?php
$code = file_get_contents('example.php');
$code = zoeey_encode($code);
$code = sprintf('<?php
zoeey_run(\'%s\');
?>', $code);
?>
```

例 2.2. 加密脚本主要片段

```
<?php
/* 2012-03-14 21:09:57 */
zoeey_run('2Xr6Jcrn5GT9Kla86c30w7IjUuQb/NAIBChIrEJ+v+IMCmz3UsenWjHMWm2bsN4j
JNDwrCTb21dcckseE8dn031PrcHdE0snEcFaS/w1583Pv1r0S/QiOMTA2AhTouJiZ0La0/TpUY
suSE2YB/n3Q0VY2CVpUY0tvMeyr7eP/CQ1JEeHUHhFkmhfRss86ErzwNhcwukiJc38S0L7wRqG
SNz427wTWDmi');
?>
```

例 2.3. example.php 加密后



### 警告

实际加密代码中并无换行，这里换行仅为适应格式。

---

# 第 3 章 使用说明

## 3.1. 函数说明

### zoeey\_encode

加密。

#### 描述

```
string zoeey_encode(  
    string  
  
    $code  
  
    = NULL  
);
```

对代码进行加密。

#### 参数

code

源代码。

#### 返回值

返回密文内容。

### zoeey\_run

运行。

#### 描述

```
bool zoeey_run(  
    string  
  
    $code  
  
    = NULL  
);
```

运行密文。

#### 参数

code

密文。

#### 返回值

执行是否成功。

## 3.2. 常规定制

大部分的定制功能可以在PHP源码内完成，如限制使用者域名，强制输出。

### 加密定制

ZoeeyGuard 基础版本提供三步数据加密

1. 数据位置交换（OBFUSCATED\_ORDER）。
2. 使用密钥求异或（PRIVATE\_KEY）。
3. 这一步可供自定义加密。下面介绍几个常用的加密方式。

RC4: <http://en.wikipedia.org/wiki/RC4>

XXTEA: <http://en.wikipedia.org/wiki/XXTEA>

DES: [http://en.wikipedia.org/wiki/Data\\_Encryption\\_Standard](http://en.wikipedia.org/wiki/Data_Encryption_Standard)

4. 乱序base64打包（OBFUSCATED\_ALPHABET）。

```
/* guard.h 内定制私有密钥 */

#define PRIVATE_KEY "28dsa7dsas12312389uy7aydh8h1h2i312"

/* 修改字符串即可 */
```

#### 例 3.1. 定制私有密钥

```
/* guard.h 内定制位置交换规律 */

static const unsigned char OBFUSCATED_ORDER[] = {
    13,  6,  5,  7,  1, 15, 14, 20
    ,  9, 16, 19,  4, 18, 10,  2,  8
    , 12,  3, 11,  0, 17
};
/* 这里提供0~20, 21个数字的乱序集合。可新增和打乱该数组以改变位置交换规律 */
```

#### 例 3.2. 定制位置交换规律

```
/* guard.h 内定制base64字母表 */

static const unsigned char OBFUSCATED_ALPHABET[] = {
    's', '4', 'N', 'E', 'k', 'X', 'c', 'u'
    , 'J', '2', 'U', 'o', 'O', 'w', 'K', 'v'
    , 'h', 'H', 'C', '/', 'D', 'q', 'l', 'R'
    , 'B', 'r', '5', 'Z', 'S', 'Q', '6', 'W'
    , '3', 'L', 'j', '8', '1', 'z', '0', 'G'
    , 'n', 'e', 'y', 'b', 'I', 'd', 'i', 'P'
    , 'A', '9', '7', '+', 'm', 'V', 'M', 'Y'
    , 'F', 'g', 'f', 'p', 'a', 'T', 't', 'x'
};

/* 打乱该数组以改变base64实现 */
```

#### 例 3.3. 定制乱序base64字母表



## 设置使用期限

```
/* zoeeyguard.c 中通过当前UNIX时间戳判断代码是否过期 */

#include <time.h>

/* 放在 zoeey_run 内，处理参数前。 */
long now = (long)time(NULL);

if (now > 123456789){
    php_printf("expired");
    return;
}
```

例 3.4. 使用时间定制

## 其他定制

机器码也可作为加密密匙的一部分。常规机器码，如CPU编号、硬盘序列、U盘序列、加密狗序列……

---

## 第 4 章 常见问题

### 为什么源码写的这么烂？

这是从商业代码精简整理来的，起初写的时候鄙人技术甚差（现在更差\_#）。多有不足，请君斧正。

### 应该在什么情况下使用它，不是倡导开源呢么？

首先鉴于windows下编译PHP扩展非常麻烦，建议您在unix\*下使用。

用来隐蔽敏感代码，如支付接口的配置文件，域名注册接口的配置文件。

当然，如果您是商业代码，也可以用来保护产品源码。

倡导开源和对源码进行加密并不冲突。

对敏感代码进行加密可以在一定程度上保障程序的运行安全，保护开发者和使用者的基本利益。

### 加密效果如何，是否很容易被破解？

在源码和扩展库没有泄露的情况下很难破解。可大幅增加破解难度,但并不保证完全无法破解。

### 这个开源版本和以前的商业版本有什么不同？

商业版本加密和解密是分离的，加密使用独立的客户端程序。解密使用独立扩展，在扩展内不存在加密方法。

商业版本和本开源免费版本安全性上并无太大区别。

开源版本定制性强，可以根据自己的需要进行扩展。生成脚本使用PHP实现，操作灵活度大大增强。

开源版本更为强大，但是用复杂度也有所增加。

### 为什么开源了？

客户要求提供多平台编译版本，操作成本高昂。定制要求多变，调试成本过高。