

Product ▾Solutions ▾Resources ▾Open Source ▾Enterprise ▾Pricing

🔍

Sign in

Sign up

📄 Neo23x0 / Raccine Public

🔔 Notifications

🍴 Fork 122

★ Star 945

<> Code

🕒 Issues 21

🔗 Pull requests

🎬 Actions

🛡 Security


📊 Insights

🔗 main ▾

🔗📁

🔍 Go to file

<> Code ▾

 **Neo23x0** Merge pull request [#134](#) from luis261/f... 🗨️ ✓ 6858a9f · last year 🕒 530 Commits

📁 .github/workflows	Update test-cases.yml	4 years ago
📁 GPO	Update Raccine.ADMX	3 years ago
📁 RaccineGUI/RaccineCfg	fix: regex for newline chars	4 years ago
📁 data_samples	Create mordor_raccine_simulation_mo...	4 years ago
📁 images	desc: new eventlog screenshot for REA...	4 years ago
📁 reg-patches	feat: kill the killer	3 years ago
📁 robot-tests	fix: robot tests detecting benign messa...	4 years ago
📁 scripts	fix: smartscreen block - manual rules u...	4 years ago
📁 sigma	refactor: adjusted Sigma rule to Raccine...	4 years ago
📁 source	refactor: no disabled GUI, uninstall refa...	3 years ago
📁 tests/Raccine-Test	update test entry points	4 years ago
📁 yara	fix: FPs with Terminus SSH client	2 years ago
📄 .gitignore	Added test project	4 years ago
📄 LICENSE	Initial commit	4 years ago
📄 README.md	Fix README typos	last year
📄 Raccine.apis	feat: app icon	4 years ago
📄 Raccine.ico	style: new icon with transparent backgr...	4 years ago
📄 Raccine.sln	moved files to new directory	4 years ago
📄 build_dist.bat	style: minor changes	3 years ago
📄 explore.bat	update test entry points	4 years ago
📄 install-raccine.bat	Update install-raccine.bat	3 years ago

Maintenance Level

Inactively Maintained

About

A Simple Ransomware Vaccine

📖 Readme

📄 Unlicense license

📈 Activity

★ 945 stars

👁 43 watching

🍴 122 forks

Report repository

Releases 41

📦 Raccine 1.4.4 Latest








on Jun 1, 2021








+ 40 releases

Packages

No packages published

Contributors 14





Languages

C++ 36.5%

C# 26.8%

Batchfile 26.1%

YARA 7.6%

PowerShell 2.3%

C 0.7%

Page 1 of 9



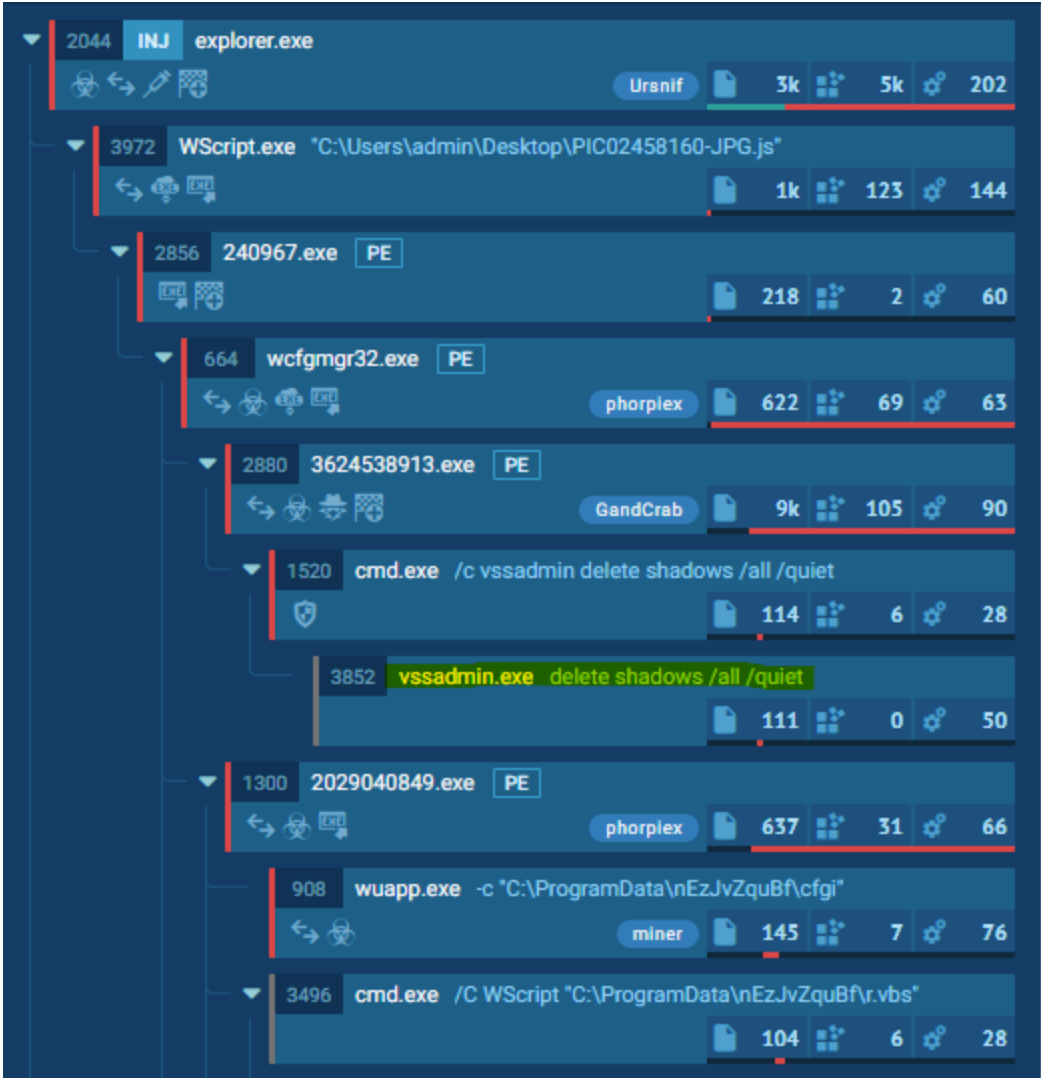
RACCINE

Raccine

A Simple Ransomware Protection

Why

We see ransomware delete all shadow copies using `vssadmin` pretty often. What if we could just intercept that request and kill the invoking process? Let's try to create a simple vaccine.



How it works

We [register a debugger](#) for `vssadmin.exe` (and `wmic.exe`), which is our compiled `raccine.exe`. Raccine is a binary, that first collects all PIDs of the parent processes and then tries to kill all parent processes.

Advantages:

- The method is rather generic
- We don't have to replace a system file (`vssadmin.exe` or `wmic.exe`), which could lead to integrity problems and could break our raccination on each patch day
- Flexible YARA rule scanning of command line params for malicious activity

README
 Unlicense license
 ⋮

- No running executable or additional service required (agent-less)

Disadvantages / Blind Spots:

- The legitimate use of `vssadmin.exe delete shadows` (or any other blacklisted combination) isn't possible anymore
- It even kills the processes that tried to invoke `vssadmin.exe delete shadows` , which could be a backup process
- This won't catch methods in which the malicious process isn't one of the processes in the tree that has invoked `vssadmin.exe` (e.g. via `schtasks`)

The Process

1. Invocation of `vssadmin.exe` (and `wmic.exe`) gets intercepted and passed to `raccine.exe` as debugger (`vssadmin.exe delete shadows` becomes `raccine.exe vssadmin.exe delete shadows`)
2. We then process the command line arguments and look for malicious combinations using Yara rules.
3. If no malicious combination could be found, we create a new process with the original command line parameters.
4. If a malicious combination could be found, we collect all PIDs of parent processes and the start killing them (this should be the malware processes as shown in the screenshots above). Raccine shows a command line window with the killed PIDs for 5 seconds, logs it to the Windows Eventlog and then exits itself.

Malicious combinations:

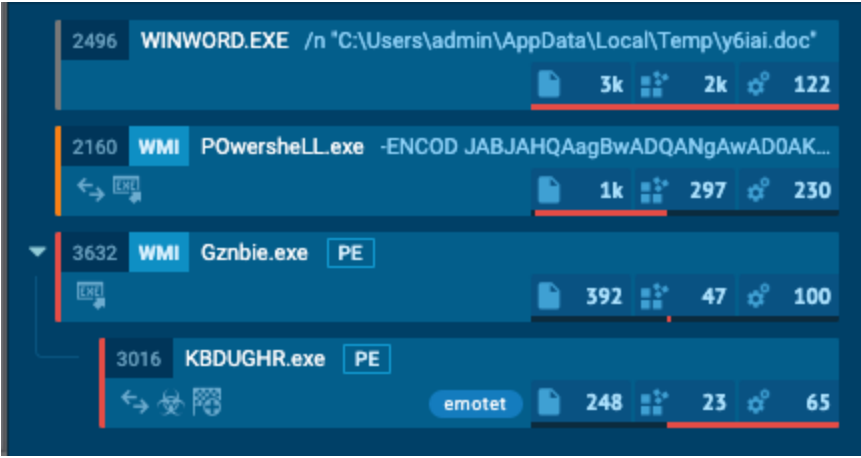
- `delete` and `shadows` (vssadmin, diskshadow)
- `resize` and `shadowstorage` (vssadmin)
- `delete` and `shadowstorage` (vssadmin)
- `delete` and `shadowcopy` (wmic)
- `delete` and `catalog` and `-quiet` (wbadmin)
- `win32_shadowcopy` or element from a list of encoded commands (powershell)
- `recoveryenabled` (bcdedit)
- `ignoreallfailures` (bcdedit)

^ outdated list: check the corresponding [YARA rule](#)

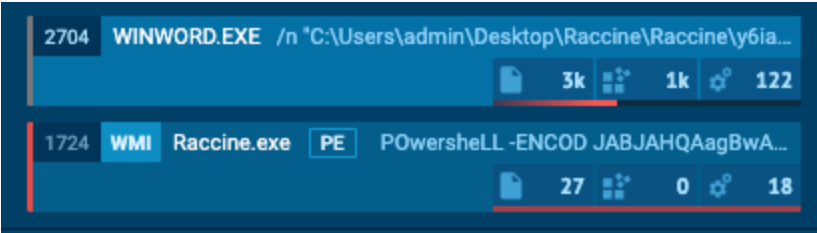
Powershell list of encoded commands: `JAB` , `SQBFAF` , `SQBuah` , `SUVYI` , `cwBhA` , `aWV4I` , `aQB1AHgA` and many more

Example

Emotet without Raccine - [Link](#)



Emotet with Raccine - [Link](#) (ignore the process activity that is related to the Raccine installation)



The infection gets nipped in the bud.

Warning !!!

USE IT AT YOUR OWN RISK!

You won't be able to run commands that use the blacklisted commands on a raccinated machine anymore until you apply the uninstall patch `raccine-reg-patch-uninstall.reg` . This could break various backup solutions that run that specific command during their work. It will not only block that request but kill all processes in that tree including the backup solution and its invoking process.

If you have a solid security monitoring that logs all process executions, you could check your logs to see if `vssadmin.exe delete shadows` , `vssadmin.exe resize shadowstorage ...` or the other blocked command lines are frequently or sporadically used for legitimate purposes in which case you should refrain from using Raccine.

Version History

- 0.1.0 - Initial version that intercepted & blocked all vssadmin.exe executions
- 0.2.0 - Version that blocks only vssadmin.exe executions that contain `delete` and `shadows` in their command line and otherwise pass all parameters to a new process that invokes vssadmin with its original parameters
- 0.2.1 - Removed `explorer.exe` from the whitelist
- 0.3.0 - Supports the `wmic` method calling `delete shadowcopy` , no outputs for whitelisted process starts (avoids problems with wmic output processing)
- 0.4.0 - Supports logging to the Windows Eventlog for each blocked attempt, looks for more malicious parameter combinations
- 0.4.1 - Statically linked binaries
- 0.4.2 - Bugfixes provided by John Lambert
- 0.5.0 - Removed Eventlog logging (basic info was unnecessary; cuased higher complexity; can be achieved by process creation logging as well), support for wbadmin filtering
- 0.5.1 - Improvements by @JohnLaTwC
- 0.5.2 - Additional check for `delete shadowstorage` by @JohnLaTwC, code review by @_hillu, application icon
- 0.5.3 - Batch installer
- 0.6.0 - Additional checks for `bcdedit.exe /set {default} bootstatuspolicy ignoreallfailures` and `bcdedit.exe /set {default} recoveryenabled no`
- 0.7.0 - Additional checks for `powershell.exe` and `win32_shadowcopy` or a list of encoded commands
- 0.7.1 - Improvements by @JohnLaTwC
- 0.7.2 - Using absolute paths in registry patches
- 0.8.0 - Creates a log file with all intercepted requests and actions performed `C:\ProgramData\Raccine_log.txt`
- 0.9.0 - Logs to Windows Eventlog by @JohnLaTwC
- 0.10.0 - Simulation mode only
- 0.10.1 - Fix for Simulation mode
- 0.10.2 - Includes `diskshadow.exe delete shadows` command
- 0.10.3-5 - Minor fixes and additions
- 1.0 BETA - GUI elements and YARA rule scanning of command line params
- 1.1 BETA - YARA rule matching with external variables, troubleshooting functions
- 1.2 BETA - Signature Updater
- 1.3 BETA - In-Memory YARA Scanning of invoking parent process
- 1.4 BETA - Full x86 support, moved static strings to YARA rules to avoid AV detections, Log of accepted executions, .NET Framework setup in installer
- 1.4.2 BETA - Exit code fix (pass through of exit code returned by the intercepted program), intercept taskkill.exe

Installation

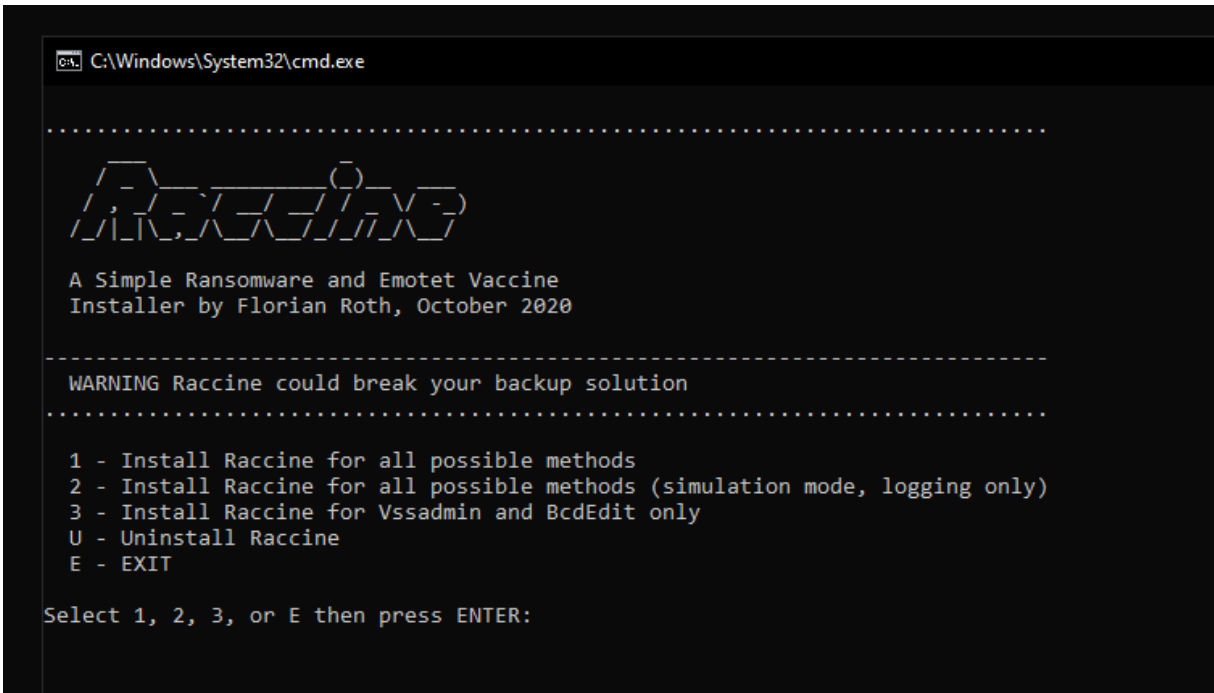
Requirements

- VC++ Runtime for YARA scanning (Installer contains the setup package from https://aka.ms/vs/16/release/VC_redist.x64.exe)
- .NET Framework 4.5
- Internet access for the YARA rule updates
- Windows 7 / Windows 2008 R2 or higher

Both the Visual C++ Redistributable package and the .NET Framework will be automatically installed running `install-raccine.bat`.

Automatic Installation

1. Download `Raccine.zip` from the [Release](#) section
2. Extract it
3. Run `raccine-installer.bat` as administrator



The batch installer includes an "uninstall" option.

Manual Uninstall

As Administrator do:

1. Run `raccine-reg-patch-uninstall.reg`
2. Remove `%ProgramFiles%\Raccine` and `%ProgramData%\Raccine` folders
3. Run `reg delete HKCU\Software\Raccine /F`
4. Run `taskkill /F /IM RaccineSettings.exe`
5. Run `reg delete "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V "Raccine Tray" /F`
6. Run `schtasks /DELETE /TN "Raccine Rules Updater" /F`

Updates

Program Upgrade

We recommend an uninstall and reinstall to upgrade. An uninstall removes all registry keys with configurations.

Signature Update

Raccine has an integrated signature-updater since version 1.2. This program named `RaccineRulesSync.exe` is configured to run once a day via scheduled task. You can run a signature update manually using the option in the tray icon menu.

YARA Matching

Since version 1.0, Raccine additionally uses YARA rules to determine if a process command line or parent process is malicious or not. Raccine uses 2 sets of rules for two different purposes.

- 1. `./yara` - rules that get applied to the command line with all parameters, e.g. `WMIC.exe delete justatest`
- 2. `./yara/in-memory` - rules that get applied to process memory of the parent process of our intercepted process, e.g. ransomware.exe running our intercepted process vssadmin.exe

YARA External Variables

Since version 1.1 we pass a list of external variables into the YARA matching process to allow for much more complex and clever YARA rules that take attributes of the process and its parent into account.

Variable	Description	Example Value
FromRaccine		true
Name	Image file name	WMIC.exe
ExecutablePath	Full path to binary	C:\Windows\System32\wbem\WMIC.exe
CommandLine	Full command line with parameters	WMIC.exe delete justatest
Priority	Process priority	32
ParentName	Parent image file name	cmd.exe
ParentExecutablePath	Full path to parent executable	C:\Windows\System32\cmd.exe
ParentCommandLine	Full parent command line with parameters	C:\WINDOWS\system32\cmd.exe
ParentPriority	Parent process priority	32

The matching process looks like this on the command line:

```
"C:\Program Files\Raccine\yara64.exe" -d FromRaccine="true" -d Name=
```

The following listing shows an example YARA rule that makes use of the external variables in its condition.

```
rule env_vars_test {
    condition:
        Name contains "WMIC.exe"
        and CommandLine contains "delete justatest"
        and ParentPriority >= 8
        and (
            ParentCommandLine contains "cmd"
            or ParentCommandLine contains "powershell"
```

```
}
)
```

Deploy Configuration via GPO

The folder `GPO` includes `Raccine.ADMX` and `Raccine.ADML` . In deployment the `Raccine.ADMX` file goes in `C:\Windows\PolicyDefinitions` . The accompanying `Raccine.ADML` files goes in `C:\Windows\PolicyDefinitions\en-US` .

To use: Open `GPEDIT.MSC` > Computer Configuration > Administrative Templates > System > Raccine

After configuring the changes, you may need to bump gpo by running `gpupdate.exe` .

Logfile

A logfile with all interceptions and actions taken is written to

`C:\ProgramData\Raccine\Raccine_log.txt`

```
C: > ProgramData > Raccine_log.txt
1 2020-10-16 12:13:41 DETECTED_CMD: 'vssadmin delete shadows ' PID: 11776 ACTION: Whitelisted
2 2020-10-16 12:13:41 DETECTED_CMD: 'vssadmin delete shadows ' PID: 9552 ACTION: Terminated
3 2020-10-16 12:13:41 DETECTED_CMD: 'vssadmin delete shadows ' PID: 14640 ACTION: Terminated
4 2020-10-16 12:16:10 DETECTED_CMD: 'wmic delete shadowcopy ' PID: 11776 ACTION: Whitelisted
5 2020-10-16 12:16:10 DETECTED_CMD: 'wmic delete shadowcopy ' PID: 9552 ACTION: Terminated
6 2020-10-16 12:16:10 DETECTED_CMD: 'wmic delete shadowcopy ' PID: 17816 ACTION: Terminated
7 2020-10-16 12:18:34 DETECTED_CMD: 'powershell -ENCOD JABbaTheHuttandmoreChars ' PID: 11776 ACTION: Whitelisted
8 2020-10-16 12:18:34 DETECTED_CMD: 'powershell -ENCOD JABbaTheHuttandmoreChars ' PID: 9552 ACTION: Terminated
9 2020-10-16 12:18:34 DETECTED_CMD: 'powershell -ENCOD JABbaTheHuttandmoreChars ' PID: 9452 ACTION: Terminated
10
```

Windows Eventlog

An entry is generated by every blocking event in the `Application` eventlog.

Application Number of events: 11.344

Level	Date and Time	Source	Event ID	Task Category
Information	03/11/2020 17:50:49	Raccine	2	None
Information	03/11/2020 17:50:44	Raccine	2	None
Information	03/11/2020 17:50:44	Raccine	1	None
Information	03/11/2020 17:46:23	Raccine	2	None

Event 2, Raccine

General Details

Raccine detected malicious activity:
vssadmin delete shadows
(simulation mode)

Raccine Context:
ChildName= "vssadmin.exe"
ChildExecutablePath= "C:\Windows\System32\vssadmin.exe"
ChildCommandLine= "vssadmin delete shadows"
ChildTimeSinceExeCreation=332
ChildPid=9820
ParentName= "cmd.exe"
ParentExecutablePath= "C:\Windows\System32\cmd.exe"
ParentCommandLine= "C:\WINDOWS\system32\cmd.exe"
ParentTimeSinceExeCreation=20
ParentPid=10096
GrandParentName= "explorer.exe"
GrandParentExecutablePath= "C:\Windows\explorer.exe"
GrandParentCommandLine= "C:\WINDOWS\Explorer.EXE"
GrandParentTimeSinceExeCreation=20
GrandParentPid=14436

The IDs that Raccine generates








- EventId 1 - Setup activity
- EventId 2 - Malicious activity detected
- EventId 3 - Benign activity detected

Simulation Mode

Since version 0.10.0, Raccine can be installed in "simulation mode", which activates all triggers, logs all actions but doesn't kill anything. This mode should be used in environments in which backup solutions or other legitimate software for a reasonable amount of time to check if Raccine would interfere with other software. The idea is to install Raccine in simulation mode, let it log for a week or month and then check the logs to see if it would have blocked legitimate software used in the organisation.

Application

Number of events: 6.961

Level	Date and Time	Source	Event ID	Task Category
 Information	18/10/2020 10:31:53	Raccine	2	None
 Information	18/10/2020 10:31:16	Raccine	2	None
 Information	18/10/2020 10:29:40	Raccine	2	None
 Information	18/10/2020 10:29:40	Raccine	1	None
 Information	18/10/2020 10:01:28	Security-SPP	16384	None
 Information	18/10/2020 10:00:48	Security-SPP	16394	None
 Information	18/10/2020 09:58:46	edgeupdate	0	None

Event 2, Raccine

General

Details

Raccine detected malicious activity:

powershell -e JABbaTheHuttandMoreBase64code

(simulation mode)

Screenshot

Run `raccine.exe` and watch the parent process tree die (screenshot of v0.1)

Command Prompt - cmd.exe /c powershell.exe - raccine.exe

Microsoft Windows [Version 10.0.19041.508]

(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\venom>cd Documents\GitHub\Raccine

C:\Users\venom\Documents\GitHub\Raccine>cmd.exe /c powershell.exe

Windows PowerShell

Copyright (c) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\venom\Documents\GitHub\Raccine> cmd.exe

Microsoft Windows [Version 10.0.19041.508]

(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\venom\Documents\GitHub\Raccine>raccine.exe

Raccine PID is 5700

Collecting PID 6768 for a kill

Collecting PID 17832 for a kill

Collecting PID 5820 for a kill

Collecting PID 14840 for a kill

Process with PID 9404 is on whitelist

Collecting PID 20808 for a kill

Kill PID 20808

Kill PID 14840

Kill PID 5820

Kill PID 17832

Kill PID 6768

Process Explorer - Sysinternals: www.sysinternals.com [TRIDENTX\venom] (Admin)

Process	CPU	Private Bytes	Working Set	PID
chrome.exe	< 0.01	42.160 K	68.268 K	14012
chrome.exe	< 0.01	53.472 K	79.896 K	3740
chrome.exe	< 0.01	55.344 K	81.992 K	12152
chrome.exe	< 0.01	55.092 K	78.444 K	7748
chrome.exe	< 0.01	54.900 K	78.296 K	12028
chrome.exe	< 0.01	43.192 K	64.796 K	4108
chrome.exe	< 0.01	36.088 K	58.380 K	18256
chrome.exe	0.01	91.064 K	137.012 K	16516
chrome.exe	< 0.01	6.252 K	15.208 K	12940
chrome.exe	< 0.01	42.448 K	75.296 K	11596
chrome.exe	< 0.01	48.676 K	82.252 K	17892
chrome.exe	< 0.01	39.076 K	70.664 K	7624
chrome.exe	< 0.01	48.224 K	81.200 K	9256
chrome.exe	< 0.01	29.420 K	58.692 K	5116
chrome.exe	< 0.01	57.528 K	92.608 K	18504
chrome.exe	< 0.01	31.308 K	65.288 K	17324
chrome.exe	< 0.01	11.996 K	22.224 K	5412
cmd.exe	< 0.01	4.004 K	4.584 K	14840
conhost.exe	< 0.01	8.228 K	21.220 K	19304
cmd.exe	< 0.01	2.212 K	4.392 K	5820
powershell.exe	< 0.01	58.040 K	65.748 K	17832
cmd.exe	< 0.01	3.488 K	4.736 K	6768
raccine.exe	< 0.01	556 K	3.728 K	5700

CPU Usage: 1.97%

Commit Charge: 60.94%

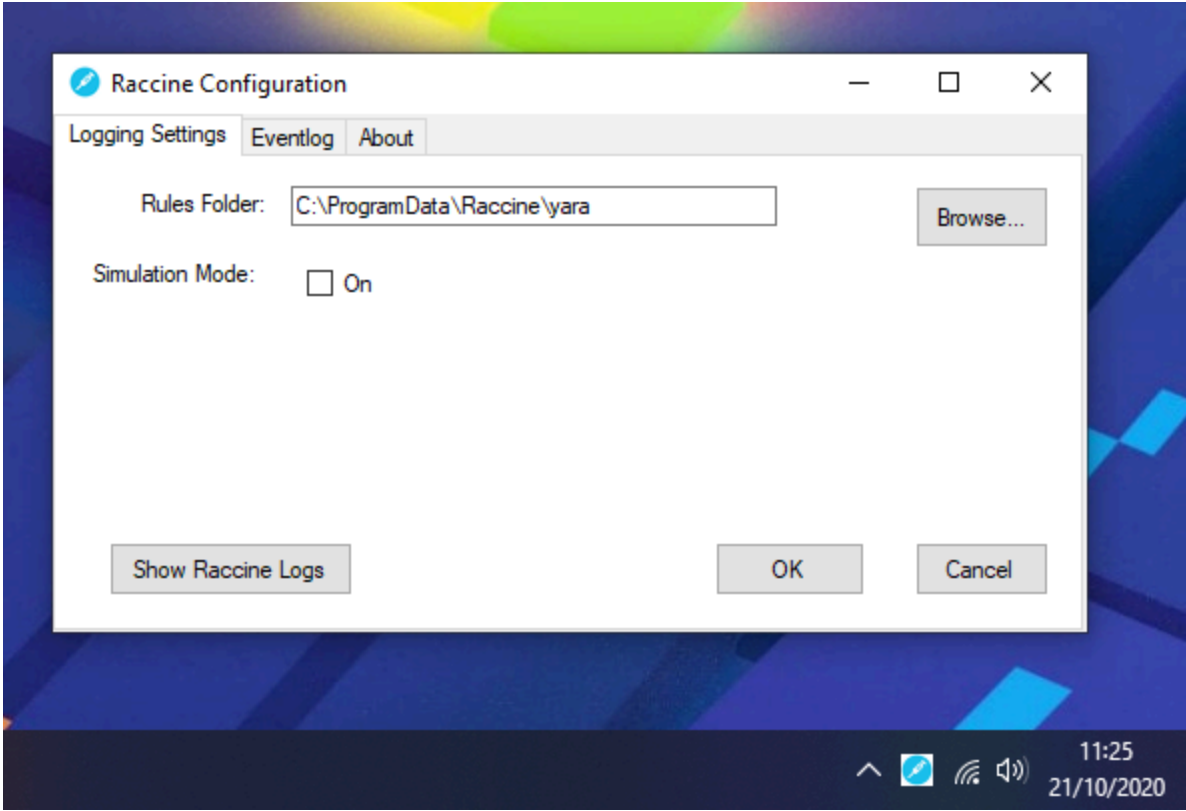
Processes: 259

Physical Usage: 48.79%

GUI

Available and required since version 1.

Page 8 of 9



Pivot

In case that the Ransomware that your're currently handling uses a certain process name, e.g. `taskd1.exe` , you could just change the `.reg` patch to intercept calls to that name and let Raccine kill all parent processes of the invoking process tree.

Help Wanted

I'd like to extend Raccine but lack the C++ coding skills, especially on the Windows platform.

Help - My System is Broken

If anything happens to your installation, e.g. sudden error messages, broken services or programs that won't start anymore, run the file `raccine-reg-patch-uninstall.reg` in the `reg-patches` sub folder. This should bring everything back to normal.

After that your should also be able to run a full uninstallation using `install-raccine.bat` .

Other Info

