

PHP代码的安全审查

汤青松

内容大纲

1 代码审计辅助工具

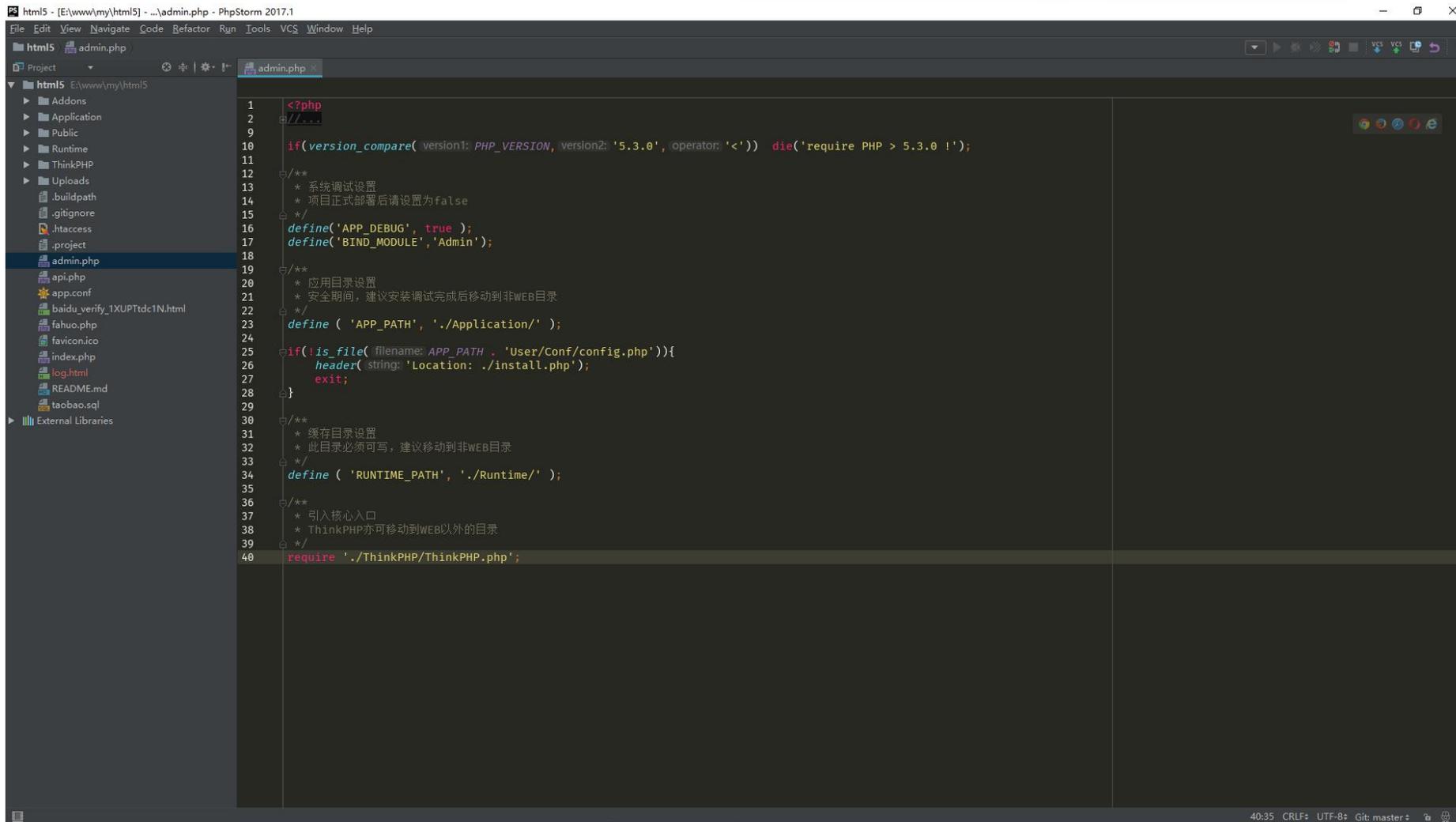
2 常规漏洞挖掘

3 逻辑漏洞挖掘

代码审计辅助工具

- 1. phpstorm 代码编辑器
- 2. PHP代码审计系统—RIPS
- 3. seay 源代码审计系统

phpstorm 代码编辑器



```
1 <?php
2 //...
9
10 if(version_compare('version1: PHP_VERSION', version2: '5.3.0', operator: '<')) die('require PHP > 5.3.0 !');
11
12 /**
13  * 系统调试设置
14  * 项目正式部署后请设置为false
15  */
16 define('APP_DEBUG', true);
17 define('BIND_MODULE', 'Admin');
18
19 /**
20  * 应用目录设置
21  * 安全期间, 建议安装调试完成后移动到非WEB目录
22  */
23 define('APP_PATH', './Application/');
24
25 if(!is_file(filename: APP_PATH . 'User/Conf/config.php')){
26     header(string: 'Location: ./install.php');
27     exit;
28 }
29
30 /**
31  * 缓存目录设置
32  * 此目录必须可写, 建议移动到非WEB目录
33  */
34 define('RUNTIME_PATH', './Runtime/');
35
36 /**
37  * 引入核心入口
38  * ThinkPHP亦可移动到WEB以外的目录
39  */
40 require './ThinkPHP/ThinkPHP.php';
```

PHP代码审计系统—RIPS

path / file: d:\cipher3\timelo

verbosity level: 1. user tainted

code style: ayti

File: D:\cipher3\timelo

Cross-Site Scripting

```
Userinput reaches
```

```
8: print print
```

```
• 2: func
```

```
requires
```

```
7:
```

```
Userinput is passed
```

```
26: get user for
```

```
• 24: $data
```

```
requires:
```

```
8:
```

```
25:
```

hide all

user defined functions and calle

graph list

○ sources
● sensitive sinks
● vulnerability

close_database()

connect_to_database()

login()

get_project_formular()

insert_user_in_databas

show_time_clock()

get_working_state()

system_()

system_()

mysql_real_escape_str

sysexit()

insert_project_into_database()

insert_project_into_da

```
rd]]);
```

```
"Surname" ] != "" ) && isset($_I
```

RIPS 0.40

seay 源代码审计系统



SQL注入漏洞

1. 挖掘经验

- 寻找裸写SQL
- 寻找\$_GET、\$_POST、\$_COOKIE

2. 漏洞防范

- 参数过滤
- 使用PDO



XSS漏洞

1. 挖掘经验

- 寻找搜索、存储后可以展示的位置
- 寻找\$_GET、\$_POST、\$_COOKIE

2. 漏洞防范

- 代码转义



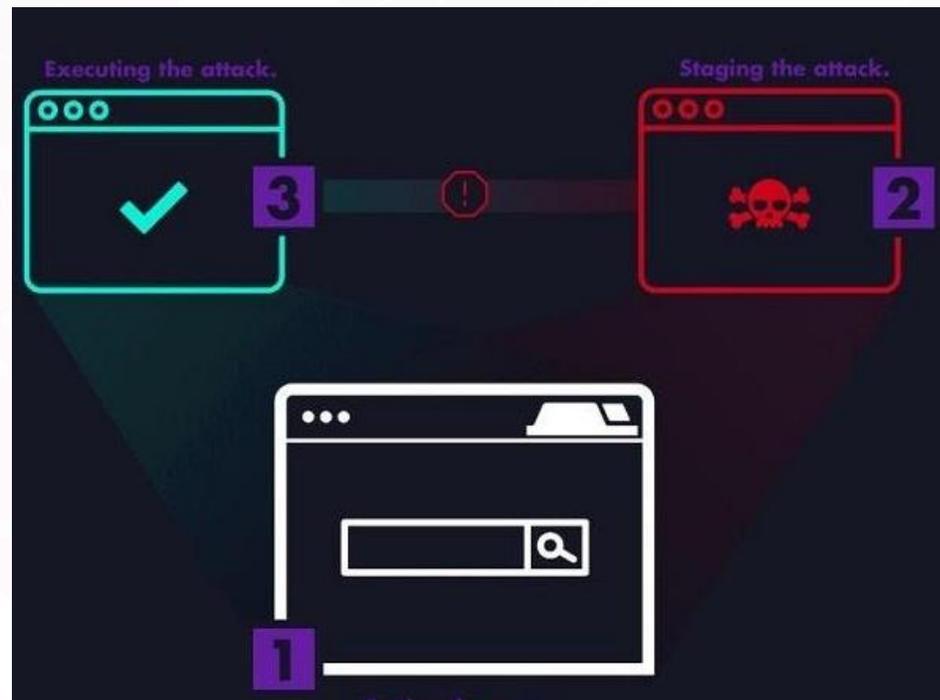
CSRF漏洞审计

1. 挖掘经验

- 打开动态页面，看看有没有token验证
- 不加refererer访问，如果返回数据一致，说明有漏洞

2. 漏洞防范

- 图片验证码验证
- token验证



常规漏洞审计总结

- 1. 常规漏洞特点
- 2. 防范方案

1. 常规漏洞特点

- 1. 参数没有过滤
- 2. 没有对参数验证来源

好疼,我把缝衣针吞下去了!



2.常规漏洞防范方案

- ☞ 1. taint PHP安全扩展
- ☞ 2. ngx_lua_waf nginx安全扩展

taint PHP安全扩展

```
<?php
$a = $_GET['a'];

$file_name = '/tmp' . $a;
$output    = "Welcome, {$a} !!!";
$var       = "output";
$sql       = "Select * from " . $a;
$sql       .= "ooxx";

echo $output;
//Warning: main(): Attempt to echo a string which might be tainted in xxx.php on line x

print $$var;
//Warning: main(): Attempt to print a string which might be tainted in xxx.php on line x

include($file_name);
//Warning: include() [function.include]: File path contains data that might be tainted in xxx.php on x

mysql_query($sql);
//Warning: mysql_query() [function.mysql-query]: First argument contains data that might be tainted in xxx.php on line x
?>
```

在一些关键函数或语句(echo, print, system, exec)

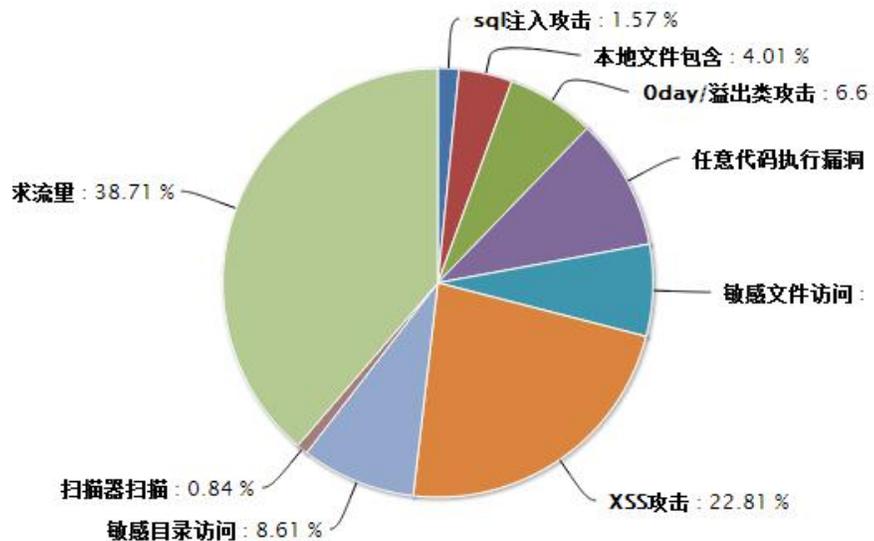
没有经过转义, 安全过滤处理, 就使用来自\$_GET,

\$_POST或者\$_COOKIE的数据,Taint会提示Warning

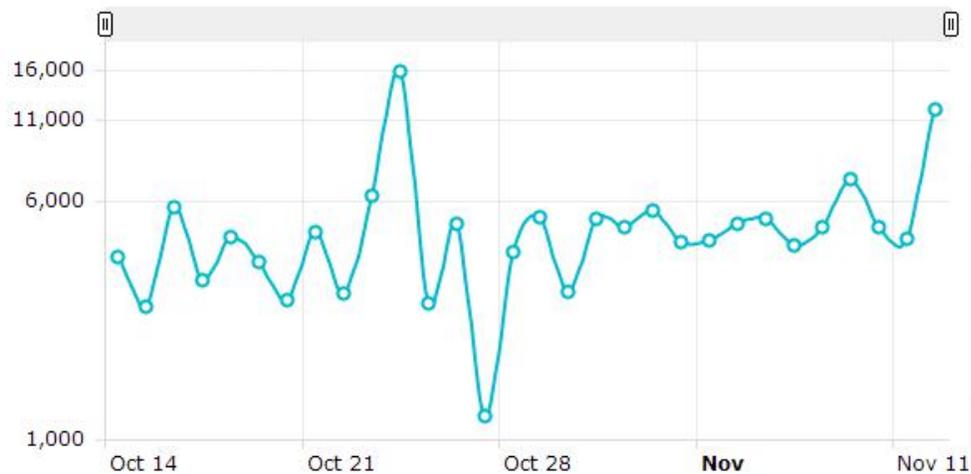


ngx_lua_waf nginx安全扩展

WEB攻击类型统计



WEB攻击情况走势



概况统计

已拦截180672次web攻击

攻击者IP地址个

攻击行为统计

SQL注入2837条

文件包含7238条

任意代码执行17869条

命令执行11988条

XSS攻击41216条

目录探测15554条

敏感文件12509条

扫描器扫描1516条

异常请求69945条



WeChat: [songboy8888](#)

Email: soupqingsong@foxmail.com

