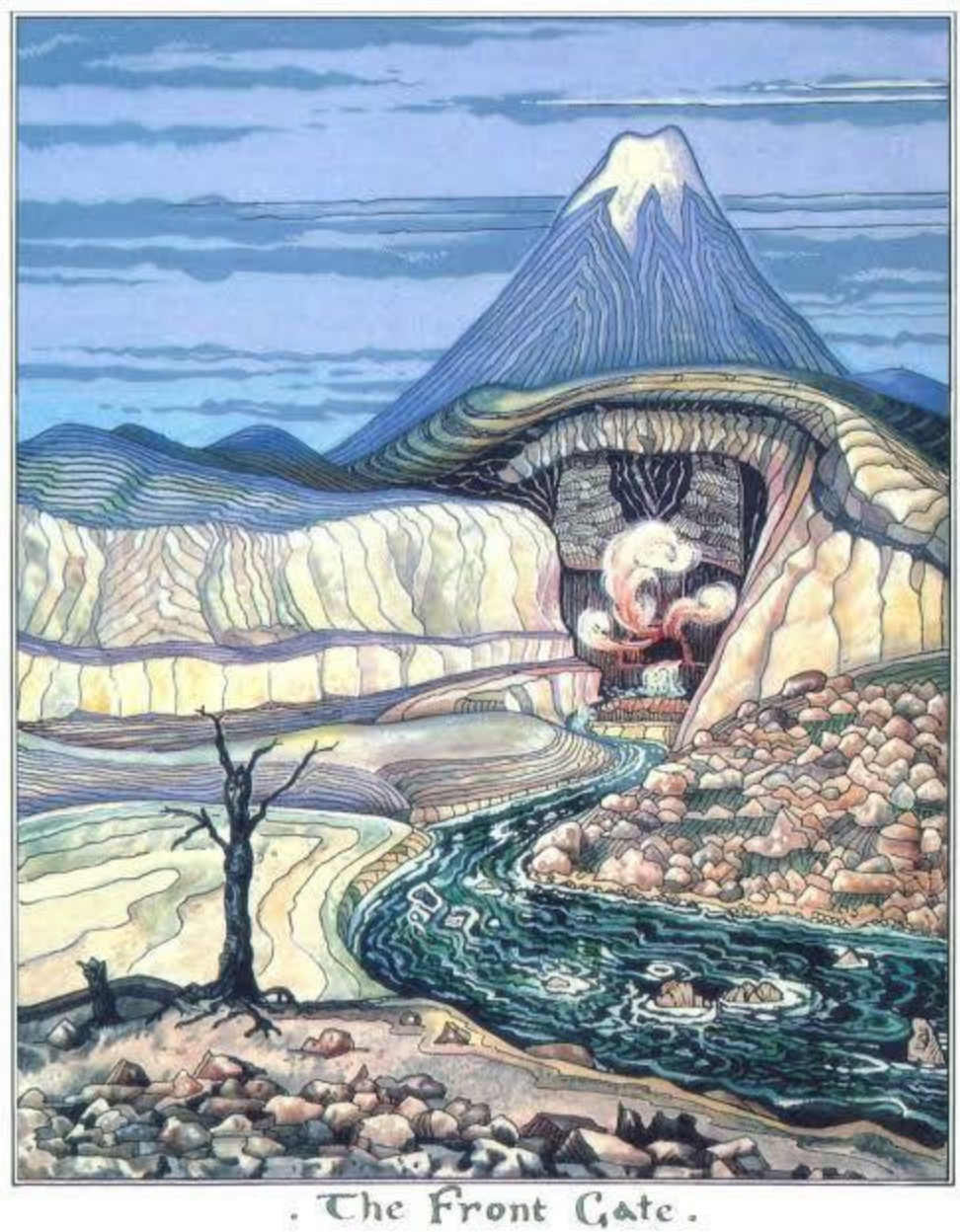
OpBlueRaven: Unveiling Fin7/Carbanak - Part I : Tirion

threatintel.blog/OPBlueRaven-Part1/

PTI Team

July 31, 2020



This article aims to provide its readers with the details about PRODAFT & INVICTUS Threat Intelligence (PTI) team's latest operation on different threat actors; who have been detected to be working in cooperation with the notorious Fin7 APT group.

Throughout this article, which is planned to be released in 6/7 successive parts (similar to other articles on our pentest blog (pentest.blog); we will approach different aspects of our operation, which had been continued for the last 3 months until the end of July.

Throughout these articles; all of which originates from a single OPSEC failure on the threat actor's side, we will try to expand the topic on a step-by-step basis, similar to how we expanded our scope as we've continued to discover further.

Neverbeforeseen Facts about Fin7 and Carbanak: Part 1

Between the months of May and July 2020; four members of PRODAFT Threat Intelligence team have conducted operation BlueRaven. A case study which originated from discovering a minor OpSec failure of a seemingly unimportant group of threat actors. Of course these threat actors have later been found to have ties with the notorious Fin7 / Carbanak threat actors.

PTI's OP has originated from an OPSEC failure on the attacker's side. Unlike previously discovered and published data what makes this OP special is we have managed to discover an important deal of unpublished information about attackers' toolset which reveals the TTP of attackers.

Carbanak Group / Fin7, which was first detected in 2014, is one of the most effective APT groups in the world, and is among the first known APT groups. The group is thought to cause damage over 900 million dollars worldwide. Our OP has resulted to discover following critical findings about these threat actors:

- The real identity of some of the attackers in Fin7 has been obtained.
- Detailed evidence has emerged about Fin7's tools and attack methods.
- The relationship between Fin7 and REvil ransomware group (which will be provided in detail in latter stages) has emerged.

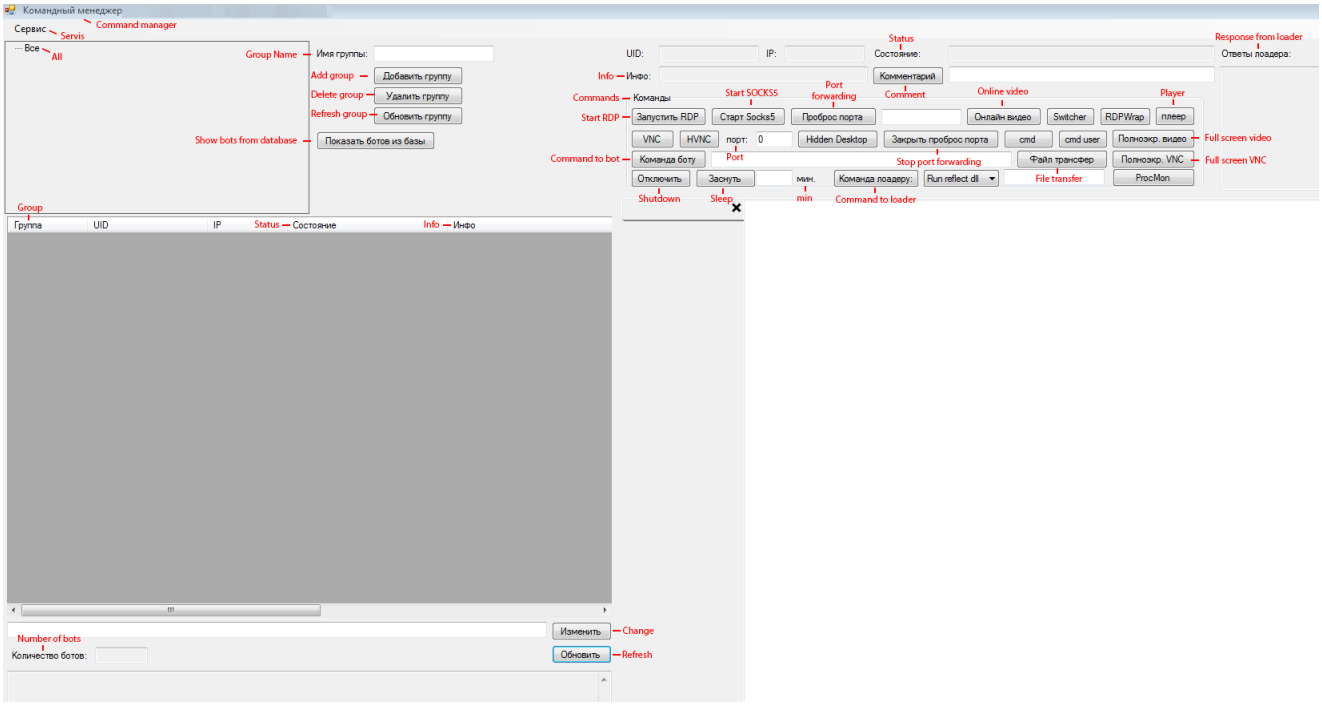
This report was written to raise awareness and assist cyber security experts' analysis. Of course, PRODAFT some of our findings have been redacted. So, authorized bodies may get in touch with PRODAFT or INVICTUS for further disclosure.

Each article will deal with a specific aspect of the operation; including but not limited to attack methods, organizations, and identities of the attackers. Also; our team has managed to eavesdrop different jabber conversations between attackers. Most of these conversations will also be published throughout these series.

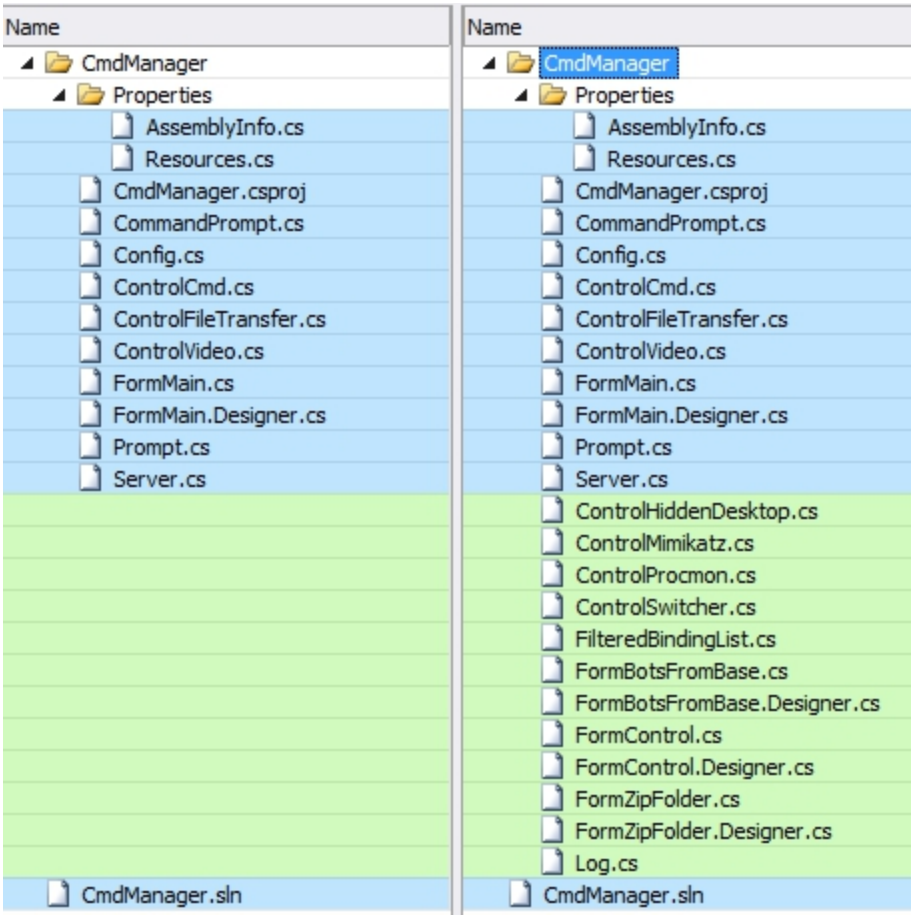
Carbanak Backdoor

Carbanak Backdoor was among the first findings that our group has acquired.
The current version of CARBANAK backdoor (the most known tool of the team that gave the name to the Carbanak group) is the first tool that caught the attention of our team. The

“3.7.5” version, which was compiled in November 2019 according to PE file header, is the last detected version of the backdoor command and control server. The screenshot of the management panel of “3.7.5” version of Carbanak backdoor is given in the screenshot below.



We compared the last version we obtained with the versions of the “Command Manager” in Virustotal in 2017 and conducted a review on this tool. In the image below, the differences between the source codes obtained as a result of the decompilation of the two versions mentioned can be seen. In the following image, which lists only the source codes that differ between the two versions, the left column belongs to the file uploaded to Virustotal in 2017, and the right column belongs to the “3.7.5” version that our team obtained. The blue lines refer to files that differ, while the green lines represent new files. As a result of the examination made on both command and control server software, it was seen that the basic changes were made to manage plugins through the GUI interface, to create a more detailed error log, and to add new language encodings.



6 versions of the malware “Command Manager” tool compiled in 2019 have been identified. The timestamps of the detected versions are given in the image below.

Version	Compile Time
3.7.5	Thu Nov 7 16:50:51 2019
3.7.3	Mon Sep 16 18:06:32 2019
3.7.2	Wed Jul 24 20:52:26 2019
3.7.1	Fri Jul 5 21:16:24 2019
3.6.3	Thu May 16 11:13:05 2019
3.6	Fri Apr 19 10:17:22 2019

In the old version of Bot.dll, which is the component of the malware working on victim devices, 981 functions were detected with disassembly while 706 functions were detected in the new version of the same software. With Diaphora binary diffing tool, 607 functions get the best match score, while 43 functions get partial match. Also, the new bot file has less than 50kb of file size compared to the old version in Virustotal. When the new bot file is examined, it is seen that the functions other than the basic functions in the old version are implemented

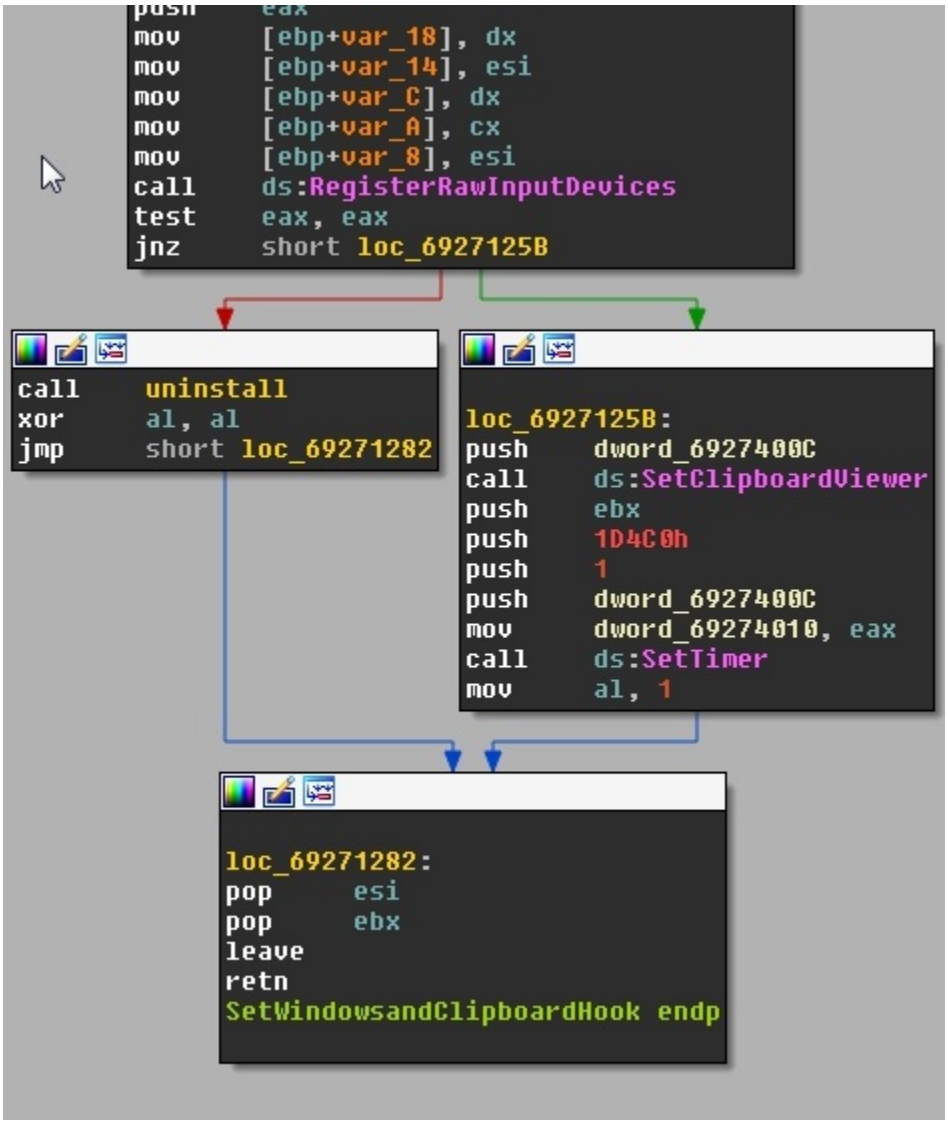
as plugins. These new plugins, which perform operations such as keylogging, process monitoring, are executed fileless with the reflective loading method. As a result, the file size of the malware shrinks, leaving less trace for forensic and signature-based security software solutions. Plugins in the last data obtained in the list below are listed.

- hd.plug
- hd64.plug
- hvnc.plug
- hvnc64.plug
- keylog.dll
- keylog64.dll
- procmon.dll
- procmon64.dll
- rdpwrap.dll
- switcher.dll
- switcher64.dll
- vnc.plug
- vnc64.plug

In this section, some of the plugins that are “not” among the previously discovered files will be examined. As these are among the neverbeforeseen features of the notorious toolkit, we believe following sections to be very important in terms of further analyzing the group’s TTP.

Keylogging Plugin

The “keylog.dll” plugin capturing user keystrokes using the RegisterRawInputDevices API. To determine in which context the keystrokes are used, “Executable File Path”, “Windows Text” and Timestamp information of the foreground process is logged together with the keystrokes.



The keylogging plug-in converts the collected data to Bitmap using Windows GDI + APIs and writes it to the folder named “SA45E91.tmp” in the user’s %TEMP% directory. The image below shows the function that malicious software uses to store data.

```
int CreateLogFile()
{
  int result; // eax@1
  int v1; // edi@1
  int v2; // eax@2
  char v3; // [sp+4h] [bp-148h]@2
  char v4; // [sp+108h] [bp-44h]@2
  unsigned __int16 v5; // [sp+138h] [bp-14h]@2
  unsigned __int16 v6; // [sp+13Ah] [bp-12h]@2
  unsigned __int16 v7; // [sp+13Eh] [bp-Eh]@2
  unsigned __int16 v8; // [sp+140h] [bp-Ch]@2
  unsigned __int16 v9; // [sp+142h] [bp-Ah]@2
  int v10; // [sp+148h] [bp-4h]@1

  result = GenerateBitmap((int)&v10);
  v1 = result;
  if ( result )
  {
    GetTempPathA(0x104, &v3);
    PathAppendA(&v3, off_69274004);             // SA45E91.tmp
    CreateDirectoryA(&v3, 0);
    v2 = PathFindFileNameA(&byte_69274198);     // Window Text
    PathAppendA(&v3, v2);
    GetLocalTime(&v5);
    wsprintfA(&v4, ".%04d%02d%02d%02d%02d.jpg", v5, v6, v7, v8, v9);
    lstrcatA(&v3, &v4);
    sub_69271AE4((int)&v3, v1, v10);
    result = free(v1);
  }
  return result;
}
```

The screenshot of the sample of the logs obtained is given in the image below.

```
----------------------------------------
Process: C:\Program Files (x86)\Google\Chrome\Application\chrome.exe
Window:  - Google Chrome
Time: 30.12.2019 10:52
[L][L][L][L][L][L][L]<Backspace>4pm[L][L][L][L]1130am[L][L][L][L][L][L][L]4pm[L]
[L][L][L][L][L]<Backspace>930am[L][L]<Backspace>330pm[L][L][L][L][L]1130am[L][L]
[L][L][L]<Backspace><Backspace><Backspace><Backspace><Backspace><Backspace><Back
space><Backspace><Backspace><Backspace>939<Backspace>0am[L][L]<Backspace><Backsp
```

Process Monitor Plugin

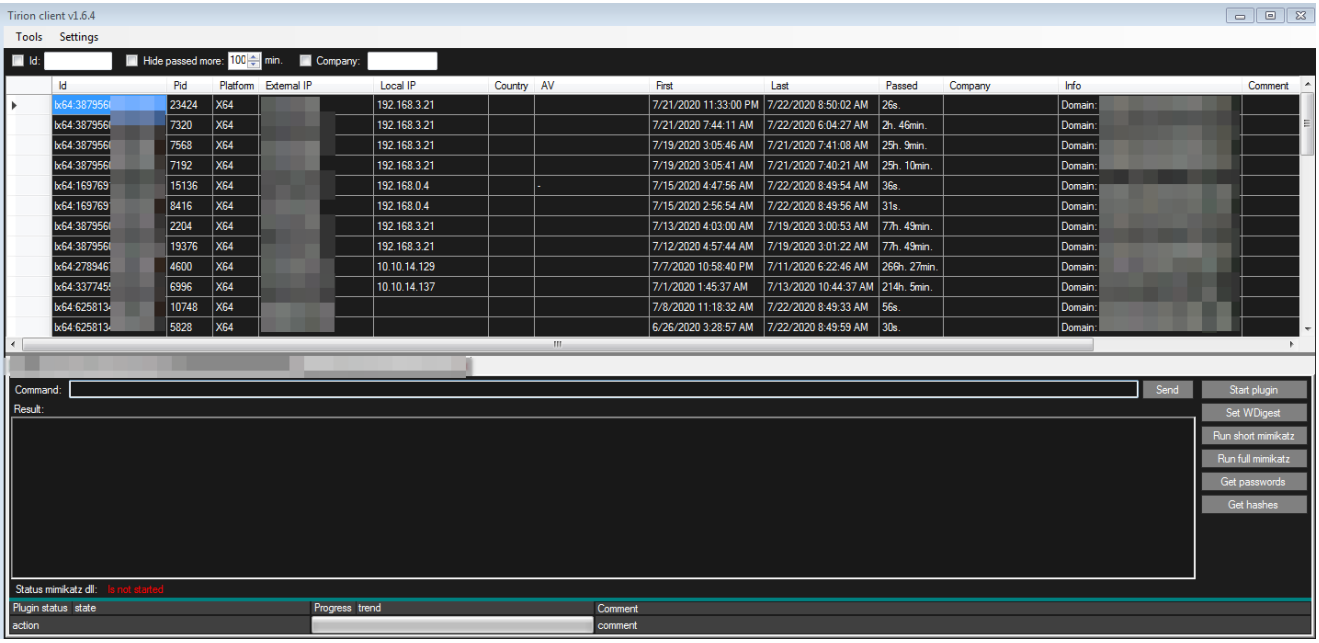
This plugin can track processes running in the target system and is used to obtain information about the start and termination times of the desired process. In the image below, a screenshot of the function that collects information about the running processes is given.

```
int __cdecl sub_10001691(int a1, int a2, DWORD dwFlags)
{
  int v3; // edi@1
  HANDLE v4; // ebx@1
  int v5; // eax@1
  _DWORD *v6; // esi@2
  PROCESSENTRY32 pe; // [sp+Ch] [bp-128h]@1

  v3 = 0;
  v4 = CreateToolhelp32Snapshot(dwFlags, 0);
  pe.dwSize = 296;
  Process32First(v4, &pe);
  v5 = sub_100012B0(0x170u);
  if ( v5 )
    v6 = (_DWORD *)sub_1000164D(v5);
  else
    v6 = 0;
  do
  {
    sub_100015B7(pe.szExeFile);
    sub_100014E4(pe.szExeFile, -1);
    sub_10001781(pe.th32ProcessID, (int)v6, 1);
    sub_100015B7(*v6);
    v6[91] = sub_10001552(v6[69], v6[70]);
    v6[89] = pe.th32ProcessID;
    v6[90] = pe.th32ParentProcessID;
    ++v3;
  }
  while ( !(unsigned __int8)((int (__cdecl *)(_DWORD *, int))a1)(v6, a2) && Process32Next(v4, &pe) );
  CloseHandle(v4);
  sub_1000167B(v6);
  sub_100012F1(v6);
  return v3;
}
```

Tirion Loader a.k.a Future of Carbanak Backdoor

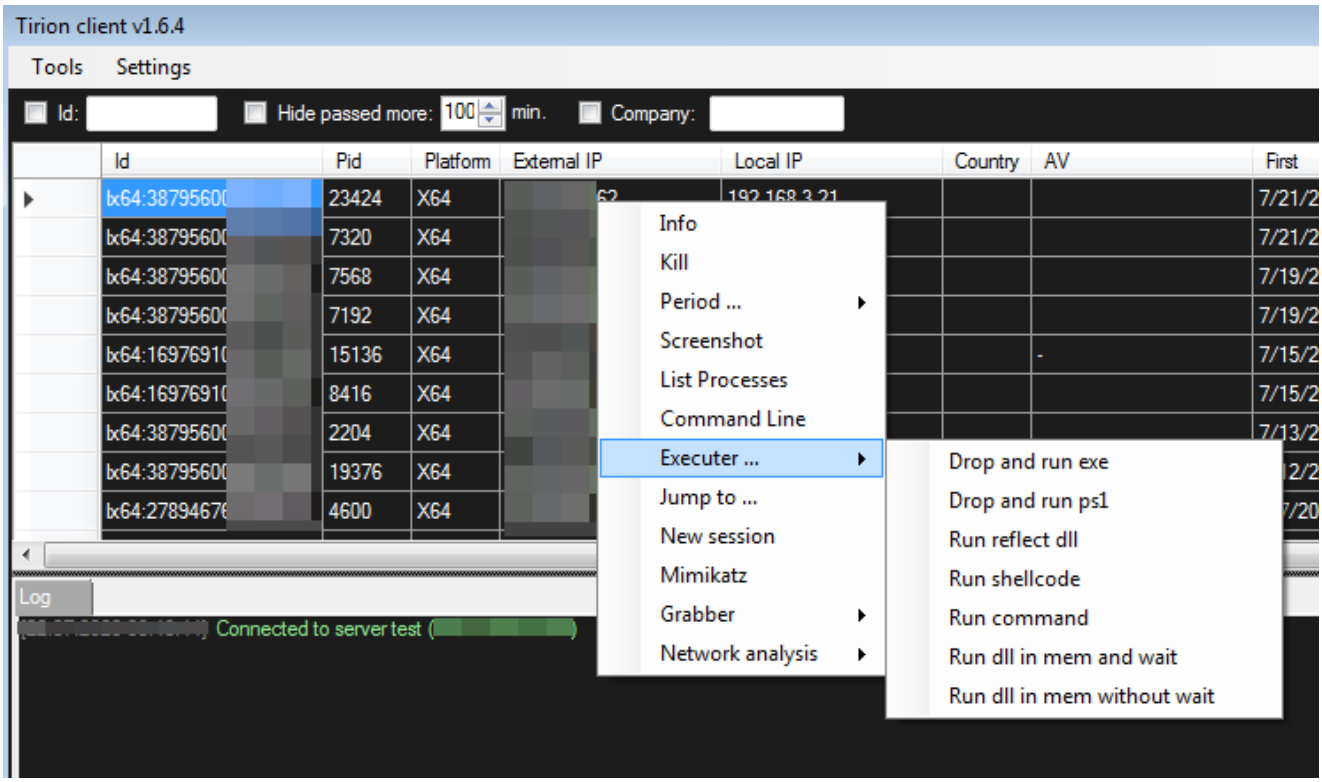
The malware named Tirion, which is thought to be developed to replace the Carbanak backdoor is the new loader tool of the Fin7 group. It contains many functions for information gathering, code execution, recon, and lateral movement purposes. As in the latest version of the Carbanak backdoor, which was examined in the previous section, many functions performed by the malware have been developed as separate plugins. They are loaded and executed fileless in the target system with the reflective loading method. Exposed data shows that the development of the Carbanak backdoor is currently stopped and development and tests are being performed on the Tirion Loader by the same team. Communication logs between attackers show that this new tool is intended to replace the Carbanak backdoor. Our team has detected 8 different Tirion Loader command and control servers currently used.



The functions of Tirion malware are as follows:

- Information Gathering
- Taking Screenshot
- List Running Processes
- Command / Code execution
- Process Migration
- Mimikatz Execution
- Password Grabbing
- Active Directory and Network Recon

The latest detected version of Tirion Loader belongs to the version “1.6.4” compiled on “Sun Jun 28 23:24:03 2020”. The image below shows the actions an attacker can take on a bot device. The “1.0” version, which is the oldest version detected and thought to be the first version used, was compiled on “Thu Mar 05 20:29:53 2020”.



The following text from the “readme.txt” file written by the attackers clearly states the basic components of the malware.

> Описание системы удаленного доступа Tirion
> Система состоит из 3-х компонентов:
>
> 1. Сервер
> 2. Клиент
> 3. Лоадер
>
> Эти компоненты связаны следующим образом:
> Лоадер переодически коннектится к серверу, клиент подключается к серверу с постоянным коннектом. Лоадер выполняет команды от сервера и передает ему ответы. Через клиента пользователь отдает команды лоадеру через сервер. Полученны ответы от лоадера, сервер передает клиенту.

English translation of the related text is given below.

> The system consists of 3 components:
>
> 1. Server
> 2. Client
> 3. Loader
>
> These components are related as follows:
> The loader periodically connects to the server, the client connects to the server with a permanent connection. The loader executes commands from server and sends it responses. Through the client, the user issues commands to the loader through the server. Received responses from the loader,the server transmits to the client.

The file organization of the malware is as follows:

File Structure

```
|-- client
|   |-- client.exe
|   |-- client.ini.xml
|   |-- jumper
|   |   |-- 32
|   |   `-- 64
|   |       `-- l64_r11.ps1
|   |-- keys
|   |   `-- client.key
|   |-- libwebp_x64.dll
|   |-- libwebp_x86.dll
|   `-- plugins
|       `-- extra
|           |-- ADRecon.ps1
|           |-- GetHash32.dll
|           |-- GetHash64.dll
|           |-- GetPass32.dll
|           |-- GetPass64.dll
|           |--
PswInfoGrabber32.dll
|           |--
PswInfoGrabber64.dll
|           |-- PswRdInfo64.dll
|           |--
powerkatz_full32.dll
|           |--
powerkatz_full64.dll
|           |--
powerkatz_short32.dll
|           `--
powerkatz_short64.dll
|-- loader
|   |-- builder.exe
|   |-- loader32.dll
|   |-- loader32.exe
```

```
|   |-- loader32_NotReflect.dll
|   |-- loader64.dll
|   |-- loader64.exe
|   `-- loader64_NotReflect.dll
|-- readme.txt
`-- server
    |-- AV.lst
    |-- System.Data.SQLite.dll
    |-- ThirdScripts
    |-- client
    |   `-- client.key
    |-- data.db
    |-- loader
    |   `-- keys
    |       |-- btest.key
    |       `-- test.key
    |-- logs
    |   |-- error
    |   `-- info
    |-- plugins
    |   |-- CommandLine32.dll
    |   |-- CommandLine64.dll
    |   |-- Executer32.dll
    |   |-- Executer64.dll
    |   |-- Grabber32.dll
    |   |-- Grabber64.dll
    |   |-- Info32.dll
    |   |-- Info64.dll
    |   |-- Jumper32.dll
    |   |-- Jumper64.dll
    |   |-- ListProcess32.dll
    |   |-- ListProcess64.dll
    |   |-- NetSession32.dll
    |   |-- NetSession64.dll
    |   |-- Screenshot32.dll
    |   |-- Screenshot64.dll
    |   |-- extra
    |   |-- mimikatz32.dll
    |   `-- mimikatz64.dll
    |-- server.exe
    |-- server.ini.xml
    |-- x64
    |   `-- SQLite.Interop.dll
    |-- x86
        `-- SQLite.Interop.dll
```

Readme.txt

The English translation of some important items of the “readme.txt” file, which indicates the changes from the first version of the malware to the “1.6.3” version and contains the build instructions, is as follows. (Original text in Russian is omitted.)

> client 1.6.3
> [+] The result of ADRecon work is saved in the database in the loader from which it was launched, when the tab is called again, the data loaded automatically
> [+] Added a form for launching the script ps2x.py (PsExec).
> server 1.5
> [+] Added support for executing scripts from the ThirdScripts folder
>
> client 1.5
> [+] Added plugin NetSession. The plugin collects information about the computers connected to the computer where the loader is running.
>
> client 1.4
> [+] added plugin info. In the context menu, select Info and after a while in the Info field there will be the user name, domain and version of Windows
>
> client 1.3.3
> [+] The “Get passwords” button has been added to the mimikatz plugin
>
> client 1.3.2
> [+] Added support for RDP grabber.
> client 1.3
> [+] added plugin mimikatz.
> [+] added grabber plugin.
>
> server 1.2:
> [*] updated data transfer protocol
> [+] added AV definition, for this there must be an AV.lst file in the server folder
>
> loader:
> [*] updated data transfer protocol
> [+] sending local
>
> server 1.1:
> [+] - added support for the jumper plugin
>
> client 1.1
> [+] - added support for the jumper plugin

Loader Component

This component of the malware that will run on victim systems is about 9kb in size and runs commands from the server. When the attacker wants to run a function on the device in the victim, the related plugin file containing this function is loaded reflectively on the victim device and filelessly executed it .

Network traffic between server and loader is encrypted with the key determined during the build phase. The following image contains the relevant encryption algorithm.

```
private void Encode(DynamicBuffer src, DynamicBuffer dst, HandlerLoader.Key key)
{
    dst.Clear();
    dst.PrepareAdd(src.Len + 4);
    dst.M[0] = src.M[0];
    Rand.GenBytes(dst.M, 1, 4);
    int num = 0;
    for (int i = 1; i < src.Len; i++)
    {
        dst.M[4 + i] = (src.M[i] ^ dst.M[num + 1]);
        num = (num + 1) % 4;
    }
    num = 0;
    dst.Len = src.Len + 4;
    for (int j = 1; j < dst.Len; j++)
    {
        byte[] m = dst.M;
        int num2 = j;
        m[num2] ^= key.M[num];
        num = (num + 1) % key.M.Length;
    }
}
```

PswInfoGrabber

It is a DLL file responsible for stealing and reporting sensitive information from the target system, especially browser and mail passwords. It was determined that the attackers also used this tool independently from Tirion Loader. In the image below, a screenshot of the logs collected by the malware is included.

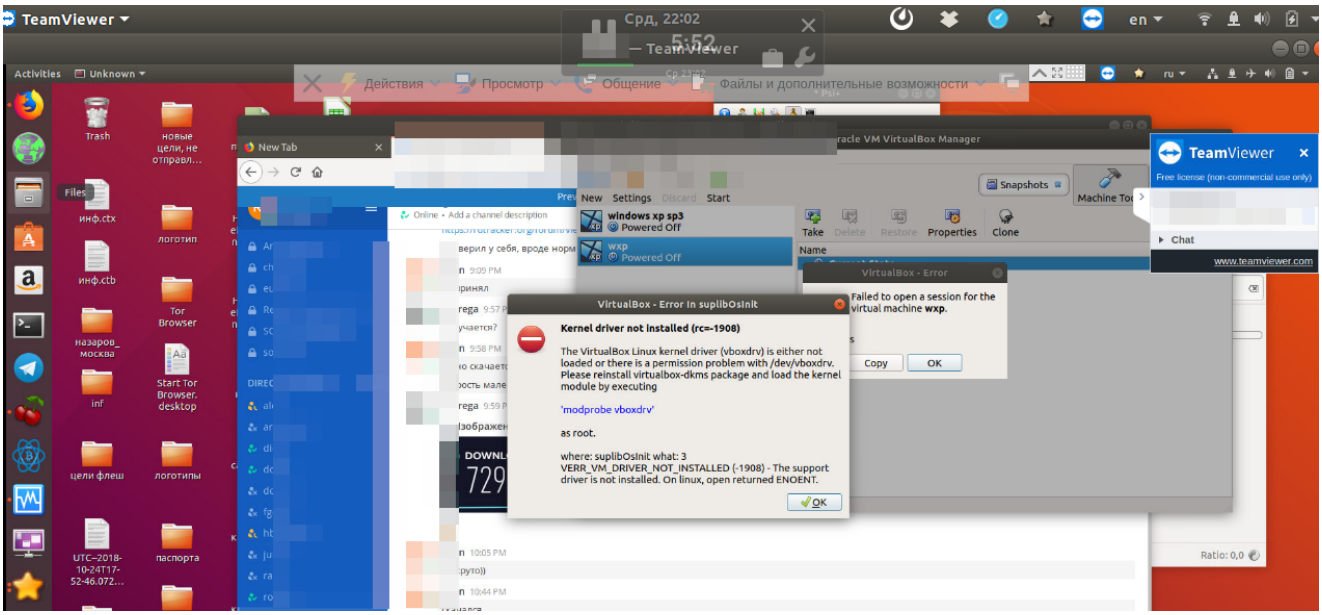
```
################################################################################################################
Chrome Profile: [Default]
Chrome History DB: C:\Users[redacted]Manager\AppData\Local\Google\Chrome\User Data\Default\History
Chrome History DB Copy: C:\Users[redacted]\AppData\Local\Temp\~hdb1570746275575.tmp
################################################################################################################

2019-09-26 02:49:52 1   https://www.google.com/search?q=[redacted]&aqs=chrome..69i60j69i57j012j69i6012.2261j0j4&sourceid=chrome&ie=UTF-8
2019-09-26 02:50:00 6   https://www[redacted]m/restaurant-employee-scheduling-software
2019-09-26 15:29:10 1   https://www.indeed.com/jobs?q=&l=J[redacted]%2C+TN&ts=[redacted]q=1&fromage=last&newcount=339
2019-09-26 15:29:47 2   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 15:30:27 2   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:42:49 3   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:43:24 3   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:44:09 3   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:50:22 3   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:54:53 2   https://employers.indeed.com/c#candidates/view?id=[redacted]sort=date
2019-09-26 16:56:41 1   file:///C:/Users/[redacted]Manager/Downloads/[redacted].pdf
```

opBlueRaven | Outro & End of Part I

In the first edition of these series; we wanted to provide an intro towards our operation by comparing the latest Carbanak toolkit, as discovered by PTI, to older versions that have been publicly accessible.

In the next article, we will be diving deeper into the TTPs of the attackers by also providing references from actual conversations between them.
Aside, we will also be providing screenshots that have been directly acquired from threat actors' machines. (An exemplary one is given below as a teaser.)



Please feel free to get in touch with us if you have further questions.

Credits: PRODAFT Threat Intelligence Team (PTI), INVICTUS Threat Intelligence Team (ITI) [namely womd, y.a.p., e.b., a.b.c. and slv]
Edited by [k.u.]

PRODAFT & INVICTUS Out!

Appendix: YARA Signatures

```
import "pe"
rule apt_Fin7_Tirion_plugins
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Tirion Loader's plugins. It is used by Fin7 group. Need
manual verification"
        version = "1.0"
        date = "2020-07-22"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "fdc0ec0cc895f5b0440d942c0ab60eedeb6e6dca64a93cecb6f1685c0a7b99ae"

    strings:
        $a1 = "ReflectiveLoader" ascii
        $a2 = "plg.dll" fullword ascii
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 15000 and
(pe.exports("[email protected]@[email protected]") or
            pe.exports("[email protected]@[email protected]"))
}

rule apt_Fin7_Tirion_PswInfoGrabber
{

    meta:
```

```
        author = "Yusuf A. POLAT"
        description = "Tirion Loader's PswInfoGrabber plugin. It is used by Fin7
group."
        version = "1.0"
        date = "2020-07-22"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "e7d89d1f23c2c31e2cd188042436ce6d83dac571a5f30e76cbbcdfaf51e30ad9"

    strings:
        $a1 = "IE/Edge Grabber Begin" fullword ascii
        $a2 = "Mail Grabber Begin" fullword ascii
        $a3 = "PswInfoGrabber" ascii
        $a4 = "Chrome Login Profile: '"
        $a5 = "[LOGIN]:[HOST]:"
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 150KB
}

rule apt_Fin7_Tirion_loader
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Tirion Loader's loader component. It is used by Fin7 group."
        version = "1.0"
        date = "2020-07-22"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "e7d89d1f23c2c31e2cd188042436ce6d83dac571a5f30e76cbbcdfaf51e30ad9"

    strings:
        $a1 = "HOST_PORTS" fullword ascii
        $a2 = "KEY_PASSWORD" fullword ascii
        $a3 = "HOSTS_CONNECT" ascii
        $a4 = "SystemFunction036"
        $a5 = "ReflectiveLoader"
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 15KB
}

rule apt_Fin7_Carbanak_keylogplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's keylogger plugin. It is used by Fin7
group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "db486e0cb94cf2bbe38173b7ce0eb02731ad9a435a04899a03d57b06cecddc4d"

    strings:
        $a1 = "SA45E91.tmp" fullword ascii
        $a2 = "%02d.%02d.%04d %02d:%02d" fullword ascii
        $a3 = "Event time:" fullword ascii
        $a4 = "MY_CLASS" fullword ascii
        $a5 = "RegisterRawInputDevices" fullword ascii

    condition:
```

```
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 15000
}

rule apt_Fin7_Carbanak_procmonplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's process monitoring plugin. It is used by
Fin7 group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "3bf8610241a808e85e6ebaac2bb92ba4ae92c3ec1a6e56e21937efec71ea5425"

    strings:
        $a1 = "[%02d.%02d.%04d %02d:%02d:%02d]" fullword ascii
        $a2 = "%s open %s" fullword ascii
        $a3 = "added monitoring %s" fullword ascii
        $a4 = "pm.dll" fullword ascii
        $a5 = "CreateToolhelp32Snapshot" fullword ascii

    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 10000
}

rule apt_Fin7_Carbanak_hdplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's hidden desktop plugin. It is used by
Fin7 group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "39b545c7cd26258a9e45923053a5a64c9461470c3d7bfce3be1c776b287e8a95"

    strings:
        $a1 = "hd%s%s" fullword ascii
        $a2 = "Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Advanced"
fullword ascii
        $a3 = "StartHDServer" fullword ascii
        $a4 = "SetThreadDesktop" fullword ascii
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 15000
}
rule apt_Fin7_Carbanak_hvncplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's hvnc plugin. It is used by Fin7 group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "40ce820df679b59476f5d277350dca43e3b3f8cac7ec47ad638371aaa646c315"

    strings:
```

```
        $a1 = "VncStartServer" fullword ascii
        $a2 = "VncStopServer" fullword ascii
        $a3 = "RFB 003.008" fullword ascii
        $a4 = "-nomerge -noframemerging" fullword ascii
        $a5 = "--no-sandbox --allow-no-sandbox-job --disable-3d-apis --disable-gpu
--disable-d3d11" fullword wide
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 300000
}

rule apt_Fin7_Carbanak_vncplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's vnc plugin. It is used by Fin7 group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "ecf3679f659c5a1393b4a8b7d7cca615c33c21ab525952f8417c2a828697116a"

    strings:
        $a1 = "VncStartServer" fullword ascii
        $a2 = "VncStopServer" fullword ascii
        $a3 = "ReflectiveLoader" fullword ascii
        $a4 = "IDR_VNC_DLL" fullword ascii
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 400000
}

rule apt_Fin7_Carbanak_rdpplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's rdp plugin. It is used by Fin7 group"
        version = "1.0"
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "0d3f1696aae8472145400d6858b1c44ba7532362be5850dae2edbd4a40f36aa5"

    strings:
        $a1 = "sdbinst.exe" fullword ascii
        $a2 = "-q -n \"UAC\"" fullword ascii
        $a3 = "-q -u \"%s\"" fullword ascii
        $a4 = "test.txt" fullword ascii
        $a5 = "install" fullword ascii
        $a6 = "uninstall" fullword ascii
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 400000
}

rule apt_Fin7_Carbanak_switcherplugin
{

    meta:
        author = "Yusuf A. POLAT"
        description = "Carbanak backdoor's switcher plugin. It is used by Fin7
group"
        version = "1.0"
```

```
        date = "2020-07-21"
        reference = "https://threatintelligence.blog/"
        copyright = "PRODAFT"
        SHA256 = "d470da028679ca8038b062f9f629d89a994c79d1afc4862104611bb36326d0c8"

    strings:
        $a1 = "iiGI1E05.tmp" fullword ascii
        $a2 = "oCh4246.tmp" fullword ascii
        $a3 = "inf_start" fullword ascii
        $a4 = "Shell_TrayWnd" fullword ascii
        $a5 = "ReadDirectoryChangesW" fullword ascii
        $a6 = "CreateToolhelp32Snapshot" fullword ascii
    condition:
        uint16(0) == 0x5A4D  and (all of ($a*)) and filesize < 15000
}
```