The directory traversal Vulnerability of ftcms

Exploit Title: directory traversal

Date: 2022-04-29

Exploit Author: sunjiaguo

Vendor Homepage: http://www.ftcms.cn/ <http://www.ftcms.cn/>

Software Link: http://www.ftcms.cn/skin/ftcms_v2.1.zip <http://www.ftcms.cn/skin/ftcms_v2.1.zip>

Version: <=v2.1

Tested on: Windows 10

1.Vulnerability analysis

The principle of this code execution vulnerability is caused by using the background template modification function. Next, analyze how it is caused according to the code.

First, locate the template and modify the file code. The file location is admin/controllers/tp.php

Plain Text | Copy

```
对应请求链接为
http://ft.rapoo.top/admin/index.php/tp/file_lists/?style=template&tp=default
```

The corresponding method of database configuration writing is file_edit

admin/controllers/tp.php Plain Text Copy

```
  //文件列表
  public  function file_lists(){

       $this->load->helper('file');//加载文件辅助函数

            $data['style'] = isset($_GET['style']) && trim($_GET['style']) ? tri
m($_GET['style']) : '';
        $data['tp']=$_GET['tp'];
            if($data['style']=='css'){
             $dir="../skin/template/".$data['tp']."/";
           $data['dir']="根目录/skin/template/".$data['tp']."/";
          }elseif($data['style']=='template'){
             $dir="../application/views/".$data['tp']."/";
           $data['dir']="根目录/application/views/".$data['tp']."/";
         }

            $data['list']=get_filenames($dir,ture);//获取所有文件列表

              $this->load->vars('data',$data);

          $this->load->view($this->router->class.'/file_lists');

      }
```

```
$this->load->helper('file');//加载文件辅助函数
```

First, the file helper function in the help class will be used

```
$data['style'] = isset($_GET['style']) && trim($_GET['style']) ? trim($_GET['style']) : '
$data['tp']=$_GET['tp'];
```

First, judge whether the two style parameters are set in the get request. If so, use the trim function to remove spaces, otherwise it is empty. Then directly obtain the TP parameter from the get request. This parameter has not been filtered and processed, which also paves the way for subsequent vulnerability exploitation

```
if($data['style']=='css'){
    $dir="../skin/template/".$data['tp']."/";
    $data['dir']="根目录/skin/template/".$data['tp']."/";
}elseif($data['style']=='template'){
    $dir="../application/views/".$data['tp']."/";
    $data['dir']="根目录/application/views/".$data['tp']."/";
}
```

Check the value of style to determine whether to modify the CSS file or the template file. You can choose at will

```
$data['list']=get_filenames($dir,ture);//获取所有文件列表
```

Then call get_ The filenames function lists the files in the specified directory

get_filenames function is in the help class. Let's go in and analyze it

```php
function get_filenames($source_dir, $include_path = FALSE, $_recursion = FALSE) {

        static $_filedata = array();

        if ($fp = @opendir($source_dir))
        {
            // reset the array and make sure $source_dir has a trailing slash on the initial call
            if ($_recursion === FALSE)
            {
                $_filedata = array();
                $source_dir = rtrim(realpath($source_dir), DIRECTORY_SEPARATOR).DIRECTORY_SEPARATOR;
            }

            while (FALSE !== ($file = readdir($fp)))
            {
                if (@is_dir($source_dir.$file) && strncmp($file, '.', 1) !== 0)
                {
                    get_filenames($source_dir.$file.DIRECTORY_SEPARATOR, $include_path, TRUE);
                }
                elseif (strncmp($file, '.', 1) !== 0)
                {
                    if($include_path == TRUE) {
                        $_filedata[]=array($source_dir.$file,$file);

                    }else{
                      $_filedata[]=$file;
                    }



                }
            }
            return $_filedata;
        }
        else
        {
            return FALSE;
        }
```

First, let's take a look at the parameter of the function, $source_ Dir represents the original path, which can be absolute path or relative path, $include_ The default value of path is false, and the default value is false$_ The default is false

```php
static $_filedata = array();
```

First, define an empty array that should be used to store the obtained file name

```php
if ($fp = @opendir($source_dir))
```

Open the specified directory and read its contents

```
if ($_recursion === FALSE)
{
    $_filedata = array();
    $source_dir = rtrim(realpath($source_dir), charlist: DIRECTORY_SEPARATOR).DIRECTORY_SEPARATOR;
}
```

Because$_ Recursion defaults to false, so the code here will be executed, that is, the path we passed in will be processed to obtain the real absolute path, that is, we can use/ To jump to the directory

```
while (FALSE !== ($file = readdir($fp)))
{
    if (@is_dir( filename: $source_dir.$file) && strncmp($file, str2: '.', len: 1) !== 0)
    {
        get_filenames( source_dir: $source_dir.$file.DIRECTORY_SEPARATOR, $include_path, _recursion: TRUE);
    }
```

Then, if the current path is a folder, recursively read all files in the specified directory

```
elseif (strncmp($file, str2: '.', len: 1) !== 0)
{
    if($include_path == TRUE) {
        $_filedata[]=array($source_dir.$file,$file);

    }else{
      $_filedata[]=$file;
    }


}
```

Then assign the obtained file path to $_filedata

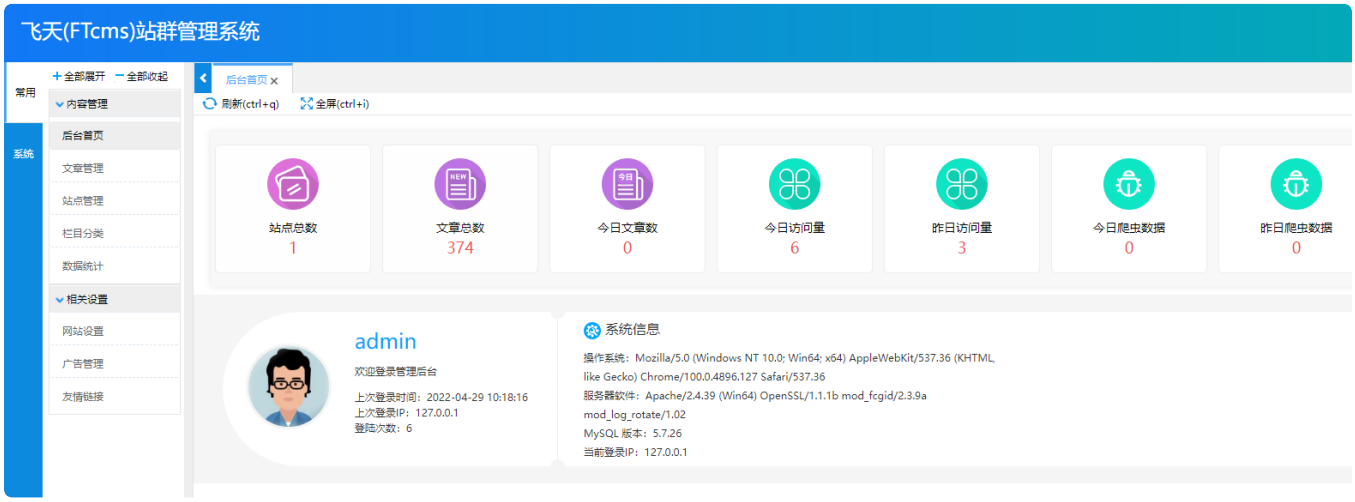
```
    $this->load->vars('data',$data);

    $this->load->view($this->router->class.'/file_lists');
```

Finally, the read file content is displayed in the view

The final POC is as follows

2.Loophole recurrence

2.1 login



2.2 request the poc

the poc is: