



Malware Persistence Methods

Information Security Inc.

Contents

- About Malware Persistence
- Why persistence?
- Persistence methods
- Persistence method: hijacking extensions handlers
- Demo
- References

About Malware persistence



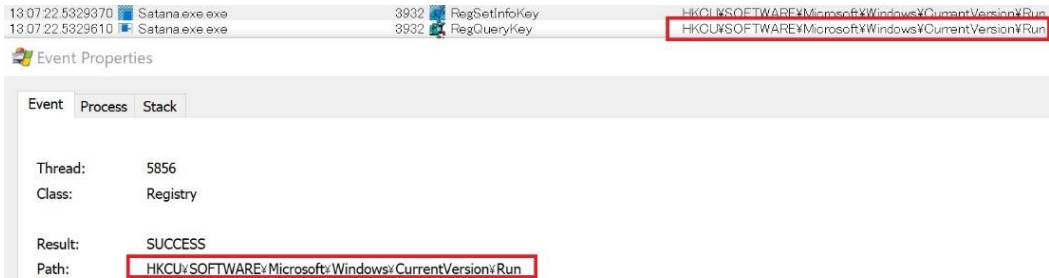
Why persistence?

- Who? Most of the malware needs it
- Why? To start the application after each reboot
- How? Windows offers a few legitimate persistence ways

Persistence methods

- Registry keys

- ▲ HKCU\Software\Microsoft\Windows\CurrentVersion\Run

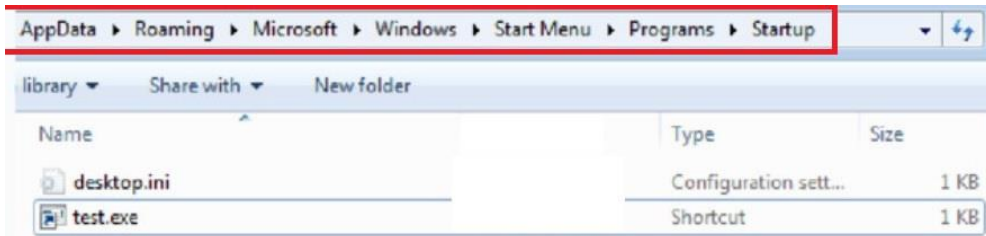


- ▲ HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce

- ▲ HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run

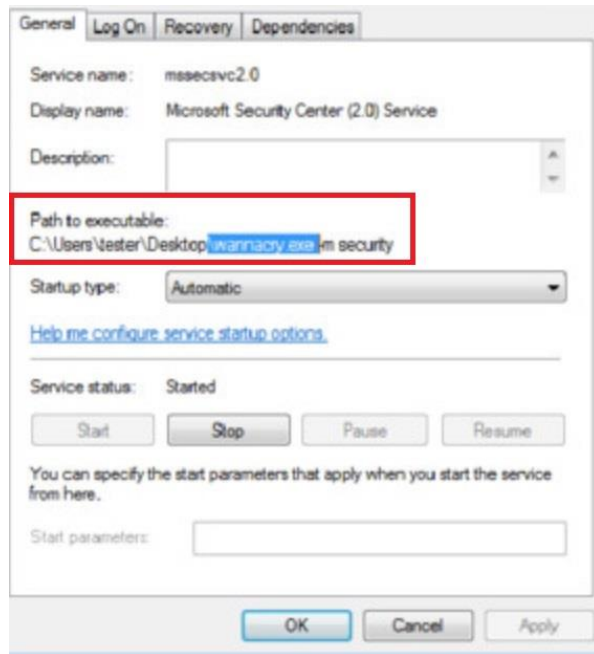
Persistence methods

- Startup link



Persistence methods

- System Services



Persistence method: hijacking extensions handlers

- How the extension handling works? (Windows 10 x64)

▲ On Windows, extensions that are known by the operating system are defined in the registry. An *.html* extension

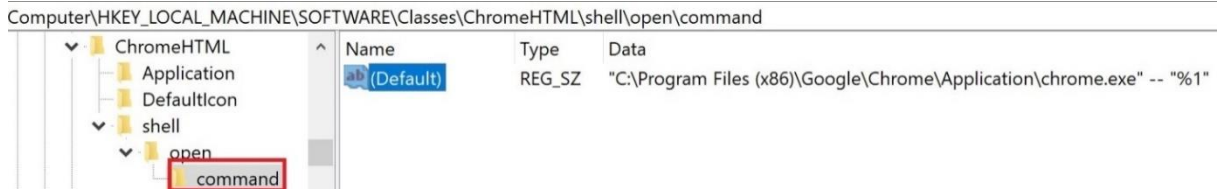
Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.html\UserChoice

Name	Type	Data
(Default)	REG_SZ	(value not set)
Hash	REG_SZ	o6tCR3z2Goc=
ProgId	REG_SZ	ChromeHTML

▲ html files are handled by ChromeHTML

Persistence method: hijacking extensions handlers

- How the extension handling works? (Windows 10 x64)
 - ▲ The most important handler feature is “command”



- ▲ The command defines what action has to be taken when the file with the particular extension is clicked

Persistence method: hijacking extensions handlers

- How the extension handling works? (Windows 10 x64)

▲ Handler command: a closer look

```
/* C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" -- "%1" */
```

The above command runs chrome.exe with one parameter. (%1) is the name of the file that was clicked. Thanks to this, Chrome opens the clicked file

Demo

- Demo environment: Windows 10 x64

Edition	Windows 10 Pro
Version	1703
OS Build	15063.608
System type	64-bit operating system, x64-based processor

Demo

- Overwrite the command feature of ChromeHTML


Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Classes\ChromeHTML\shell\open\command

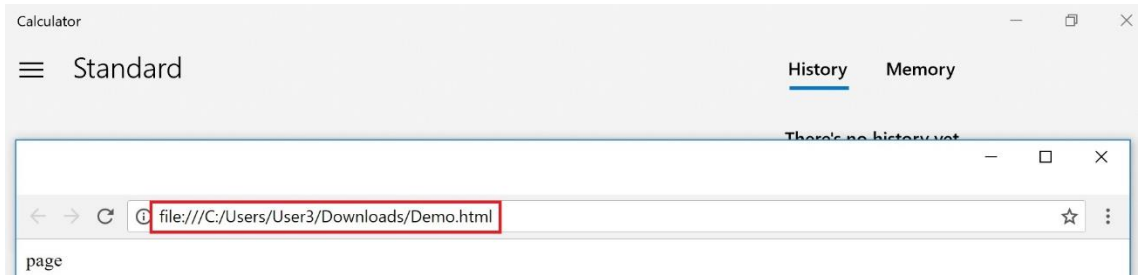
Name	Type	Data
(Default)	REG_SZ	C:\ProgramData\ProxyApp.exe "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" -- "%1

Demo


- The .html extension is not handled by chrome.exe, but by the ProxyApp.exe – that deploys chrome.exe, but also calc.exe (attacker can implement a malicious app here)
- From the point of view of the user nothing has changed – chrome opens the document as it was before – but in the background another application starts to run

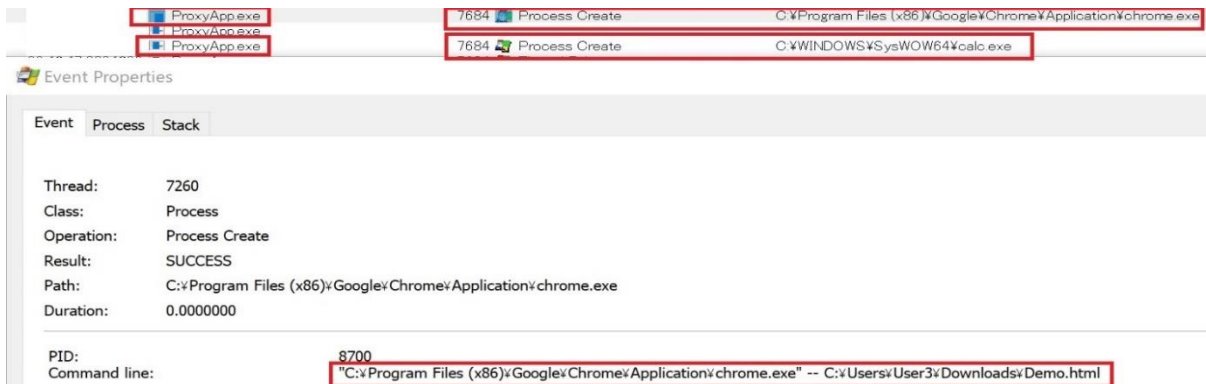
Demo

- The .html extension is not handled by chrome.exe, but by the ProxyApp.exe – that deploys chrome.exe, but also calc.exe. Clicking on Demo.html ( Demo.html) opens calc.exe and chrome.exe



Demo

- The .html extension is not handled by chrome.exe, but by the ProxyApp.exe – that deploys chrome.exe, but also calc.exe. Clicking on Demo.html ( Demo.html) opens calc.exe and chrome.exe



The screenshot displays the Windows Task Manager and Event Viewer. In the Task Manager, three instances of ProxyApp.exe are visible, each with a red box around its name. Below, the Event Viewer shows two 'Process Create' events. The first event, with PID 7684, shows the creation of chrome.exe at the path C:\Program Files (x86)\Google\Chrome\Application\chrome.exe. The second event, also with PID 7684, shows the creation of calc.exe at the path C:\WINDOWS\SysWOW64\calc.exe. The Event Viewer details for the chrome.exe event are shown below, with the command line highlighted in a red box: "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" -- C:\Users\User3\Downloads\Demo.html

Event	Process	Stack
Thread:	7260	
Class:	Process	
Operation:	Process Create	
Result:	SUCCESS	
Path:	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe	
Duration:	0.0000000	
PID:	8700	
Command line:	"C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" -- C:\Users\User3\Downloads\Demo.html	

References

- MSDN

<https://msdn.microsoft.com/ja-jp/library/aa767914%28v=vs.85%29.aspx>

- Mitre

<https://attack.mitre.org/wiki/Technique/T1042>