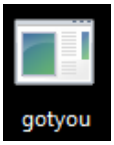
==========

GOTYOU.EXE ANALYSIS PAPER

THIS IS AN EXAMINATION OF MALWARE DURING/AFTER EXECUTION

==========

FILE EXAMINED



File examined: **GotYou.exe**
md5 hash: 8d2ce6396379b74c573346bb951252a7
File type: **EXE**
File size: **660 KB**
Description: **No Description**
Download Source: https://app.any.run/tasks/360049d0-3e28-43cb-9cfe-9c5e6d4bfe0a/

==========

PURPOSE OF EXAMINATION

Determine the behavior of the malware

==========

SUMMARY

1. Virus Total reports **56/70** detect this file as malware
2. Disk and Registry Alert reported a newly created directory in the user's profile at AppData\Local where an executable file was dropped
3. Process Explorer detects that the process takes almost 100% CPU usage
4. Process Monitor reveals the persistence mechanism used by the malware and the command line argument being used in a child process
5. Wireshark detects network communication with a mining pool

==========

EXAMINATION DETAILS
This examination began on June 21, 2021 on a Windows 7 VM running the following analysis programs:

1. Disk and Registry Alert - capturing changes made to disk
2. Process Monitor- capturing disk and registry activity in real-time
3. Process Explorer - monitor the process, look at TCP/IP tab
4. Wireshark - capturing network activity
5. API Monitor - monitoring for API calls
6. PE Studio - detects suspicious indicators
7. Bandicam - video capturing malware detonation

I first started with strings.

```
\Software\Microsoft\Windows\CurrentVersion\RunOnce
taskmgr.exe
svchost.exe
\KnownDlls\ntdll.dll
100%CPU
```

So there is a hint that the malware will establish persistence on the machine as indicated by "\Software\Microsoft\Windows\CurrentVersion\RunOnce". This registry key is used to launch the malware upon login to the system but only executes once and then the registry entry is deleted from RunOnce. More commonly I see "Run" instead of "RunOnce". So while this will cover its tracks it won't be able to run after the initial reboot unless the malware has established persistence elsewhere (didn't see CurrentVersion\Run in strings).

Taskmgr.exe is an interesting string because this most likely means that it will interact with Taskmgr.exe or behave a certain way while Taskmgr.exe is running. Same with the "svchost.exe" string. Next referring to the "\KnownDlls\ntdll.dll" I find that the KnownDlls is a mechanism in Windows to cache frequently used dlls. Codeproject.com states the following about KnownDlls, "Initially, it was added to accelerate application loading, but also can be considered as a security mechanism, as it prevents malware from putting Trojan versions of system DLLs to the application folder...." They continue "We cannot say that this security mechanism is very strong...but still it helps to protect the system." Source: https://www.codeproject.com/Articles/325603/Injection-into-a-Process-Using-KnownDllsSo

I am curious if the malware intends to create a trojanized version of ntdll.dll. The next string is 100%CPU which seems likely that this sample will crank up the CPU

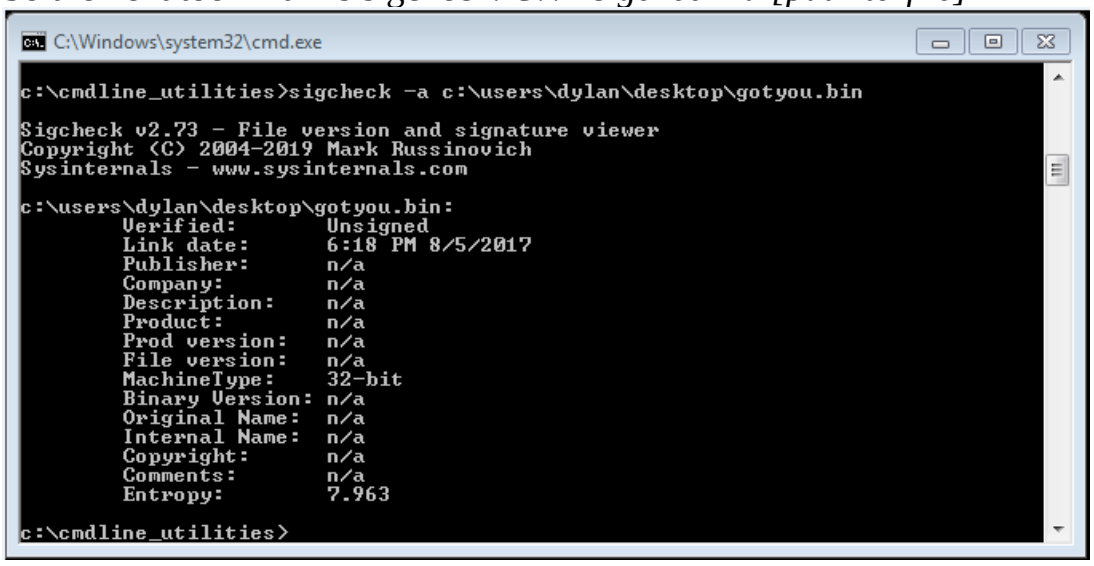
on the machine and this is commonly done with malicious bitcoin miners to make their owners as much money as possible. So I think that is interesting to mention because I would think an average user would notice their computer is running slow so what would they do? Reboot the machine. Once they do the persistence mechanism is gone and this would erase its tracks that it ever had persistence on the machine. Again unless there is another entry in the registry or elsewhere in Windows.

I also launched PE studio which does an excellent job at showing suspicious strings.

encoding (2)	size (bytes)	file-offset	blacklist (34)	hint (82)	group (11)	value (12537)
ascii	16	0x000043DF	x	import	security	OpenProcessToken
ascii	10	0x00004403	x	import	security	IsValidSid
ascii	16	0x00004273	x	import	reckoning	GetThreadContext
ascii	17	0x00004260	x	import	memory	ReadProcessMemory
ascii	16	0x0000438F	x	import	keyboard-and-mouse	GetLastInputInfo
ascii	11	0x00004124	x	import	execution	OpenProcess
ascii	16	0x0000414B	x	import	execution	TerminateProcess
ascii	18	0x00004161	x	import	execution	GetExitCodeProcess
ascii	23	0x0000417C	x	import	execution	SetThreadExecutionState
ascii	24	0x00004229	x	import	execution	CreateToolhelp32Snapshot
ascii	14	0x0000423B	x	import	execution	Process32First
ascii	13	0x0000424C	x	import	execution	Process32Next

GetLastInputInfo is an interesting string because it checks for the last time input was detected on a system. So anytime you move the mouse, press a key on the keyboard, etc. So malicious bitcoin miners will look for this because it can evade detection by idle mining your resources while it is likely no one is sitting at the computer.

So the next tool I run is sigcheck. *C:\> sigcheck -a [path-to-file]*



As you can see from the output the file is unsigned and no company name, description, product, product version, etc. The entropy is 7.963 which is very high meaning it has been compressed and/or encrypted. In this case I don't think it is encrypted otherwise I wouldn't be able to see all the juicy strings as discussed.

WHAT DOES VIRUS TOTAL SAY?

This file had to be uploaded by me on June 20th, 2021 because it was never analyzed by Virus Total. Virus Total reports **56/70** anti-virus engines detected this file as malware. Webroot and F-Secure have went undetected on this file. The consensus classification for this malware between the AV engines is CoinMiner.

Virus Total lists some of the imports which can reveal the malware's capabilities:
ADVAPI32.dll -- GetTokenInformation, OpenProcessToken, RegSetValueExW

So, with the API call, GetTokenInformation the malware can find out the privileges of a process that is running. Malware will typically want administrator permissions to a process so that it has more control of a system. OpenProcessToken is used to essentially get the token handle of a process which would indicate the process it wants to target. RegSetValue is an indicator that it will make a modification in the registry so I will be looking for that.

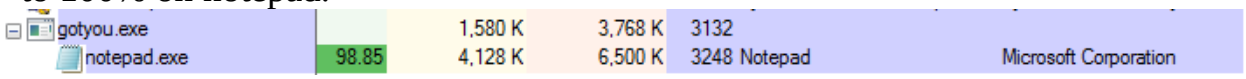
KERNEL32.dll -- CreateToolhelp32Snapshot, GetSystemInfo, GetTickCount, GetWindowsDirectory, Process32First, Process32Next, ReadProcessMemory, WriteFile, TerminateProcess, ResumeThread, SetThreadExecutionState, CreateProcess, Sleep

Most of these API calls do exactly what they sound like but the ones I want to point are Process32First, Process32Next and CreateToolhelp32Snapshot. These three calls are used in malware to locate a process to be used in a process injection scenario. CreateToolhelp32Snapshot is a call that is used to create a snapshot of processes that are running on a system and Process32First/Process32Next enumerate through the processes. Then the malware typically uses CreateProcess to retrieve the process ID of the target process. So, there is a possibility that there will be some process injection involved during the execution of this malware.

There are no comments from the Virus Total community on this file.

PROCESS EXPLORER
Here is some behavior I noticed while watching gotyou.exe in Process Explorer

1. Launches notepad.exe as a child process which then cranks the CPU almost to 100% on notepad.



2. When you launch taskmgr.exe notepad is immediately terminated and gotyou.exe remains dormant waiting for taskmgr.exe to be closed, once you terminate taskmgr.exe it launches notepad.exe again and gets right back to work

3. If you kill notepad.exe in Process Explorer, gotyou.exe appears to respawn notepad.exe again, if you terminate gotyou.exe, notepad.exe will still continue to max out the CPU but can easily terminate and will not respawn as gotyou.exe is no longer present.

   PS C:> Stop-Process -Name gotyou ; Stop-Process -Name notepad
   *terminating both processes at the same time works as well

4. Even when the malware doesn't have an Internet connection it will still attempt to mine for cryptocurrency and max out the CPU on a machine

5. If you try to run gotyou.exe in Sandboxie it will not launch

DISK AND REGISTRY ALERT
The following files were added to the infected system:
Added -> c:\users\dylan\appdata\local\khylfepyyr
Added -> c:\users\dylan\appdata\local\khylfepyyr\svchost.exe

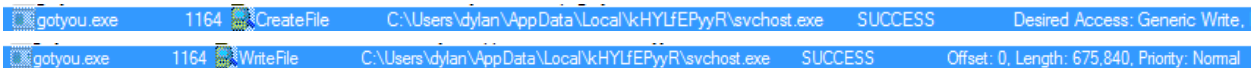
So this is pretty typical place that malware likes to hide on a system from what I have been seeing. This location is where the average user probably won't venture to, so it makes sense. The svchost.exe md5 hash is the same as gotyou.exe indicating it is the same file. Sometimes malware will delete itself in the location where it was detonated but gotyou.exe did not. A folder that is named as random characters I think is also fairly common as well with malware.

PROCESS MONITOR

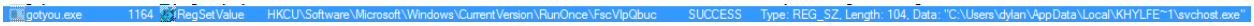
So here is what is going on behind the scenes as far as registry and file system activity:

GotYou.exe/Notepad.exe

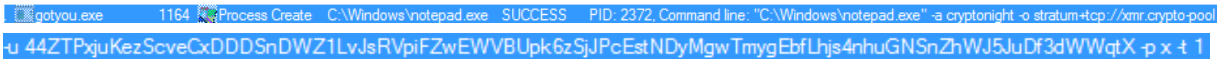
1. Creates a new directory and drops a copy of itself in c:\users\[username]\AppData\Local\kHYLfEPyyR\ with a file name of svchost.exe and writes 660KB to the file (same size as gotyou.exe)



2. Establishes persistence on the machine using the registry at HKCU\Microsoft\Windows\CurrentVersion\RunOnce\FscVIpQbuc and assigning this key to the path for the svchost.exe file

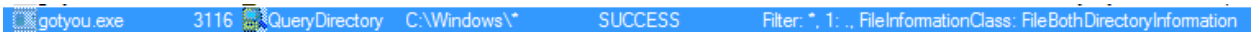


3. Launches notepad.exe with the following argument: "C:\Windows\notepad.exe" -a cryptonight -o stratum+tcp://xmr.crypto-pool.fr:80 -u 44ZTPxjuKezScveCxDDDSnDWZ1LvJsRVpiFZwEWVBUpk6zSjJPcEstNDyMgwTmygEbfLhjs4nhuGNSnZhWJ5JuDf3dWWqtX -p x -t 1

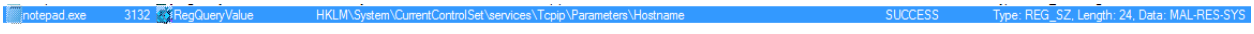


*-a cryptonight -> this is the mining algorithm utilized, cryptonight is suitable for ordinary PCs
*-o stratum+tcp://xmr.crypto-pool.fr:80 -> this is the mining pool
*-u -> mining pool address
*-p x -> password, yes the password is "x"
*-t -> apparently is threads

4. Queries the c:\Windows\ directory (GetWindowsDirectory)



5. Grabs the hostname of the infected machine



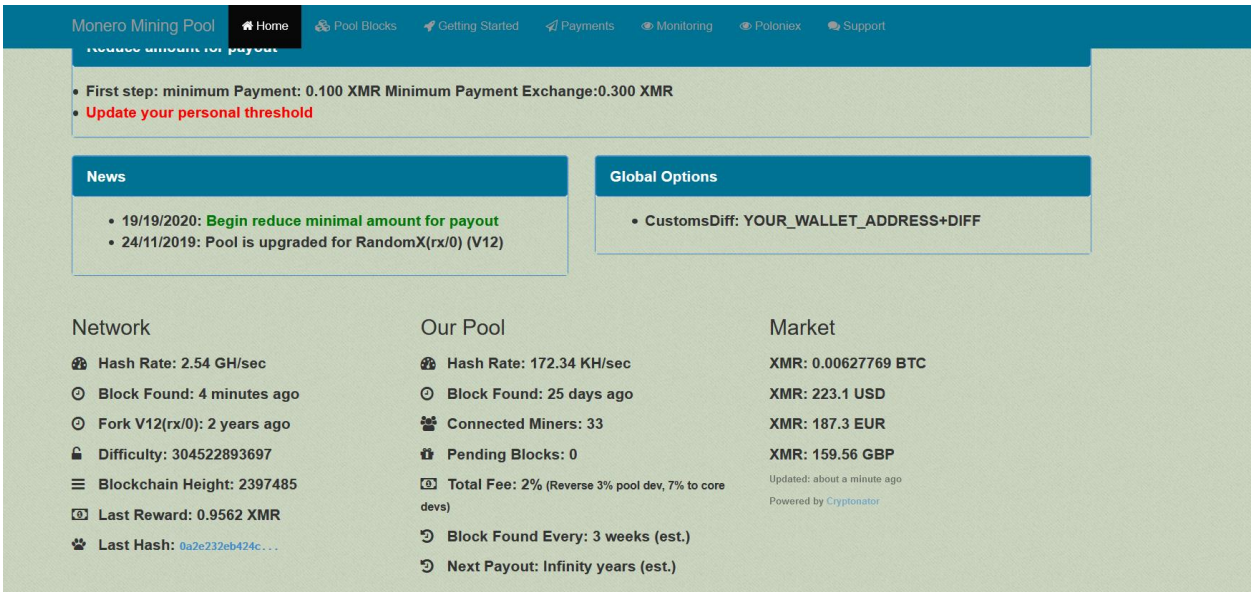
WIRESHARK

Network communication begins with a DNS request for xmr.crypto-pool.fr and the DNS server responds with the IP address 163.172.226.137. The TCP handshake is initiated over port 80 with this IP address.

{"method": "login", "params": {"login": "44ZTPxjuKezScveCxDDDSnDWZ1LvJsRVpiFZwEWVBUpk6zSjJPcEstNDyMgwTmygEbfLhjs4nhuGNSnZhWJ5JuDf3dWWqtX", "pass": "x", "agent": "cpuminer-multi/0.1"}, "id": 1}
{"id":1,"jsonrpc":"2.0","error":null,"result":{"id":"924558085848346","job":{"height":2397477,"blob":"0e0ec1ae878706aae161a6cb5d7bcc15e47d29fd964045c39d2a3d896bbf28ead2b0b4d07eb60900000000d53f175b23db3436daca707c71456a6558c80135097bbb910f635683e74befc00c","job_id":"877750753159619","seed_hash":"883985df67dda82ed5eadbd886b94867e3587692a9f1d9c6275d35f50f232bef","target":"11a40300","algo":"rx/0"},"status":"OK"}}

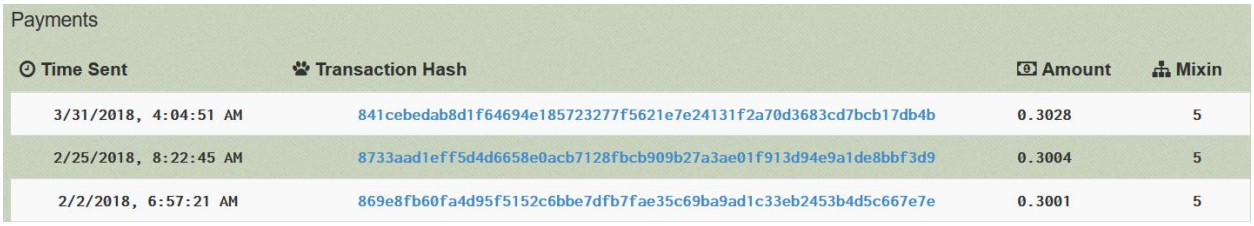
Once the TCP handshake is complete the client initiates a login request to the address pool using the credentials in the login parameter (address pool = "44ZTP...." and password of "x"). The server then responds with the mining job instructions. The height is how many blocks are in the chain, seed_hash is the password hash used for authentication. After this every minute or so a new job is sent.

If at any point taskmgr.exe is launched on an infected machine network communication also immediately ends. Task Manager for the win! The below screenshot is the xmr.crypto-pool.fr landing page.



If you plug in the pool address captured in Wireshark into this website you can lookup the current status of this pool as shown below.

The pending balance equates to $9.47 (USD). The total paid amount equates to $11,216 (USD). The last transaction was on March 31, 2018 but goes back to April 26th, 2017. From April 26th, 2017 to March 31st, 2018 there has had frequent activity as far as payments.



INDICATORS OF COMPROMISE

File System Compromise

c:\users\[username]\appdata\local\khylfepyyr\svchost.exe

Registry Compromise

HKCU\Microsoft\Windows\CurrentVersion\RunOnce\FscVIpQbuc = Data: c:\users\[username]\AppData\Local\KHYLFE~1\svchost.exe

Network Compromise

xmr.crypto-pool.fr (163.172.226.137) -> port 80

YARA RULE

```
rule evasive_btc_miner {

    meta:
        Author = "u/dkaye_mal_anst18"
        Date = "7-4-2021"
        Md5 = "8d2ce6396379b74c573346bb951252a7"
        Reference = "https://app.any.run/tasks/360049d0-3e28-43cb-9cfe-9c5e6d4bfe0a/"

    strings:
        $taskMgr =  "taskmgr.exe"  // detects if taskmgr.exe is running, if it is then immediately terminates
        $highCPU = "100%CPU"  // high CPU usage
        $lastInput = "GetLastInputInfo"  // malware is looking for the last time input was detected
        $processInj1 = "CreateToolhelp32Snapshot"  // takes a snapshot of processes running on a system
        $processInj2 = "Process32First"  // starts the enumeration of processes, run in conjunction with CreateToolhelp32Snapshot
        $processInj3 = "Process32Next"  // combs through processes running on system to obtain info about said processes
        $processInj4 = "NtMapViewOfSection" // carves out section of memory to insert code into
```

```
        $processInj5 = "NtWriteVirtualMemory" // writes code to the selected
        process
        $processInj6 = "NtResumeThread" // resumes the thread and executes the
        code

    condition:
        $taskMgr or $highCPU or $lastInput or $processInj1 or $processInj2 or
        $processInj3 or $processInj4 or $processInj5 or $processInj6

        }
```

==========

END ANALYSIS

==========

Author: u/dkaye_mal_anst18