

Custom php Introspection for 0-Day Research

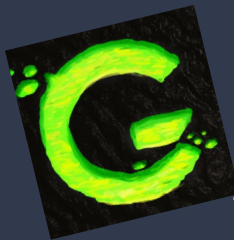


GREHACK

New is **still not** always better.

1. Whoami
2. Introspection 101
3. Php-Internalog & Iterations
4. Pentest, Méthodology, & Findings
5. Kudos

@Groumpf_ & @TheLaluka
thinkloveshare.com 



1. Whoami



@TheLaluka



@TheLaluka



@Groumpf_

2. Introspection 101

What is Introspection ?

Information about Objects, Functions, Classes, Types, Properties...

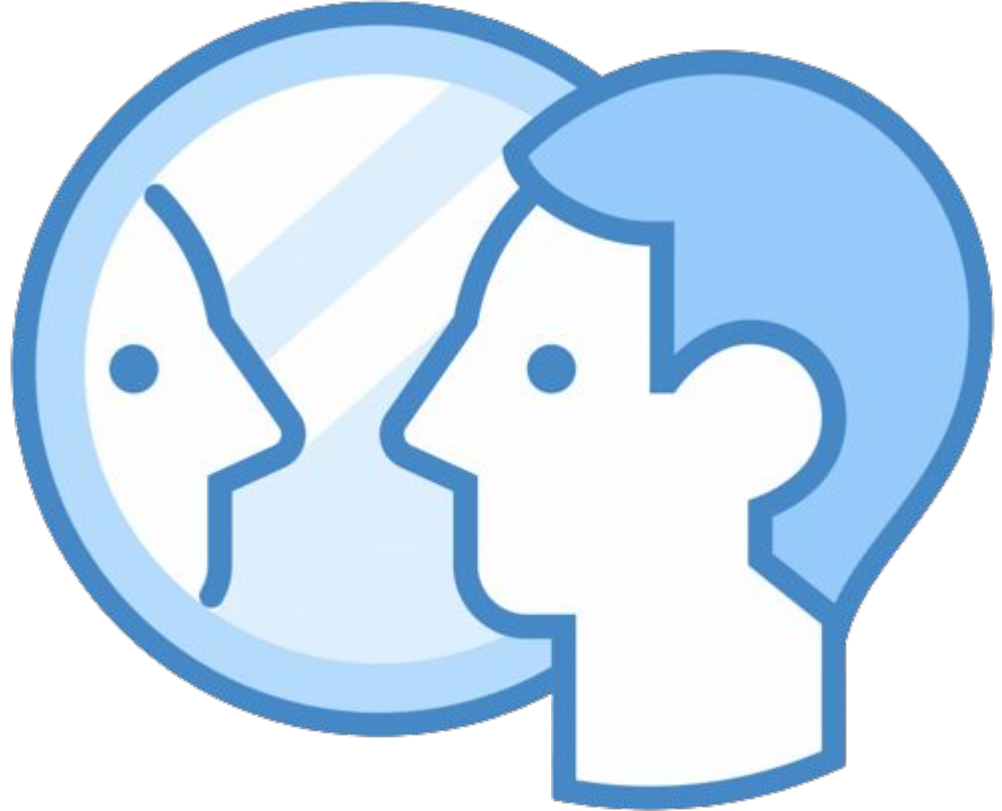
- All at runtime

Python

- `help`, `dir`, `hasattr`, `type`

PHP

- `class_exists`, `get_class`,
`get_parent_class`,
`is_subclass_of`



Why Introspection ?

- Debug -> ipdb
- Optimisation -> perf
- Fuzzing -> burp infiltrator
- Security checks -> sqreen



What do we want?

What do we want?

Need.

To.

Know.

WTF.

Is.

Going.

On.

What do we want?

“Log every ‘interesting’ function,
its name, parameters, and values”

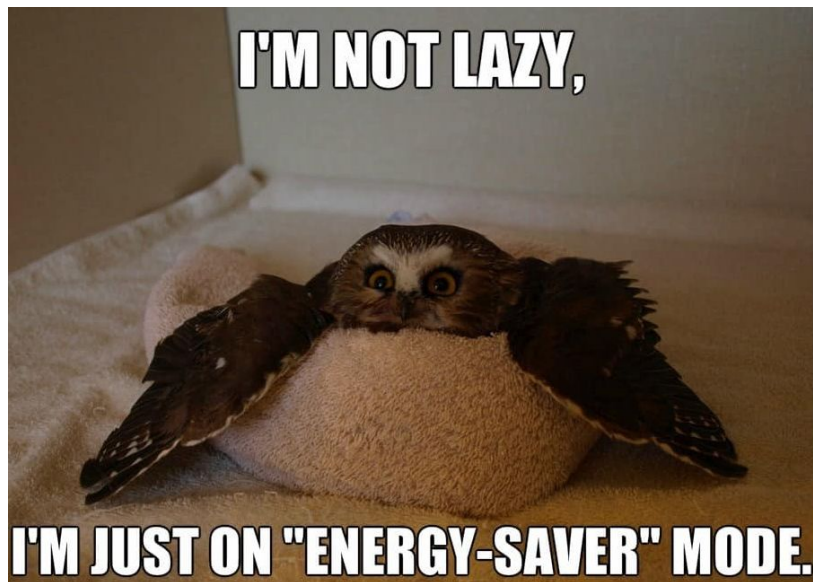


3. Php-Internallog & Iterations

Vo – ltrace, strace, LD_PRELOAD

```
<?php
assert_options(ASSERT_ACTIVE, 1); // On active les assertions
phpinfo();
$handle = popen("/usr/bin/whoami", "r");
$read = fread($handle, 256);
pclose($handle);
shell_exec('ls');
exec("whoami");
system("whoami");
passthru("whoami");
$test = 1;
assert('$test == 1');
$string = 'The quick brown fox jumps over the lazy dog.';
$patterns = array();
$patterns[0] = '/quick/';
$patterns[1] = '/brown/';
$patterns[2] = '/fox/';
$replacements = array();
$replacements[2] = 'bear';
$replacements[1] = 'black';
$replacements[0] = 'slow';
echo preg_replace($patterns, $replacements, $string);
copy("not_a_file", "not_a_file_either");
$descriptorspec = [STDIN, STDOUT, STDOUT];
$cmd = "whoami";
$proc = proc_open($cmd, $descriptorspec, $pipes);
proc_close($proc);
eval ("echo \"test \\n\\n\";");
include 'include.php';
include_once 'include.php';
require 'include.php';
require_once 'include.php';
```

```
ltrace -f /usr/local/bin/php samples/index.php
strace -f /usr/local/bin/php samples/index.php
```



Vo - ltrace, strace, LD_PRELOAD

```
[pid 151574] --- Called exec() ---
[pid 151574] +++ exited (status 0) +++
[pid 151573] --- SIGCHLD (Child exited) ---
[pid 151573] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
[pid 151575] --- Called exec() ---
[pid 151576] --- Called exec() ---
[pid 151576] +++ exited (status 0) +++
[pid 151575] --- SIGCHLD (Child exited) ---
[pid 151575] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
[pid 151577] --- Called exec() ---
[pid 151578] --- Called exec() ---
[pid 151578] +++ exited (status 0) +++
[pid 151577] --- SIGCHLD (Child exited) ---
[pid 151577] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
[pid 151579] --- Called exec() ---
[pid 151580] --- Called exec() ---
lalu
[pid 151580] +++ exited (status 0) +++
[pid 151579] --- SIGCHLD (Child exited) ---
[pid 151579] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
[pid 151581] --- Called exec() ---
[pid 151582] --- Called exec() ---
lalu
[pid 151582] +++ exited (status 0) +++
[pid 151581] --- SIGCHLD (Child exited) ---
[pid 151581] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
The bear black slow slumps over the lazy dog.
Warning: copy(not_a_file): failed to open stream: No such file or directory in /opt/php7.4.26-internalog/
[pid 151583] --- Called exec() ---
[pid 151584] --- Called exec() ---
lalu
[pid 151584] +++ exited (status 0) +++
[pid 151583] --- SIGCHLD (Child exited) ---
[pid 151583] +++ exited (status 0) +++
[pid 151571] --- SIGCHLD (Child exited) ---
test
me-iz-includedme-iz-included[pid 151572] +++ exited (status 0) +++
[pid 151571] +++ exited (status 0) +++
-----
/opt/php7.4.26-internalog (feature/threaded_udp_requests) » ltrace -f /usr/local/bin/php samples/index.php
```

```
[pid 153601] munmap(0x7fadad063000, 116494) = 0
[pid 153601] openat(AT_FDCWD, "/etc/passwd", O_RDONLY|O_CLOEXEC) = 4
[pid 153601] lseek(4, 0, SEEK_CUR) = 0
[pid 153601] fstat(4, {st_mode=S_IFREG|0644, st_size=2949, ...}) = 0
[pid 153601] read(4, "\root:x:0:0:root:/root:/bin/bash\n...", 4096) = 2949
[pid 153601] close(4) = 0
[pid 153601] fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...}) = 0
[pid 153601] write(1, "\lalu\n", 5)lalu
) = 5
[pid 153601] close(1) = 0
[pid 153601] close(2) = 0
[pid 153601] exit_group(0) = ?
[pid 153601] +++ exited with 0 +++
[pid 153600] <... wait4 resumed>[WIFEXITED(s) && WEXITSTATUS(s) == 0]; 0, NULL) = 153601
[pid 153600] --- SIGCHLD {s1_signo=SIGCHLD, s1_code=CLD_EXITED, s1_pid=153601, s1_uid=1000, s1_status=0, s1_utime=0, s1_
[pid 153600] rt_sigreturn(mask=[]) = 153601
[pid 153600] exit_group(0) = ?
[pid 153600] +++ exited with 0 +++
[pid 153588] <... wait4 resumed>[WIFEXITED(s) && WEXITSTATUS(s) == 0]; 0, NULL) = 153600
[pid 153588] --- SIGCHLD {s1_signo=SIGCHLD, s1_code=CLD_EXITED, s1_pid=153600, s1_uid=1000, s1_status=0, s1_utime=0, s1_
[pid 153588] write(1, "test \n", 6)test
) = 6
[pid 153588] getcwd("/opt/php7.4.26-internalog", 4096) = 26
[pid 153588] lstat("/opt/php7.4.26-internalog/.include.php", 0x7ffffeaa3830) = -1 ENOENT (No such file or directory)
[pid 153588] lstat("/usr/local/lib/php/include.php", 0x7ffffeaa3830) = -1 ENOENT (No such file or directory)
[pid 153588] lstat("/opt/php7.4.26-internalog/samples/include.php", {st_mode=S_IFREG|0664, st_size=47, ...}) = 0
[pid 153588] openat(AT_FDCWD, "/opt/php7.4.26-internalog/samples/include.php", O_RDONLY) = 4
[pid 153588] fstat(4, {st_mode=S_IFREG|0664, st_size=47, ...}) = 0
[pid 153588] read(4, "php\n// include test\necho \"me-i\"... , 47) = 47
[pid 153588] close(4) = 0
[pid 153588] write(1, "me-iz-included", 14)me-iz-included
) = 14
[pid 153588] getcwd("/opt/php7.4.26-internalog", 4096) = 26
[pid 153588] lstat("/opt/php7.4.26-internalog/.include.php", 0x7ffffeaa3da0) = -1 ENOENT (No such file or directory)
[pid 153588] lstat("/usr/local/lib/php/include.php", 0x7ffffeaa3da0) = -1 ENOENT (No such file or directory)
[pid 153588] getcwd("/opt/php7.4.26-internalog", 4096) = 26
[pid 153588] lstat("/opt/php7.4.26-internalog/.include.php", 0x7ffffeaa3830) = -1 ENOENT (No such file or directory)
[pid 153588] lstat("/usr/local/lib/php/include.php", 0x7ffffeaa3830) = -1 ENOENT (No such file or directory)
[pid 153588] openat(AT_FDCWD, "/opt/php7.4.26-internalog/samples/include.php", O_RDONLY) = 4
[pid 153588] fstat(4, {st_mode=S_IFREG|0664, st_size=47, ...}) = 0
[pid 153588] read(4, "<?php\n// include test\necho \"me-i\"... , 47) = 47
[pid 153588] close(4) = 0
[pid 153588] write(1, "me-iz-included", 14)me-iz-included
) = 14
[pid 153588] getcwd("/opt/php7.4.26-internalog", 4096) = 26
[pid 153588] lstat("/opt/php7.4.26-internalog/.include.php", 0x7ffffeaa3da0) = -1 ENOENT (No such file or directory)
[pid 153588] lstat("/usr/local/lib/php/include.php", 0x7ffffeaa3da0) = -1 ENOENT (No such file or directory)
[pid 153588] close(0) = 0</pre
```

Vo – strace, ltrace, LD_PRELOAD*

- Fast setup
- Fast enough at runtime
- No code to maintain
- Always available
- Breaks sometimes for weird reasons
- Information is too “low level”
- Not really flexible nor configurable
- “All your ~~base~~ debuggers are belong to us”



V1 – fork php-7.4, UDP client & netcat



```
1  /* Setup server with : nc -lnvkup 8888 */
2  #include <netdb.h>
3  #define LALUKA_LOG(lalu_p_str_logme) {
4      int lalu_port = 8888;
5      char *lalu_ip = "127.0.0.1";
6      char *lalu_log_fmt = "LALUKA : %s\n";
7      int lalu_sock;
8      struct sockaddr_in lalu_server_addr;
9      struct hostent *lalu_host;
10     char *lalu_log_line;
11     ssize_t lalu_bufsz;
12
13     lalu_host = (struct hostent *)gethostbyname(lalu_ip);
14     lalu_bufsz = snprintf(NULL, 0, lalu_log_fmt,
15         lalu_p_str_logme);
16     lalu_log_line = malloc(lalu_bufsz + 1);
17     snprintf(lalu_log_line, lalu_bufsz + 1, lalu_log_fmt,
18         lalu_p_str_logme);
19     if ((lalu_sock = socket(AF_INET, SOCK_DGRAM, 0)) == -1) {
20         perror("socket");
21         exit(1);
22     }
23
24     lalu_server_addr.sin_family = AF_INET;
25     lalu_server_addr.sin_port = htons(lalu_port);
26     lalu_server_addr.sin_addr = *((struct in_addr *)
27         lalu_host->h_addr);
28     bzero(&(lalu_server_addr.sin_zero), 8);
29     sendto(lalu_sock, lalu_log_line, strlen(lalu_log_line), 0,
30         (struct sockaddr *)&lalu_server_addr, sizeof(struct
31         sockaddr));
32     close(lalu_sock);
33     free(lalu_log_line);
```

V1 - fork php-7.4, UDP client & netcat

```
149     ZEND_PARSE_PARAMETERS_START(1, 2)
150     |     Z_PARAM_ZVAL(assertion)
151     |     Z_PARAM_OPTIONAL
152     |     Z_PARAM_OBJ_OF_CLASS_OR_STR_OR_NULL(description_obj, zend_ce_throwable)
153     ZEND_PARSE_PARAMETERS_END();
154     char * log_fmt = "assert('%s')";
155     ssize_t bufisz = snprintf(NULL, 0, log_fmt, Z_STRVAL_P(assertion));
156     char* final_log = malloc(bufisz + 1);
157     snprintf(final_log, bufisz + 1, log_fmt, Z_STRVAL_P(assertion));
158     LALUKA_LOG(final_log);
159
160     if (zend_is_true(assertion)) {
161     |     RETURN_TRUE;
162     }
```


V1 – fork php-7.4, UDP client & netcat

- Hooking every function is Doable
- Actually logs useful stuff 🎉
- Garbage flooded no more
- Fast-ish, no RTT with UDP



- Hooking every function is PAINFUL
- Not everything is a function 🤔🤔🤔
- Backend is as dumb as a socket 🧦
- Heard about threads? 🧵
- Compiling.....



V2 - xDebug is all you need, but SLOW AF

X-debug one liner, no docker

<https://xdebug.org/docs/trace>

```
sudo apt install php-xdebug
php -d xdebug.trace_output_name=trace -d
xdebug.trace_options=1 -d display_startup_errors=1 -d
display_errors=1 -d xdebug.auto_trace=1 -d
xdebug.collect_params=4 -d xdebug.trace_format=0 -d
xdebug.trace_output_dir=/home/lalucloud/Spip/spip/traces/
-S 0.0.0.0:8000
tail -f trace.xt | unbuffer -p sed "s#\t# #g;
s#\n#\n#g; s#\s#\s#g" | grep -i eval -C 10
```



xDebug with php.ini, syntax for xDebug 2.X, no docker


```
cd /dev/shm
cat > php.php << EOF
<?php
    phpinfo();
?>
EOF
XDEBUG_S0=$(find /usr -name xdebug.so)
CONF=current.ini
cat > $CONF << EOF
display_errors=1
display_startup_errors=1
html_errors=1
xdebug.auto_trace=1
xdebug.collect_params=4
xdebug.collect_return=1
xdebug.trace_format=0
xdebug.trace_options=0 ; 0 = append, 1 = create new
xdebug.trace_output_dir=/dev/shm/
xdebug.trace_output_name=trace
zend_extension=$XDEBUG_S0
EOF
php -c $CONF -S 0.0.0.0:8000 -t /dev/shm/

cd /dev/shm/; rm trace*; curl
http://127.0.0.1:8000/php.php ; cat trace*
```


V2 - xDebug is all you need, but SLOW AF

```
0.0032 556240 -> define('_IS_BOT_FRIEND', FALSE) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:240
0.0032 556264 >=> TRUE
0.0032 556264 -> strncmp('_ENV', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0032 556264 >=> -663313
0.0032 556264 -> strncmp('_SERVER', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0033 556264 >=> -659738
0.0033 556264 -> strncmp('GLOBALS', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0033 556264 >=> -2234384
0.0033 556264 -> strncmp('f', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0033 556264 >=> -3
0.0033 556264 -> strncmp('_exceptions', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0033 556264 >=> -655079
0.0033 556264 -> strncmp('val', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0033 556264 >=> 851213
0.0033 556264 -> strncmp('var', 'id_', 3) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:271
0.0034 556264 >=> 851219
0.0034 556264 -> preg_match('^(.*)?spip_acces_doc\\.\\.', '/rubrique/sous-rubrique/article/article2') /opt/audit-framework
0.0034 556264 >=> 0
0.0034 556264 -> function_exists('get_magic_quotes_gpc') /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:405
0.0034 556264 >=> TRUE
0.0034 556264 -> get_magic_quotes_gpc() /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:405
0.0034 556264 >=> FALSE
0.0035 556264 -> serialize(array ()) /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:407
0.0035 556520 >=> 'a:0:{}'
0.0035 556520 -> strpos('a:0:{}', '\\000') /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:408
0.0035 556520 >=> FALSE
0.0035 556264 -> function_exists('tmp_lkojfgx') /opt/audit-frameworks/spip/spip8/config/ecran_securite.php:477
```

V2 – xDebug is all you need, but SLOW AF

- Logs useful stuff
- Nothing to maintain
- Mostly portable
- Fast setup 



- Logs way too much
- Configurable to some extent
- Magento2 >= 20Go for 1 dynamic css file 😱
- Slowwwwwwwwwwwwwwwwwwwww 



V3 - fork php-7, curl, sync

- Where/How to insert ?
 - macro `PHP_FUNCTION`
 - `zend_include_or_eval`

=> both exposing *zval**

- `zval` are ugly, we want JSON (*smol lib plz*)
 - 2 files
 - C90, no dependencies
 - MIT License



- How to log ?
 - POST request
 - Libcurl is our friend

- Grep through the build (*aka the C nightmare*)
 - `./configure => grep the configure.ac`
 - `PHP_ADD_SOURCES()` for new `.c` files
 - `LIBS="$LIBS -lcurl"` for new libs



V3 - fork php-7, curl, sync

```
#define ILOG_FUNCTION(name)
    zval*     argc_ = ZEND_NUM_ARGS();
    int      args_ = safe_emalloc(argc_, sizeof(zval), 0);

    if (zend_get_parameters_array_ex(argc_, args_) == FAILURE) {
        // blah blah we deal with errors
    } else {
        log_zval_parameters(args_, argc_, name);
    }

    efree(args_);
```

← The big macro

← Call the logging function on *zval

```
PHP_FUNCTION(pcntl_exec)
{
    ILOG_FUNCTION("pcntl_exec")
    [...]
}
```

← We simply insert it after PHP_FUNCTION

V3 - fork php-7, curl, sync

```
zend_op_array* zend_include_or_eval(zval *inc_filename, int type)
{
    [...]
    switch (type) {
        case ZEND_INCLUDE_ONCE:
        {
            log_zval_parameters(inc_filename, 1, "include_once");
            [...]
        }
        case ZEND_REQUIRE_ONCE:
        {
            log_zval_parameters(inc_filename, 1, "require_once");
            [...]
        }
        case ZEND_INCLUDE:
        {
            log_zval_parameters(inc_filename, 1, "include");
            [...]
        }
    }
}
```

Deal with language constructions
(like eval, include, require, ...)


Log the parameters with the appropriate
function name

V3 - fork php-7, curl, sync

- Faster than XDebug 🎉
- Standardized communication
- Backend easy to implement
- Still really slow
 - Huge HTTP overhead
 - Slows down the main PHP thread
- Need to recompile to add targets
- We broke some tests 🙄



V4 - fork php-7, async UDP + python backend

- Where/How to insert function log ? ✓
- zval are ugly, we want JSON ✓
- How to thread ? (*small lib plz*)
 - pthread has all we need ❤️
- How to communicate between thread ?
 - Avoiding locking the main thread is nice
 - Lock-free FIFO
 - Lock-free ring buffer
- Need init/join, where to hook ? (*aka debugger time!*)
 - Init:
`zend_signal.c::zend_signal_startup()`
 - Join:
`SAPI.c::sapi_shutdown()`
- How to speed up even more ? UDP = 
 - STD is enough : `<netinet/in.h>`
 - UDP Client is managed by the thread routine
- How to build ?
 - `LIBS="$LIBS -lpthread"`



 [Taymindis / lfqueue](#) will be the way

V4 - fork php-7, async UDP + python backend

```
#include <pthread.h>
#include <stdatomic.h>
#include "lfqueue.h"

static pthread_t THREAD = 0;
static atomic_bool SHOULD_TERMINATE = false;
static lfqueue_t* QUEUE = NULL;

void ilog_thread_init() {
    if (! ilog_is_enabled()) { return; }

    // Init the queue before starting the thread
    QUEUE = malloc(sizeof(lfqueue_t));
    int q_ret = lfqueue_init(QUEUE);

    // Init the thread
    int t_ret = pthread_create(&THREAD, NULL, routine, NULL);

    [...]
}

void ilog_thread_join() {
    if (! ilog_is_enabled()) { return; }

    [...]

    // Tell the ILOG thread it should empty the queue then terminate
    SHOULD_TERMINATE = true;
    pthread_join(THREAD, NULL);
}
```

use atomics when used by several threads

init the queue and the thread

tell the thread it should end

V4 – fork php-7, async UDP + python backend

- FASTEEEEER 🚀🚀
- Doesn't even need libcurl anymore
- Threading makes it more flexible
- Fixed some tests ✅
- Need to be compiled for changes (could be more dynamic)
- Tedious to configure
- Still contains broken tests 🙄
- Still missing some features...



V4 - fork php-7, async UDP + python backend

- <https://github.com/laluka/php7.4.26-internalog>



```
☰ README.md ✎
```

Php-internalog

Hi folks, please note this repo is ~~not maintained anymore~~ as we explained in the [Rump a Rennes](#) (and soon [GreHack](#)) talk what is the future of this project: [Snuffleupagus!](#)

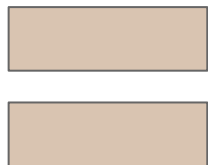
That being said, this project contains multiple approaches related to php introspection for offensive research purpose.

Concept

Long story short, we create a C macro that will help us send the function's name and parameters value in a lock-free FIFO queue that will then send async POST requests to our python backend that can be easily tweaked for more filtering. So you can know what's going on in PHP without suffering from the huge xDebug's overhead! ^.^

HowTo - Internalog

V5 - rump@StHack, long live Snuffleupagus



V5 – rump@StHack, long live Snuffleupagus

Quickstart

```
git clone https://github.com/jvoisin/snuffleupagus
cd snuffleupagus/src
phpize
./configure --enable-snuffleupagus
make
make install
```

This should install the `snuffleupagus.so` file in your extension directory. The final step is adding an extension loading directive, and to specify the location of the [configuration file](#), either in a `conf.d/20-snuffleupagus.ini` file, or directly in your `php.ini` if you prefer:

```
extension=snuffleupagus.so

# This is only an example,
# you can place your rules wherever you want.
sp.configuration_file=/etc/php/conf.d/snuffleupagus.rules
```

V5 - rump@StHack, long live Snuffleupagus

```
# Prevent various `include`-related vulnerabilities
sp.disable_function(function("require_once").param_r(".".*").drop().simulation();
sp.disable_function(function("include_once").param_r(".".*").drop().simulation();
sp.disable_function(function("require").param_r(".".*").drop().simulation();
sp.disable_function(function("include").param_r(".".*").drop().simulation();
sp.disable_function(function("require_once").param_r(".".*").drop().simulation();
sp.disable_function(function("include_once").param_r(".".*").drop().simulation();
```

```
function 'eval', because its argument 'code' content (echo "test \n");
function 'shell_exec', because its argument 'cmd' content (ls) matched
function 'exec', because its argument 'command' content (whoami) match
function 'exec', because its argument 'command' content (whoami) match
function 'system', because its argument 'command' content (whoami) mat
```

If you did not receive a copy of the PHP license, or have any questions about PHP licensing, please contact license@php.net.

```
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'shell_exec', because its argument 'cmd' content (ls) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 13
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'exec', because its argument 'command' content (whoami) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 16
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'exec', because its argument 'command' content (whoami) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 16
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'system', because its argument 'command' content (whoami) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 19
lalu
lalu
The bear black slow jumps over the lazy dog,PHP Warning: copy(not_a_file): failed to open stream: No such file or directory in /opt/php7.4.26-internallog/samples/index.php on line 41
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'proc_open', because its argument 'pipes' content () matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 46
lalu
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'eval', because its argument 'code' content (echo "test \n"); matched a rule in /opt/php7.4.26-internallog/samples/index.php(53) : eval()'d code test
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'include', because its argument 'inclusion path' content (include.php) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 54
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'include', because its argument 'inclusion path' content (include.php) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 54
me-iz-includedPHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'require', because its argument 'inclusion path' content (include.php) matched a rule in /opt/php7.4.26-internallog/samples/index.php
PHP Warning: [snuffleupagus][0.0.0.0][disabled_function][simulation] Aborted execution on call of the function 'require', because its argument 'inclusion path' content (include.php) matched a rule in /opt/php7.4.26-internallog/samples/index.php on line 56
me-iz-included
```

```
-----
/opt/php7.4.26-internallog (master*) » php samples/index.php
```

V5 – rump@StHack, long live Snuffleupagus

- Some rules for you... :)

<https://gist.github.com/laluka/7d9bfb245475d3747af5e5a071f4d167>






snufflepague-audit.rules

```
1 sp.log_media("php");
2
3 sp.ini.key("display_errors");
4 sp.ini.key("default_socket_timeout");
5 sp.ini.key("highlight.comment");
6
7 # Harden the `chmod` function
8 sp.disable_function(function("chmod").param_r(".").drop().simulation());
```

```
10 # Prevent various `include`-related vulnerabilities
11 sp.disable_function(function("include_once").param_r(".").drop().simulation());
12 sp.disable_function(function("include").param_r(".").drop().simulation());
13 sp.disable_function(function("require_once").param_r(".").drop().simulation());
14 sp.disable_function(function("require").param_r(".").drop().simulation());
15
16 # Prevent `system`-related injections
17 sp.disable_function(function("eval").param_r(".").drop().simulation());
18 sp.disable_function(function("system").param_r(".").drop().simulation());
19 sp.disable_function(function("shell_exec").param_r(".").drop().simulation());
20 sp.disable_function(function("exec").param_r(".").drop().simulation());
21 sp.disable_function(function("proc_open").param_r(".").drop().simulation());
22
23
24 # Prevent runtime modification of interesting things
25 sp.disable_function(function("ini_get").param_r(".").drop().simulation());
26 sp.disable_function(function("ini_set").param_r(".").drop().simulation());
27
28 # Need to be allow for example to execute Scheduled tasks
29 sp.disable_function(function("function_exists").param_r(".").drop().simulation());
30 sp.disable_function(function("is_callable").param_r(".").drop().simulation());
31
32 # Ghetto sqli hardening
33 sp.disable_function(function("QueryBuilder::setParameter").param_r(".").drop().simulation());
34
35 # File upload
36 sp.disable_function(function("move_uploaded_file").param_r(".").drop().simulation());
```


V5 – rump@StHack, long live Snuffleupagus

- Fine-grained filters
- Small overhead
- Maintain only your rules
- Send your feature request to @jvoisin
- Niche-Project == Niche-Bugs
- 30 contributors, mostly 2 guys 
- Various log behaviors (for now, needs PR!)
- `~_(\ツ)_/~`



4. Pentest, Méthodology, & Findings

I DON'T KNOW WHO YOU ARE

**BUT I'LL FIND YOU,
CRAWL YOU, FFUF YOU, X8 YOU**

Does it actually work?

Rediscovering n-days for

- Various Wordpress plugins
- Maarch Courier
- Spip ❤️
- Custom tests 🤔

By logging

- Filesystem functions
- SSRF-related functions
- Exec / pass_thru / shell_exec / proc_*
- Include / require
- Custom functions



Does it actually work? Yes: Spip 0-day SSTI

- Why & How

- Find Spip + Credentials
- Set article title to `<?php phpinfo(); ?>`
- Bump article state - draft to evaluation
- Email(**eval**(Article X submitted))
- Enjoy shell

- Sources

- <https://github.com/laluka/php7.4.26-internallog>
- <https://github.com/spip/SPIP>
- https://thinkloveshare.com/hacking/rce_on_spip_and_root_me_v2/



Demo time!

- Without Snuffleupagus
 - www.youtube.com/watch?v=q33U3hfpGRs



- With Snuffleupagus
 - www.youtube.com/watch?v=lBDVIeeXBKQ





Does it actually work? Yes: Spip 0-day SSTI

- Did it work on our beloved hacking platform?

- Yes
- Yes, again?
- Weird but Yes...

```
root@chaos ~  
└─> /opt/lalulife/web/dns-and-http-bins/.py3/bin/python -u /opt/lalulife/web/dns-and-http-  
Request: [173.194.170.14:40206] (udp) / 'rce1.d.thinkloveshare.com.' (A)  
queried_host rce1.d.thinkloveshare.com.  
Reply: [173.194.170.14:40206] (udp) / 'rce1.d.thinkloveshare.com.' (A) / RRs: A,AAAA,MX  
└─>
```

```
Request: [172.217.41.13:39135] (udp) / 'rce1.d.thinkloveshare.com.' (A)  
queried_host rce1.d.thinkloveshare.com.  
Reply: [172.217.41.13:39135] (udp) / 'rce1.d.thinkloveshare.com.' (A) /
```

```
Laluka 04/10/2022  
Heuuu, les potes ? La payload a re trig xD  
└─>  
root@chaos ~  
└─> /opt/lalulife/web/dns-and-http-bins/.py3/bin/python -u /opt/lalulife/web/dns-and-http-  
Request: [109.0.64.15:10699] (udp) / 'rce1.d.thinkloveshare.com.' (A)  
queried_host rce1.d.thinkloveshare.com.  
Reply: [109.0.64.15:10699] (udp) / 'rce1.d.thinkloveshare.com.' (A) / RRs: A,AAAA,MX  
Request: [109.0.64.16:11023] (udp) / 'rce1.d.thinkloveshare.com.' (AAAA)  
queried_host rce1.d.thinkloveshare.com.  
Reply: [109.0.64.16:11023] (udp) / 'rce1.d.thinkloveshare.com.' (AAAA) / RRs: A,AAAA,MX  
└─>  
No clue why, mais ca a re trig x)
```



Do not rush,
The road is long









But at some point,
we'll have to attack
big frameworks! ;)

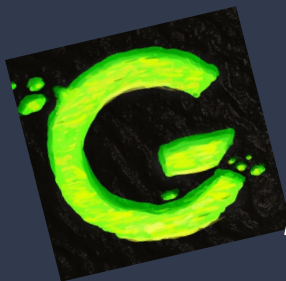


5. Kudos

Kudos

- @elvanderb - StHack 
- @dustriorg - Snuffleupagus 
- @newsoft - Entremetteur 
- Php Forever 
- Staff .RAR & GreHack 
- Community (you) 

Custom php Introspection for 0-Day Research



GREHACK
New is *still not* always better.



@Groupf_ & @TheLaluka
Thinkloveshare.com
soundcloud.com/groupf-prod



Thank you!