Shortcut to Windows Update

cyberandramen.net/2022/02/04/shortcut-to-windows-update/

February 4, 2022

Summary

On January 27, 2022, Malwarebytes Labs shared an article covering new tactics including abusing the Windows Update Client for code executing believed to be the work of Lazarus.

The purpose of this post will be to cover possible detection points for defenders to identify adversaries misusing the Windows Update Client.

Please give the blog post by Ankur Saini and Hossein Jazi a read. The information for the piece of malware this blog will focus on is below:

Filename	File Size	SHA256	Reference
wuaueng.dll	227.48KB	829eceee720b0a3e505efbd3262c387b92abdf46183d51a50489e2b157dac3b1	https://blog.malwarebytes.com/threat-intelligence/2022/01/north-koreas-lazarus-apt-leverages-windows-update-client-github-in-latest-campaign/

Table 1

Shortcut to Success

Upon opening the malicious document, a chain of code injections occur and numerous files are dropped to disk. Persistence is achieved via the Word document creating a hidden C:\Windows\System32 and dropping a shortcut file (LNK) to the Startup folder.

The purpose of this shortcut file named windowsupdateconf.lnk, is to execute the malicious DLL identified above in Table 1, which is embedded with an additional payload that will communicate with the C2.

wuauclt.exe which resides at C:\Windows\System32 is intended to be used via the command line and accepts a number of options, good luck finding worthwhile documentation though.

More common options that may be seen used with wuauclt.exe are:

- /detectnow
- /resetauthorization
- /reportnow

The following is to achieve code execution in the LNK file's target path: "*C:\Windows\system32\wuauclt.exe" /UpdateDeploymentProvider C:\Windows\system32\wuaueng.dll /RunHandlerComServer*".

Indicators/Detection

Image 1: Sysmon Event 1, Process Creation

Looking at the above process creation event output, a few items should jump out to defenders.

- The Current Directory is the startup folder as opposed to cmd.exe.
- The command line options “/UpdateDeploymentProvider” and “/RunHandlerComServer” are used to load a DLL. This should be considered suspicious and warrant a closer look.

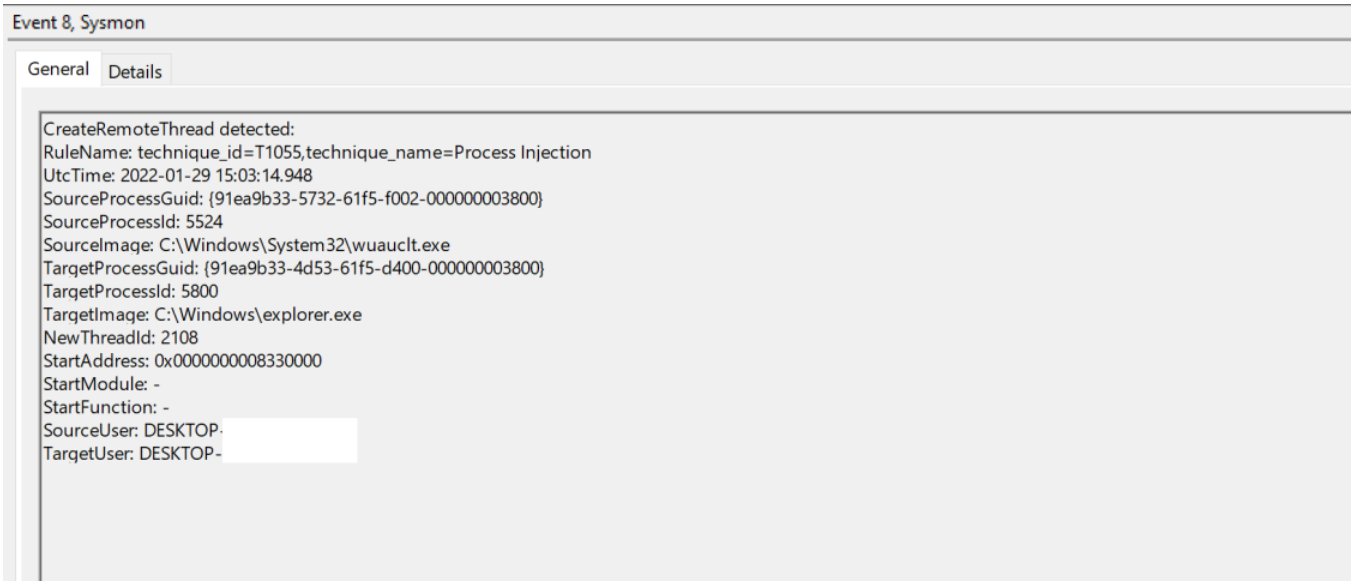

Image 2: Sysmon CreateRemoteThread detected

As discussed earlier, this particular malware makes use of code injection at numerous points during the execution flow. As we can see in Image 2, wuauclt has injected itself into the explorer.exe process.

After running a few tests in a lab environment, I could not identify a legitimate purpose for wuauclt to inject into explorer.

Identifying the unusual command-line arguments in Splunk is very simple and can be completed in one line. This is an extremely basic rule, and an additional check for the DLL from well-known paths used by adversaries could be used as well.

```
sysmon EventID 1 AND ParentCommandLine "/UpdateDeploymentProvider" AND "/RunHandlerComServer" AND ParentImage NOT
"C:\\Windows\\System32\\cmd.exe"
```

Image 3: Splunk query for suspicious wuauclt arguments

After reading a recent CrowdStrike blog on Microsoft Protection Logging, or MPLog, I wanted to see what events if any may be detected during the execution of the LNK file.

Unfortunately, Microsoft doesn’t provide a great deal of documentation on the event types in MPLog but in addition to the possible detection areas identified by CrowdStrike, the below event is also worth mentioning.

```
38044  2022-01-29T15:03:14.965Z Engine:Process C:\Windows\System32\wuauclt.exe is tainted. TaintType:0x1
38045
38046  BEGIN BM telemetry
38047  GUID:{12D73544-69C1-5260-5336-1D6529057FAC}
38048  SignatureID:59043914424653
38049  SigSha:26869454c04e58691f5d2fb7b6753044110f3921
38050  ProcessID:5524
38051  ProcessCreationTime:132879421948393011
38052  SessionID:1
38053  CreationTime:01-30-2022 00:03:14
38054  ImagePath:C:\Windows\System32\wuauclt.exe
38055  Taint Info:Friendly: Y; Reason: 1,; Modules: C:\Windows\System32\wuaueng.dll:15,; Parents: C:\Windows\explorer.exe:5800:1,
38056  Operations:None
38057  END BM telemetry
```

Image 4: MPLog BM telemetry

The BM or Behavior Monitoring telemetry as the name implies captures suspicious activity on workstations for additional analysis. Your environment may vary, but the only time a BM event was initiated in my lab was when the LNK was run.

This is out of the scope of this blog, but forwarding some of the more useful MPLog events to a SIEM could likely be a useful addition to identifying compromise on a system.

Enough of Sysmon and logs, let's take a look at the DLL file.

```
md5       | 490c885dc7ba0f32c07ddfe02a04bbb9
sha1      | 294690c1aee8dc7723858dafcb2a0ed273296641
sha256    | 829eceee720b0a3e505efbd3262c387b92abdf46183d51a50489e2b157dac3b1
os        | windows
format    | pe
arch      | amd64
path      | c:\users\target1\downloads\mals\wuauclt.dll
```

ATT&CK Tactic	ATT&CK Technique
DEFENSE EVASION	Obfuscated Files or Information:: T1027
DISCOVERY	Process Discovery:: T1057
	Software Discovery:: T1518

MBC Objective	MBC Behavior
DATA	Check String:: [C0019]
	Encode Data::Base64 [C0026.001]
EXECUTION	Install Additional Program:: [B0023]

CAPABILITY	NAMESPACE
reference Base64 string	data-manipulation/encoding/base64
contain a resource (.rsrc) section	executable/pe/section/rsrc
contain an embedded PE file	executable/subfile/pe
enumerate processes	host-interaction/process/list

Image 5: capa output of wuaueng.dll

Using Mandiant's capa tool, we can get a quick overview of the capabilities of the DLL. In the output, capa confirms what we already know, that the DLL file contains an embedded PE file and installs an additional program.

Interesting Strings

– {"message":"Commit changed details","content":"Q29tcGxldGVkIHN1Y2Nlc3NmdWxseQ=="}

– Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 Safari/537.36

– Accept: application/vnd.github.v3+json

– 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

– api.github.com

– metafiles/tmp%.4d%.2d%.2d%.2d%.2d%.2d.txt

I think this is probably a good place to stop my analysis to avoid going deeper down a rabbit hole.

Hope you enjoyed reading!

More Info

Yara Rule

```
rule APT_LAZARUS_WINUPDATE_DLL {
    meta:
    description = "Detect malicious DLL executed via the Windows Update Client."
    author = "Michael Rippey"
    reference = "https://blog.malwarebytes.com/threat-intelligence/2022/01/north-koreas-lazarus-apt-leverages-windows-update-
client-github-in-latest-campaign/"
    date = "2022-02-06"
    hash = "829eceee720b0a3e505efbd3262c387b92abdf46183d51a50489e2b157dac3b1"


    strings:
        $n1 = "api.github.com" fullword ascii
        $n2 = "repos/%s/%s/contents/%s" fullword ascii
        $n3 = "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 Safari/537.36"
fullword ascii
        $n4 = "metafiles/tmp%.4d%.2d%.2d%.2d%.2d%.2d.txt" fullword ascii

        $e1 = "{\"message\":\"Commit changed details\",\"content\":\"Q29tcGxldGVkIHN1Y2Nlc3NmdWxseQ==\"}" fullword ascii
        $e2 = "CreateRemoteThreNtProtectVirtualWriteProcessMemoRtlCreateUserThr" fullword ascii
        $e3 = "omni callsig" fullword ascii
        $e4 = " Type Descriptor'" fullword ascii
        $e5 = "$Sectigo Public Code Signing Root R460" fullword ascii
        $e6 = "Sleef" fullword ascii

    condition:
        uint16(0) == 0x5A4D and
        (any of ($n*) or 3 of ($e*))
        and filesize < 228KB
}
```