# File Inclusion

## 文件包含:

文件包含这个漏洞是因为服务器执行PHP文件时，可以通过文件包含函数加载另一个文件中的PHP代码，并且当PHP来执行，这会为开发者节省大量的时间。这意味着您可以创建供所有网页引用的标准页眉或菜单文件。当页眉需要更新时，您只更新一个包含文件就可以了，或者当您向网站添加一张新页面时，仅仅需要修改一下菜单文件，但是因为函数加载的参数没有经过过滤或者严格的定义，可以被用户控制，包含其他恶意文件，导致了执行了非预期的代码。文件包含还分为两种

LFI/RFL这两种：

LEI是本地文件包含漏洞，顾名思义，指的是能打开并包含本地的漏洞。大部分情况下遇到的文件包含漏洞都是LEI。

RFL是远程文件包含漏洞。是只能够包含远程服务器上的文件并执行。由于远程服务器的文件是我们可控的，所以漏洞一旦存在危害极大。

文件包含函数：

PHP常见的包含函数

require()

require\_once()

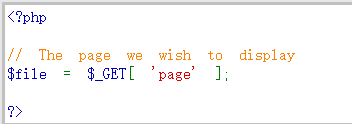
include()

include\_once()

include 和require的区别主要是，include在包含过程中如果出现错误，会抛出一个警告。程序继续正常运行；而require出现错误的时候，会直接报错并推出程序的执行。

而include\_once和require\_once这两个函数，与之前的不同在于这两个函数只包含一次，实用于在脚本执行期间同一个文件有可能被包括超过一次的情况下，你想确保它只被包含一次避免函数重定义，变量重新赋值等问题。

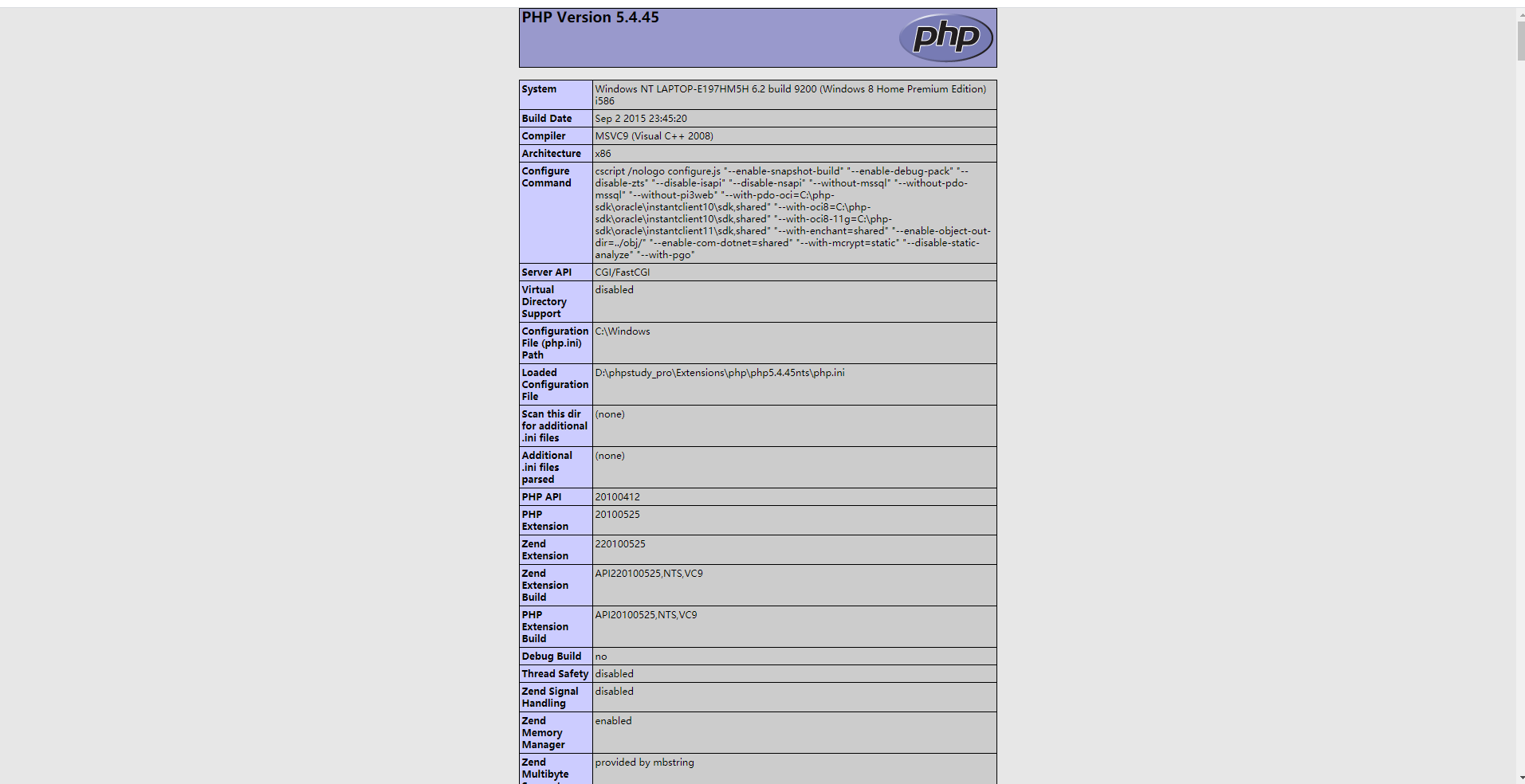
# Level:Low



代码的意思就是就直接接收一个‘page’的GET请求

通过File Upload上传的一个webshell文件再通过给的路径来请求它



弹回

Phpinfo的界面，是因为上传的文件中有一句话木马<?php phpinfo()?>服务器把它当PHP文件来执行了

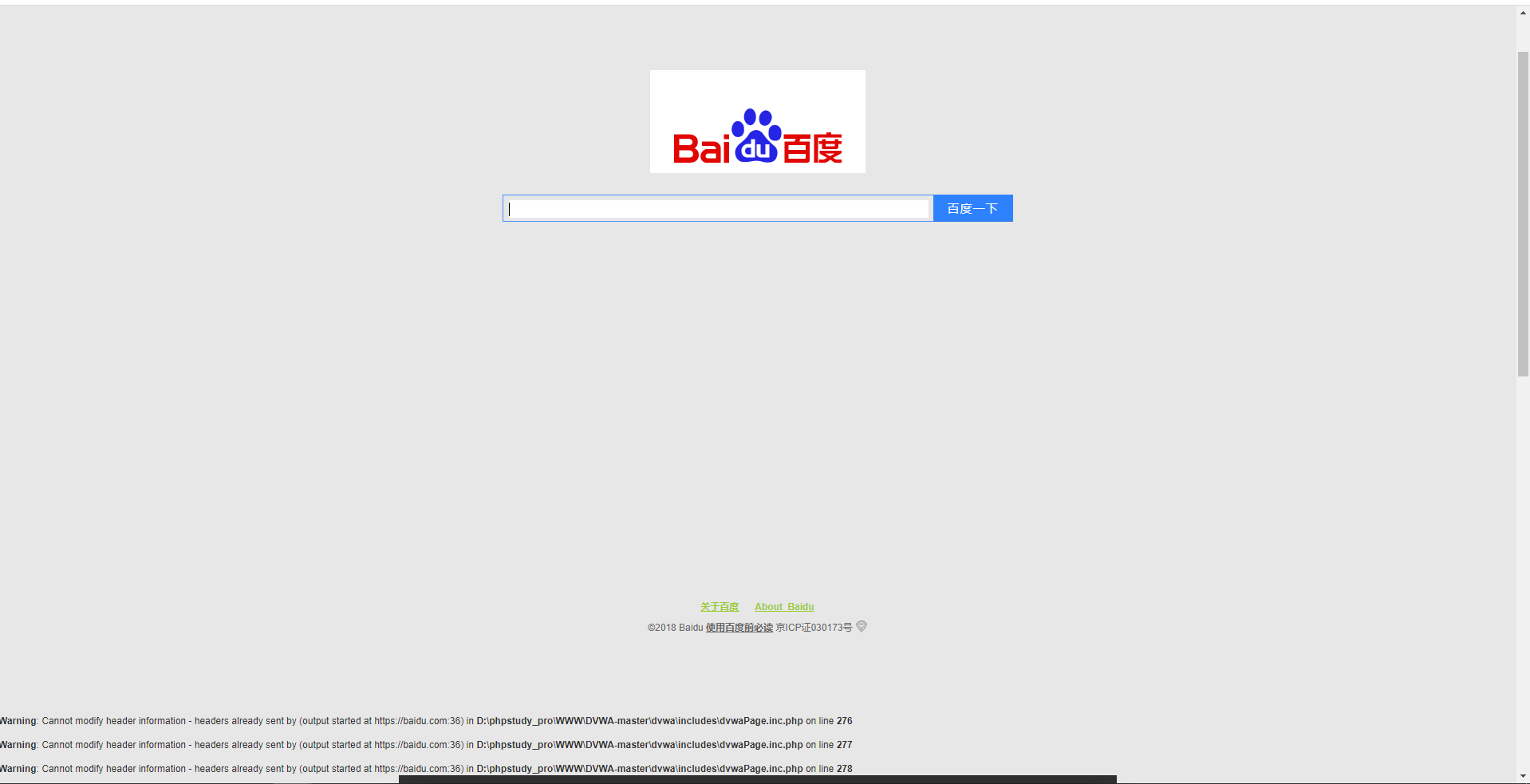
# Level:Medium



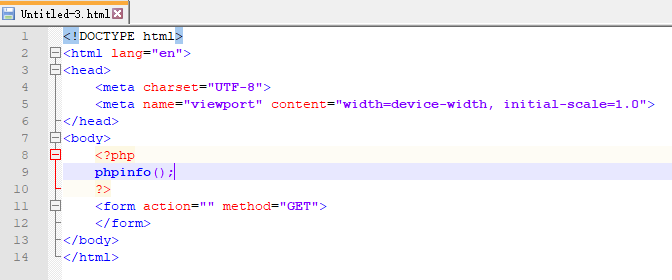
通过这段代码得知这一次把“[http://](NULL)“，“[https://](NULL)”，“../”，“..\”给过滤成了””空置

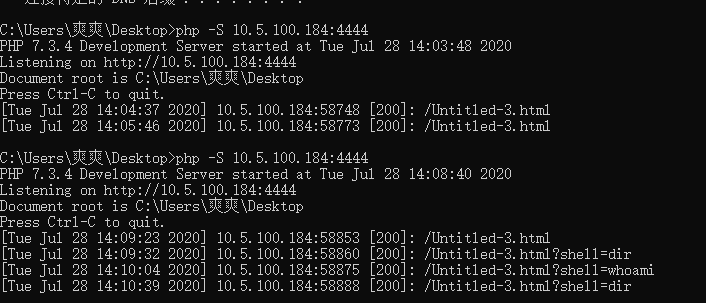
由于这个过滤只过滤一次，就可以使用叠加的方式请求”hthttps://tps://www.baidu.com”

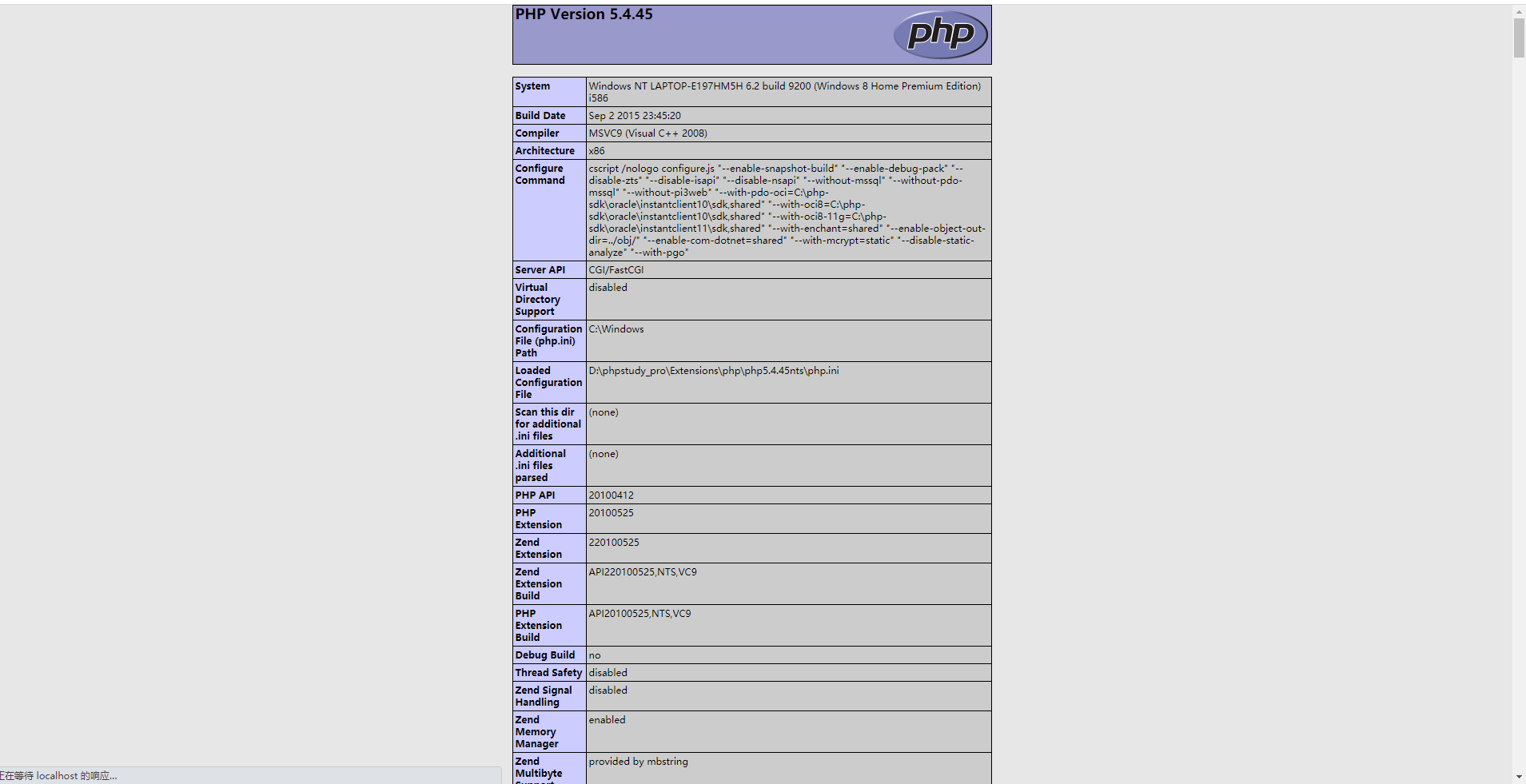




这样可以自己构造一个网站然后通过这个网站来执行一句话木马



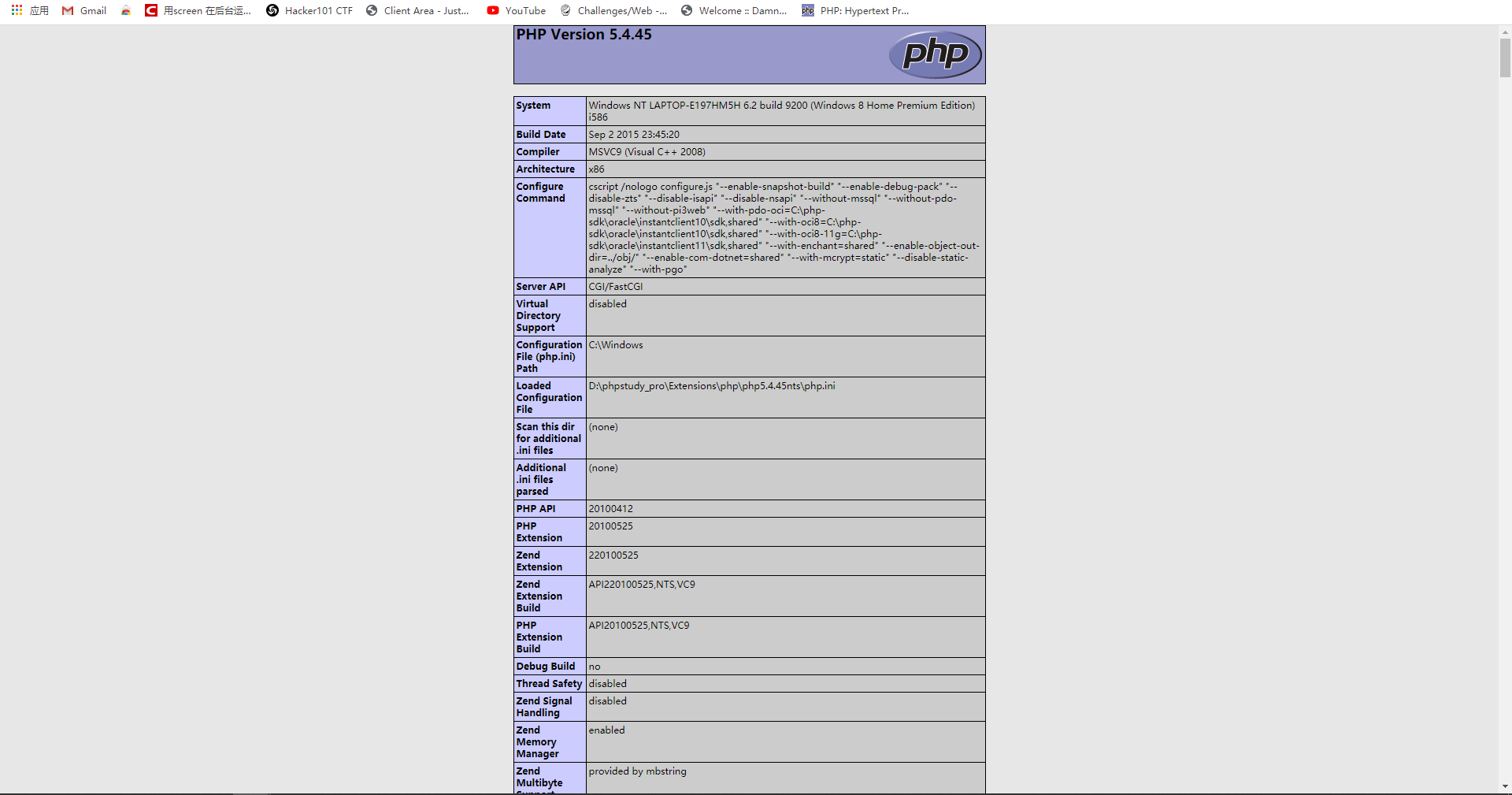
再打开端口，



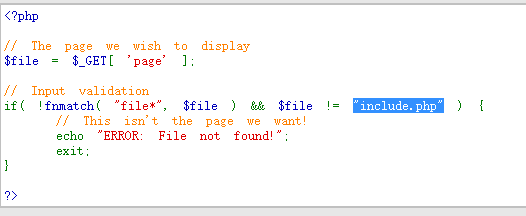
就进入到了phpinfo了

也可以用绝对路径来读取利用File Upload上传的的文件





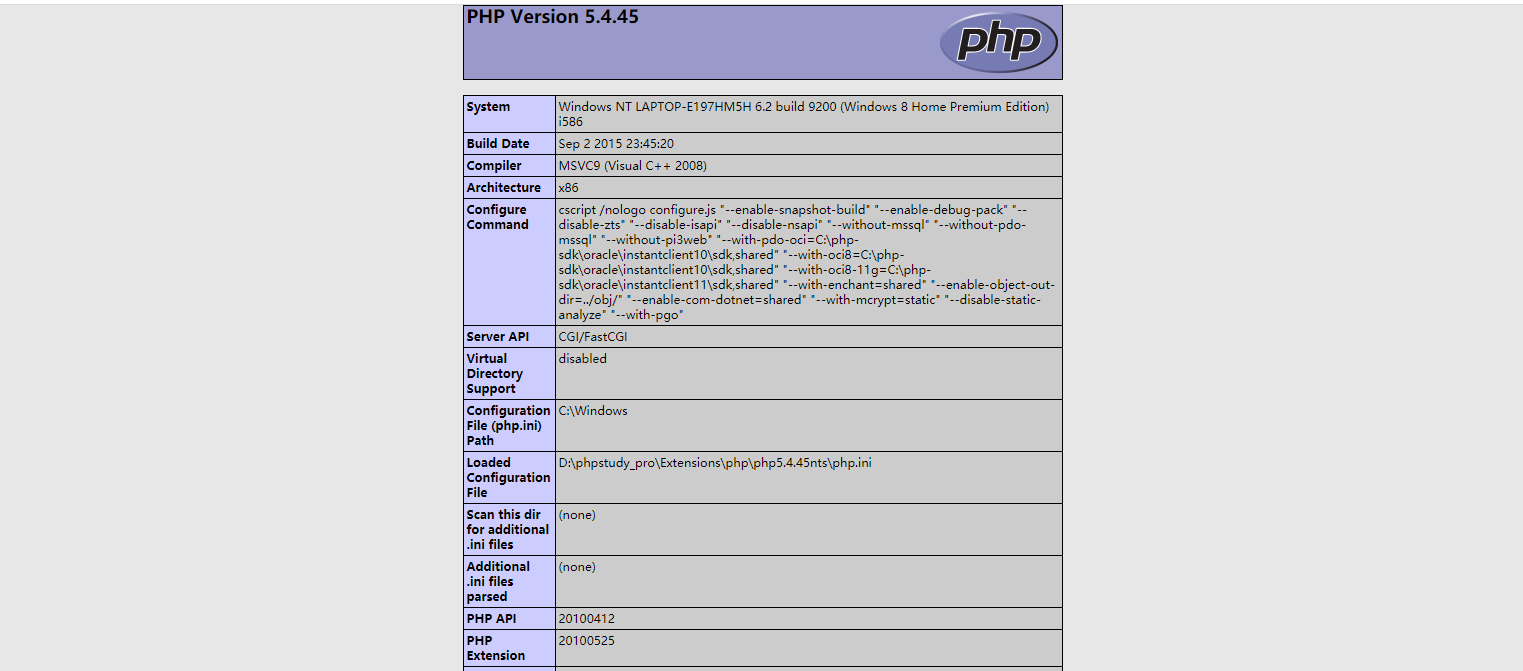
# Level:Higth



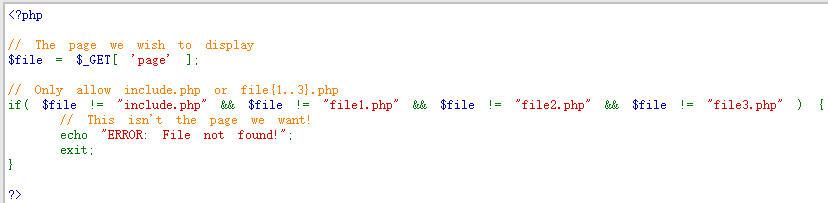
通过这些代码知道了page请求的参数必须是file开头的才会执行

利用File Upload漏洞给的路径再结合报错给的绝对路径构造一个file协议来执行

得出



# Level:impossible



Impossible等级的源码设置了page参数只能是“include.php”、“file1.php”、“file2.php”、“file3.php”