GhostMiner: Cryptomining Malware Goes Fileless

blog.minerva-labs.com/ghostminer-cryptomining-malware-goes-fileless

- Tweet
-

Cybercriminals are increasingly relying on malicious cryptominers as a way of making money online, often shifting from using ransomware or diversifying revenue streams.

Though in late 2017 these activities were relatively niche, as illustrated by the case of the WaterMiner, 2018 has shown the use of increasingly aggressive tactics to deploy malicious miners, including the use of advanced exploit kits.

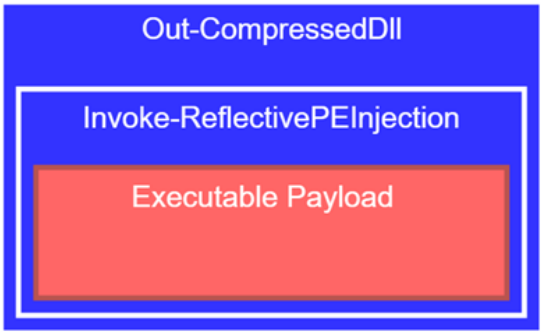
Security vendors reacted by improving their detection capabilities; however, as we have seen in the past, cyber criminals remain one step ahead of the defenders, this time shifting to fileless techniques in order to remain undetected.

This post describes a recent attack Minerva's research team dissected, dubbed GhostMiner, after our solution prevented this infection at a customer site. It provides an example of how malicious miners are evolving to use advanced fileless techniques to succeed in mining Monero and spreading silently on a global scale. In this attack, we also witnessed how competing miners are fighting each other to generate more income for themselves, removing other miners on the endpoint. Minerva Labs analyzed the attack and presents a novel way of turning the tables on mining attackers by using their scripts to remove competitors, against them.

Dissecting GhostMiner – How it Spreads and Mines

The Use of Fileless Evasion Frameworks

The core activity of GhostMiner's components was performed by a compiled malicious Windows executable. To stay undetected, the executable relied on a couple of nested PowerShell evasion frameworks - Out-CompressedDll and Invoke-ReflectivePEInjection, which employed fileless techniques to conceal the presence of the malicious program.

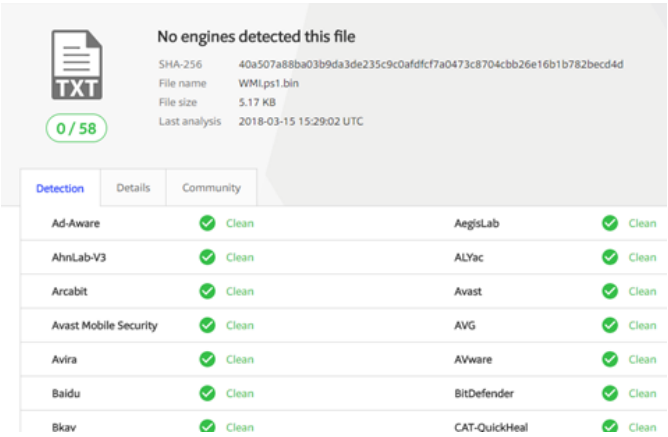


Two PowerShell frameworks wrap the payload which is unpacked directly into the memory at runtime

Each component was launched from a different PowerShell script:

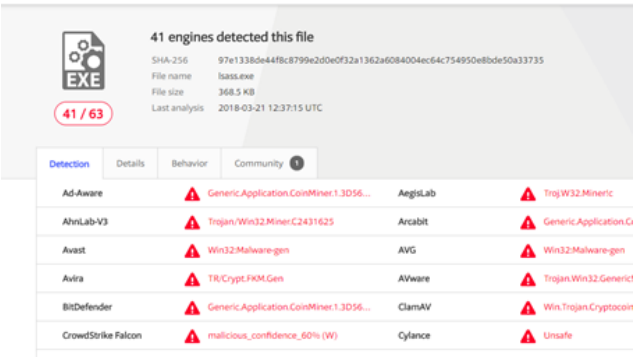
- *ps1*: in charge of propagation by infecting new victims,
- *ps1* (or *WMI64.ps1* on x64 machines): mining Monero cryptocurrency

This evasive approach was highly effective at bypassing many security tools: some of the payloads analyzed were fully undetected by all the security vendors:



WMI.ps1 malicious miner goes undetected by all security vendors, analysis results as of March 21st 2018

This amazing result can be compared with the compiled executable below from the same actors, implementing the same functionality of the miner hidden within WMI.ps1. Once the miner doesn't use a fileless technique, 41 vendors now detect the malicious payload:



Miner now detected once fileless techniques are not used

Minerva's Memory Injection Prevention thwarted the fileless attack from both spreading and mining as it tried to unpack in memory.

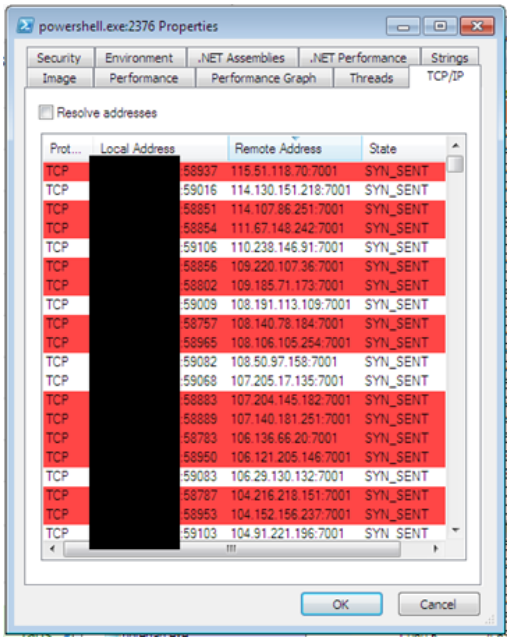
Spreading GhostMiner

Neutrino.ps1 infects new victims, seeking and attacking servers running the following applications:

- Oracle's WebLogic (using CVE-2017-10271, similar case already reported by FireEye)
- MSSQL

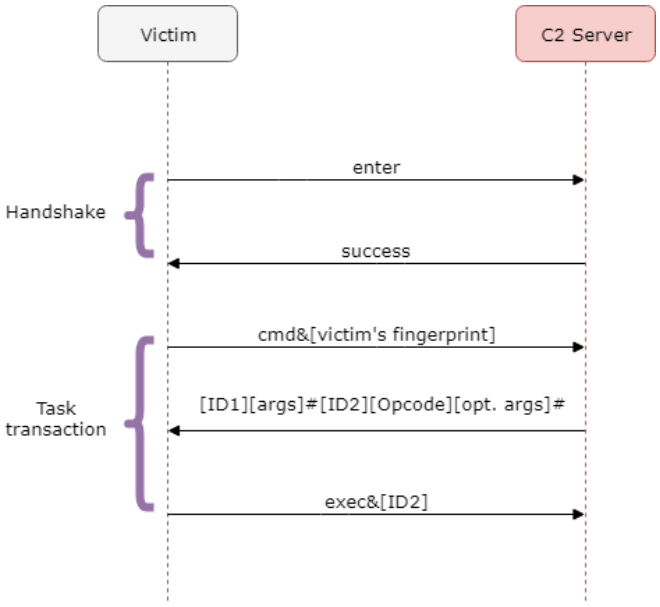
- phpMyAdmin

Despite the potential of targeting several applications, the attacks analyzed only aimed to exploit WebLogic servers. The malware accomplished this by randomly probing IP addresses, creating numerous new TCP connections each second with the expectation of eventually finding a vulnerable target.



The distribution component of the miner scans for exploitable WebLogic servers on TCP port 7001

To avoid detection by network security tools, this component of the attack communicates with its C2 server over HTTP by encoding requests and replies in Base64. Messages are exchanged using a protocol which includes a simple hand shake followed by a request to perform various tasks, such as, infect other servers or take screenshots. Once the task is completed, the client will report to the C2 and request another task:



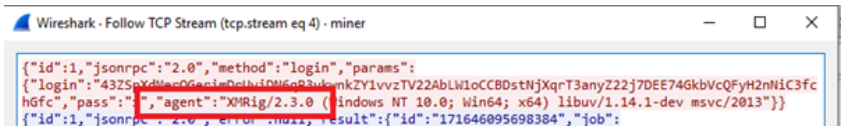
In the following example, the original encoded message was blurred for confidentiality reasons; however, we added in the core elements of the decoded content to display key aspects of the message. The request (in red) includes a request for a new task with identifiers of the infected endpoint, the response (in blue) orders the malware to initiate a random hunt for exploitable WebLogic servers:

HTTP request and response example which demonstrates how the threat was spread e.g. a task to scan and exploit WebLogic servers

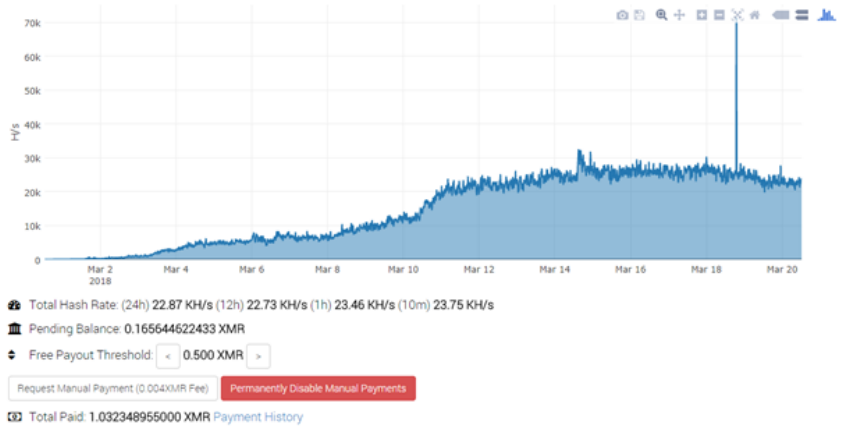
Note the referrer header is hard-coded: *qq.com* which is one of the most popular Chinese websites. This, alongside other indicators we found, suggests that the attacker crafted the malicious HTTP requests to hide in organizations with Chinese-speaking users.

How Mining was Performed in GhostMiner

The mining component itself, as mentioned above, is launched directly from memory using the evasion frameworks described above. It is a slightly customized version of the open source XMRig miner, as we observed in its outgoing traffic:



During the time of analysis, the operation was already running for roughly three weeks:



Attackers' address stats, courtesy of http://www.minexmr.com

As of the date of this publication, the accumulated sum in the XMR wallet associated with this campaign was a mere 1.03 XMR, equivalent to just over $200. However, it is highly plausible that there are other addresses used in this campaign, undetectable due to Monero's anonymity features.

Another potential explanation for the low "revenues" of the GhostMiner campaign is the aggressive rivalry between mining gangs. There are plenty of potential victims, but the exploits and techniques they use are public. The attackers are aware that their competitors share the same toolset and try to infect the same vulnerable machines.

Eliminating Malicious Mining Competitors

The battle on exploitable endpoints is fierce. This miner starts its operation only after eliminating the competition–killing any miner on the endpoint that they can identify. The sample we analyzed implemented a wide range of techniques, some weren't reported before, including:

- Kill running miners using PowerShell's "Stop-Process -force" command, detecting it by a hard-coded blacklist
- Stop and delete miner blacklisted services by name using *exe*
- Remove miners that run as blacklisted scheduled tasks by the task name using *exe*
- Stop and removie miners by their commandline arguments, e.g. if one of the arguments contains the word *"*cryptonight**"*–using WMI and PowerShell
- Stop and remove miners by going through the list of established TCP connections, looking for ports associated with miners (the data is collected using *exe*)

```
$CmdLine = Get-WmiObject -Class Win32_Process | Where-Object {$_.CommandLine -like '*cryptonight*' -Or $_
-Or $_.CommandLine -like '*--donate-level*' -Or $_.CommandLine -like '*--max-cpu-usage*' -Or $_.CommandLi
-Or $_.CommandLine -like '*pool.electroneum.hashvault.pro*'}

if ($CmdLine -ne $Null)
{
    $PathArray = @()
    foreach ($m in $CmdLine)
    {
        $evid = $($m.ProcessId)
        if (($evid -eq $PId) -or ($evid -eq $minerPId)) { continue }
        Write-Host "[i] Miner PId: $evid"
        Get-Process -Id $evid | Stop-Process -Force


        $Path = $($m.Path)
        if ($Path -eq "$Env:WinDir\System32\cmd.exe" -Or $Path -eq "$Env:WinDir\SysWOW64\cmd.exe" -Or $Pa
        -Or $Path -eq "$Env:WinDir\Notepad.exe") { continue }
        if ($PathArray -NotContains $Path) { $PathArray+=$Path }
```

Detecting and removing other miners by their arguments

In the past, malware researchers such as Xavier Mertens (@xme) suggested that defenders can use scripts similar to the "competitors killer" we've found as an IOC list to detect systems compromised by malicious miners.

We suggest taking it a bit further and consider running a slightly modified version of the "killer script" as an aid for incident response teams. We believe that it is a good starting point for writing one's own PowerShell script for removing malicious miners. The script is provided in Minerva's research team's GitHub account.

It implements all the aforementioned tactics – removing known processes, tasks and services by name and unfamiliar ones by arguments or TCP connections typical to miners.

Note that MinerKiller is provided as-is with no liability, use it only if you know what you are doing! It might stop or even delete important processes in some cases!

Interested to learn more about what we're doing here at Minerva to protect endpoints?

Book a demo at RSA 2018, we'll be at booth #2329, Moscone South or come hear us as BSides San Francisco on April 15th where our VP Research and co-founder, Omri Moyal will be talking about the Rise of Coinminers**.**

Request a Demo Today!

IOCs

- C2 IP Address:
  123[.]59[.]68[.]172

- Hashes (SHA-256)
  - Neutrino.ps1:
    - 4b9ce06c6dc82947e888e919c3b8108886f70e5d80a3b601cc6eb3752a1069a1
    - 9a326afeeb2ba80de356992ec72beeab28e4c11966b28a16356b43a397d132e8
  - WMI.ps1:
    40a507a88ba03b9da3de235c9c0afdfcf7a0473c8704cbb26e16b1b782becd4d
  - WMI64.ps1:
    8a2bdea733ef3482e8d8f335e6a4e75c690e599a218a392ebac6fcb7c8709b52
  - Associated Monero address:
    43ZSpXdMerQGerimDrUviDN6qP3vkwnkZY1vvzTV22AbLW1oCCBDstNjXqrT3anyZ22j7DEE74GkbVcQFyH2nNiC3fchGfc

- “Killer” script:
  - Service names
    - xWinWpdSrv
    - SVSHost
    - Microsoft Telemetry
    - lsass
    - Microsoft
    - system
    - Oracleupdate
    - CLR
    - sysmgt
    - gm
    - WmdnPnSN
    - Sougoudl
    - Nationaaal
    - Natimmonal
    - Nationaloll
  - Task names
    - Mysa
    - Mysa1
    - Mysa2
    - Mysa3
    - ok
    - Oracle Java
    - Oracle Java Update
    - Microsoft Telemetry
    - Spooler SubSystem Service
    - Oracle Products Reporter
    - Update service for products
    - gm
    - ngm
  - Process names
    - msinfo
    - xmrig*
    - minerd
    - MinerGate
    - Carbon
    - yamm1
    - upgeade
    - auto-upgeade
    - svshost
    - SystemIIS
    - SystemIISSec
    - WindowsUpdater*
    - WindowsDefender*
    - update
    - carss
    - service
    - csrsc
    - cara
    - javaupd
    - gxdrv
    - lsmosee

- Miner related server side TCP ports
  - 1111
  - 2222
  - 3333
  - 4444
  - 5555
  - 6666
  - 7777
  - 8888
  - 9999
  - 14433
  - 14444
  - 45560
  - 65333
  - 55335
- Miner related command line arguments
  - *cryptonight*
  - *stratum+*
  - *--donate-level*
  - *--max-cpu-usage*
  - *-p x*
  - *pool.electroneum.hashvault