

# Deep Malware and Phishing Analysis

By Joe Security LLC

Archived: 2026-04-06 00:37:31 UTC



In [Joe Sandbox Cloud Basic](#), our community version of Joe Sandbox, we often get very interesting and recent malware samples. On the September 16th, 2020 we came across a new GuLoader variant (MD5: 01a54f73856cfb74a3bbba47bcec227b). GuLoader is a malware loader well known for its anti-evasion techniques.

## Slow VM Exits

The initial analysis on a virtual machine showed the following results:

A screenshot of the JoeSandbox Cloud analysis report interface. The report title is "Analysis Report New Inquiry 90383873777721102029393003938.exe". The interface is divided into four main sections: "General Information", "Detection", "Signatures", and "Classification".  
- **General Information:** Shows sample name, analysis ID, MD5, SHA1, and SHA256 hashes. Below this is a "Most Interesting Screenshot" showing a Windows desktop.  
- **Detection:** A vertical bar indicates the detection level: MALICIOUS (red), SUSPICIOUS (orange), CLEAN (green), and UNKNOWN (grey). Below this, a table shows: Score: 76, Range: 0 - 100, Whitelisted: false, Confidence: 100%.  
- **Signatures:** A list of detection signatures, including "Antivirus / Scanner detection for submitted sample", "MultiAV Scanner detection for submitted file", "Potential malicious icon found", "Tries to detect sandboxes and other dynamic analysis tool...", "Tries to detect virtualization through RDTSC time measure...", "Tries detected VDI Downloader Generic", "Abnormal high CPU Usage", "Antivirus or Machine Learning detection for unpacked file", "Contains functionality for executive linking, often used to...", "Contains functionality to call native functions", "Contains functionality to read the PEB", "PE file contains strange resources", "Sample file is different than original file name gathered fr...", and "Uses code obfuscation techniques (call, push, ret)".  
- **Classification:** A circular radar chart showing various detection categories and their scores.

As we can see in the Signature section, there are some RDTSC based evasion checks executed:



instructions like RDTSC. This difference is measured and used to decide if the loader is going to execute the payload or not.

## **Instruction Hammering**

Secondly, the measurements are not executed once but executed thousands of times. The result is an overall delay which often exceeds the execution time on a sandboxed analyzer. As a result, the payload execution is never reached. This method of executing massive amounts of delay instructions to prevent the execution - also known as **Instruction Hammering** - is very similar to [API hammering](#), a technique we saw in TrickBot and many other malware samples.

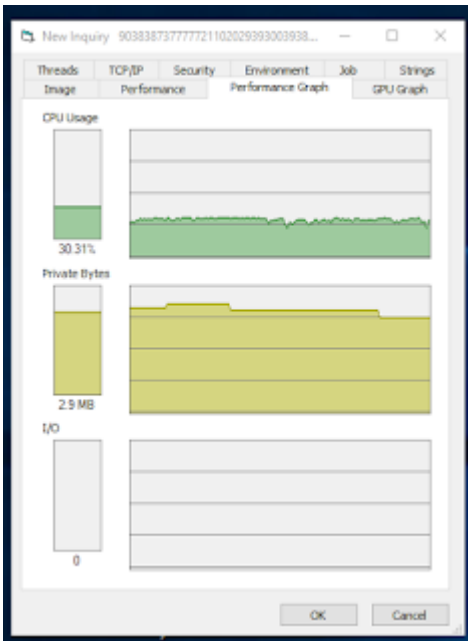
Instruction Hammering is extremely powerful since it is hard to detect and challenging to bypass, as it exploits the architecture of virtualization. The GuLoader creators seem to have noticed that, and in the new version they have even increased the number of delay instructions being executed:

## Disassembly:

```
0: 0f 31          rdtsc
2: b8 01 00 00 00 mov  eax,0x1
7: 0f a2          cpuid
9: 61            popa
a: e8 03 00 00 00 call 0x12
f: 0f ae e8      lfence
12: 8b 15 14 00 fe 7f mov  edx,DWORD PTR ds:0x7ffe0014
18: 0f ae e8      lfence
1b: c3            ret
1c: 29 f2          sub  edx,esi
1e: c3            ret
1f: 66 39 c8      cmp  ax,cx
22: 85 c2          test edx,eax
24: 59            pop  ecx
25: 01 d7          add  edi,edx
27: 49            dec  ecx
28: 66 39 d1      cmp  cx,dx
2b: 83 f9 00      cmp  ecx,0x0
2e: 75 e6          jne  0x16
30: 51            push ecx
31: 66 39 c0      cmp  ax,ax
34: e8 26 00 00 00 call 0x5f
39: e8 15 00 00 00 call 0x53
3e: 0f ae e8      lfence
41: 8b 15 14 00 fe 7f mov  edx,DWORD PTR ds:0x7ffe0014
47: 0f ae e8      lfence
4a: c3            ret
4b: 89 d6          mov  esi,edx
4d: 60            pusha
4e: 0f 31          rdtsc
```

This code executes RDTSC and CPUID 11 million times. In addition, *UserSharedData.SystemTime* is being used for time measurements.

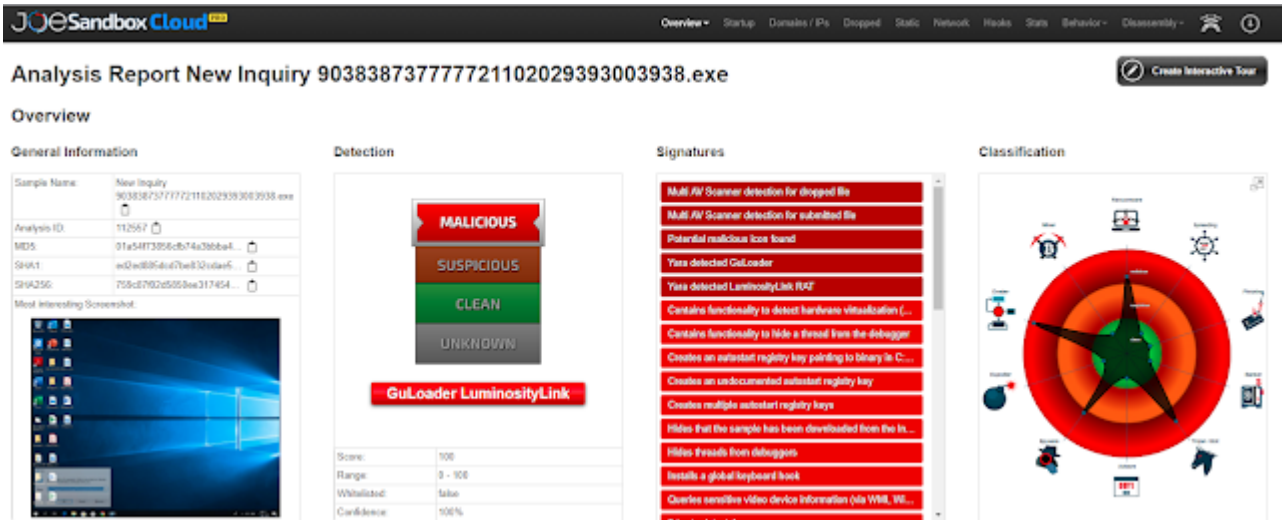
On a Windows 10 x64 system running on VirtualBox the delay loop takes several minutes to finish:



On real hardware, the loop is executed in under one second!

## Bare Metal Analysis to the Rescue

Joe Sandbox is one of a few vendors offering analysis on bare metal. In this setup, the malware sample is run on a real physical machine. Physical machines are much closer to the real target of the malware. As a result, VM-based evasions don't work and the sandbox can catch and record the real payload. If we analyze GuLoader on bare metal the delay loop is passed in under a second and we can see that the LuminosityLink RAT is dropped:



## Startup

- System is w10x64native
- New Inquiry 90383873777721102029393003938.exe (PID: 6652 cmdline: "C:\Users\user\Desktop\New Inquiry 90383873777721102029393003938.exe" MD5: 01A54F73856CFB74A3BBBA47BCEC227B)
- RegAsm.exe (PID: 1576 cmdline: "C:\Users\user\Desktop\New Inquiry 90383873777721102029393003938.exe" MD5: 6AFAE79556E125202DCF1D3FE74A3638)
- RegAsm.exe (PID: 1440 cmdline: "C:\Users\user\Desktop\New Inquiry 90383873777721102029393003938.exe" MD5: 6AFAE79556E125202DCF1D3FE74A3638)
- conhost.exe (PID: 1584 cmdline: "C:\Windows\system32\conhost.exe 0x00000000 -ForceV1 MD5: C221707E5CE93515AC87507E19181E2A)
- anitbcnt.exe.exe (PID: 4188 cmdline: "C:\ProgramData\782401\anitbcnt.exe.exe" MD5: 6AFAE79556E125202DCF1D3FE74A3638)
- conhost.exe (PID: 1500 cmdline: "C:\Windows\system32\conhost.exe 0x00000000 -ForceV1 MD5: C221707E5CE93515AC87507E19181E2A)
- demispherekledskene.exe (PID: 1812 cmdline: "C:\Users\user\Hustankesgy3\demispherekledskene.exe" MD5: 01A54F73856CFB74A3BBBA47BCEC227B)
- RegAsm.exe (PID: 4328 cmdline: "C:\Users\user\Hustankesgy3\demispherekledskene.exe" MD5: 6AFAE79556E125202DCF1D3FE74A3638)
- conhost.exe (PID: 4084 cmdline: "C:\Windows\system32\conhost.exe 0x00000000 -ForceV1 MD5: C221707E5CE93515AC87507E19181E2A)
- RegAsm.exe (PID: 2652 cmdline: "C:\Windows\Microsoft.NET\Framework\v2.0.50727\RegAsm.exe" MD5: 6AFAE79556E125202DCF1D3FE74A3638)
- conhost.exe (PID: 5952 cmdline: "C:\Windows\system32\conhost.exe 0x00000000 -ForceV1 MD5: C221707E5CE93515AC87507E19181E2A)
- demispherekledskene.exe (PID: 2140 cmdline: "C:\Users\user\Hustankesgy3\demispherekledskene.exe" MD5: 01A54F73856CFB74A3BBBA47BCEC227B)
- cleanup

The full analysis report of the GuLoader variant is [available here](#).

---

Source: <https://www.joesecurity.org/blog/3535317197858305930>