2_bitcoin_2 writes up

by Hence Zhang@Lancet

default filter 后门 Assert

介绍：

在Application/Common/Conf/config.php处，我们定义了MY_DEFAULT_FILTER这个变量，而这个变量在基于thinkphp的框架中，会被MY_I()函数使用，并使用类似array_map的函数进行处理，使MY_DEFAULT_FILTER中的字符串变成函数指针，从而形成可能的后门。

难度：

高（需要了解thinkphp特性，很难定位到后门代码）

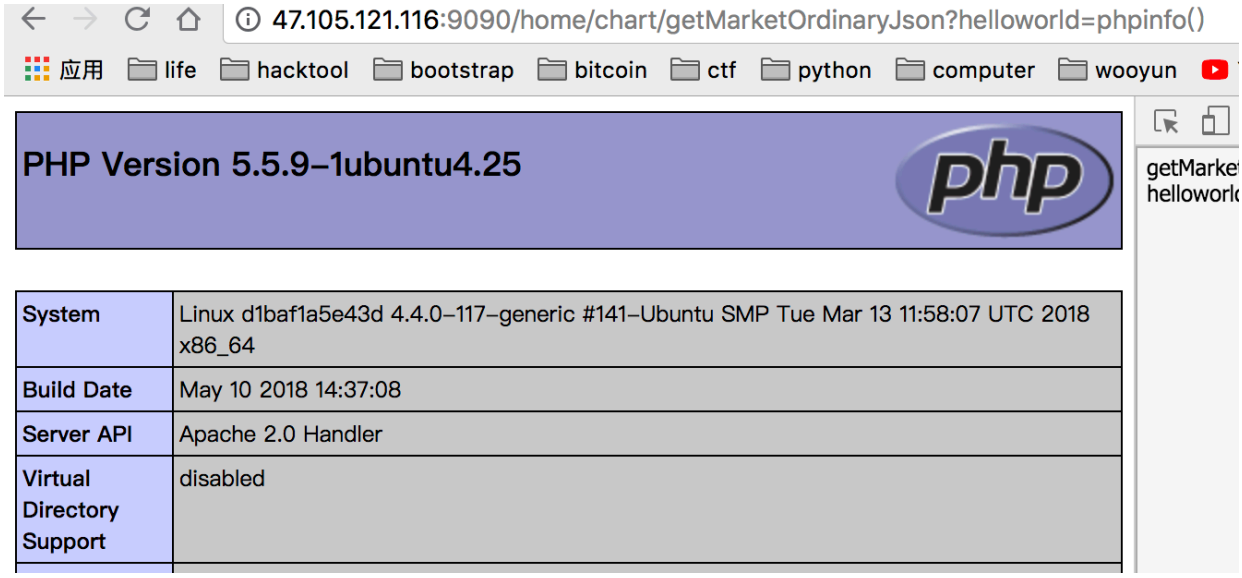
利用：

随意找一处使用MY_I()函数的地方，可以在其请求的参数中任意地注入代码，如：

Application/Home/Controller/ChartController.class.php

访问如下url:

/home/chart/getMarketOrdinaryJson?helloworld=phpinfo()



模板文件名可控，本地文件包含

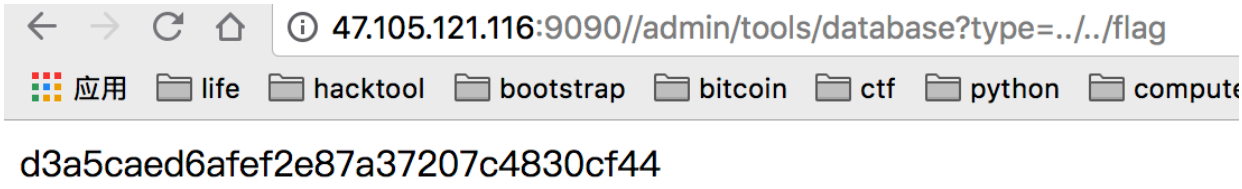
介绍：

display函数包含模版文件，如果display的变量可控，则可以产生文件包含漏洞

难度：高（需要了解thinkphp特性）

利用：在Application/Admin/Controller/ToolsController.class.php中的database函数中，this->display函数中的参数为用户可控。

/admin/tools/database?type=../../flag



配合图片上传可以实现任意代码执行

temp文件后缀php 通过控制缓存可以造成任意代码执行

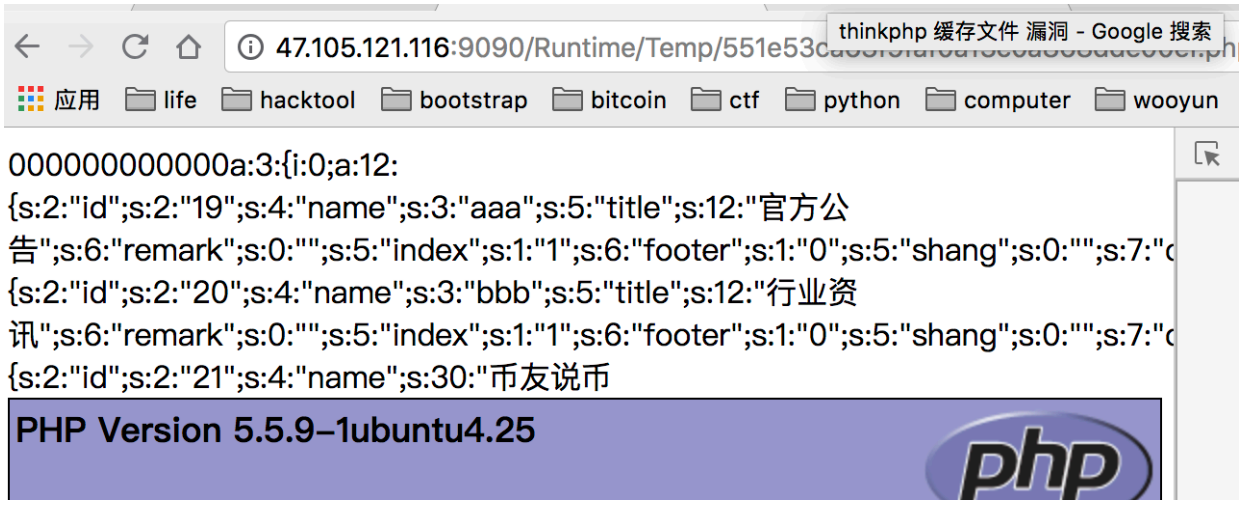
介绍：

在程序中大量使用了thinkphp中的S函数，而S函数将用户提供的变量输出到本地的php文件中，造成任意文件内容写入，从而导致任意代码执行

难度：中（thinkphp的核心代码经过了修改，修补漏洞时要找到修改的关键位置）

利用：存在问题的代码在：ThinkPHP/Library/Think/Cache/Driver/File.class.php

只要找到一处能够控制缓存内容的地方，即可实现任意代码执行。如在后台/admin/article/typeedit/id/21.html修改类别标识为PHP代码，然后重新访问index.php生成缓存，此时可以看到Runtime/Temp/551e53ca63f9faf0a13c0a868dde00ef.php中已经出现了php代码，直接访问即可执行php代码



admin的相关api接口ssrf 可以读取flag

介绍：admin 控制器中的callOnce函数可以构造任意的本地请求和远程请求，从而导致ssrf

难度：低

利用：攻击payload：/admin/admin/callOnce?url=file:///flag

firephp 读取flag

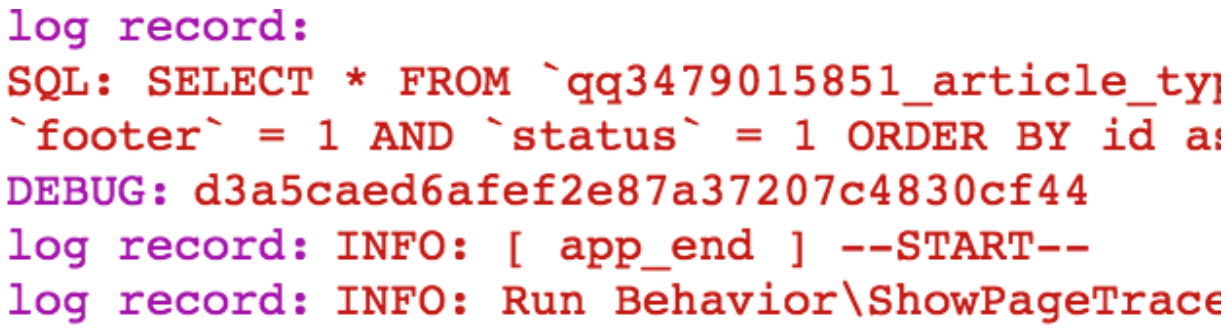
介绍：该项目全局引入了firephp中的debug函数，一旦接收到firephp的相关头部信息，会在返回头中带上debug函数中指定的字符串，从而造成信息泄露。

难度：高

利用：攻击payload：/home/index/flag?len=35 根据Application/Home/Controller/IndexController.class.php中的代码逻辑，如果flag的文件内容长度和猜测的len的数值不一样，就会返回error，并把flag交给debug函数。

我们在user-agent中写入： FirePHP/0.7.4

即可在返回头部中获得flag



PS：因为题目来源于对某开源项目，漏洞来源于对该项目未公开的漏洞，除以上列出的漏洞，该项目很可能还会存在其他的漏洞。