Cryptowall Ransomware - Analysis

John Tolomay

Filename: cryptowall.bin

Static Analysis

PeStudio

flag (30)	label (139)	group (17)	value (2121)
x	-	windowing	GetWindowText
x	import	sharing	CloseClipboard
x	import	sharing	GetClipboardData
x	import	sharing	EnumClipboardFormats
x	import	sharing	OpenClipboard
x	import	sharing	SetClipboardData
x	import	sharing	EmptyClipboard
x	import	sharing	GlobalDeleteAtom
x	import	security	OpenProcessToken
x	import	security	AdjustTokenPrivileges
x	-	security	LookupPrivilegeValue
x	-	registry	RegDeleteKey
x	-	registry	RegSetValue
x	-	registry	RegCreateKey
x	import	reconnaissance	GetCurrentProcessId
x	import	memory	VirtualAlloc
x	import	file	WriteFile
x	import	execution	GetThreadTimes
x	import	execution	OpenProcess
x	import	execution	GetCurrentThreadId
x	import	execution	TerminateProcess
x	import	execution	GetCurrentProcess
x	import	execution	GetEnvironmentStrings
x	import	execution	GetEnvironmentStrings
x	import	diagnostic	ImageRvaToVa
x	import	diagnostic	ImageNtHeader
x	import	diagnostic	ImageRvaToSection
x	-	desktop	GetProcessWindowStation
x	-	desktop	GetUserObjectInformation
x	import	-	GlobalCompact
-	import	windowing	GetActiveWindow

List of strings used by the malware that were flagged as potentially malicious. For example, this contains an API call, AdjustTokenPrivileges, which is used to enable or disable privileges in a specified process token. The malware is likely using this to gain additional permissions. RegSetValue and RegCreateKey are also seen in the list, which creates and updates values in the registry. It's not shown in the screenshot, but GetTickCount is also used to see if it is being run in a virtualized environment. Other API calls that the malware is using to see if it is in a virtual environment are GetStartupInfoA and GetSystemMetrics. The malware might be calling GetStartupInfoA to see if there are any strange settings for process startups that indicate a virtual environment. Similarly, GetSystemMetrics is likely called to see if any system metrics or configuration settings are unique to virtual machines. This includes screen dimensions/information and checking if the session is remote. In order for analysts to create a legitimate seeming virtual environment, these settings should be as accurate as can be to

normal machines. If not, the malware is more likely to detect that and not execute its malicious code entirely.

library (6)	flag (1)	description
USER32.dll	-	Multi-User Windows USER API Client Library
COMDLG32.dll	-	Common Dialogs Library
ADVAPI32.dll	-	Advanced Windows 32 Base API
dbghelp.dll	x	Windows Image Helper
COMCTL32.dll	-	Common Controls Library
KERNEL32.dll	-	Windows NT BASE API Client

List of libraries used by the malware. Dbghelp.dll is the only lowercase dll and is flagged as malicious by PeStudio. This could be a fake dll the malware created or uses a legitimate dll to gain more information about the environment it is running in.

Capa

```
PS C:\Users\Administrator\Desktop> capa -r C:\capa-rules-5.1.0 -s C:\1_flare_msvc_rtf_32_64.sig .\cryptowall.bin
matching: 100%|                    | 197/197 [00:10<00:00, 19.08 functions/s, skipped 158 library functions (80%)]
+-------------------------+-------------------------------------------------------------------------------------+
| md5                     | 47363b94cee907e2b8926c1be61150c7                                                    |
| sha1                    | ca963033b9a285b8cd0044df38146a932c838071                                            |
| sha256                  | 45317968759d3e37282ceb75149f627d648534c5b4685f6da3966d8f6fca662d                    |
| os                      | windows                                                                             |
| format                  | pe                                                                                  |
| arch                    | i386                                                                                |
| path                    | cryptowall.bin                                                                      |
+-------------------------+-------------------------------------------------------------------------------------+

+-------------------------+-------------------------------------------------------------------------------------+
| ATT&CK Tactic           | ATT&CK Technique                                                                    |
|-------------------------+-------------------------------------------------------------------------------------|
| DEFENSE EVASION         | Virtualization/Sandbox Evasion::System Checks T1497.001                             |
| DISCOVERY               | System Information Discovery T1082                                                  |
| EXECUTION               | Command and Scripting Interpreter T1059                                             |
|                         | Shared Modules T1129                                                                |
+-------------------------+-------------------------------------------------------------------------------------+

+-------------------------+-------------------------------------------------------------------------------------+
| MBC Objective           | MBC Behavior                                                                        |
|-------------------------+-------------------------------------------------------------------------------------|
| ANTI-BEHAVIORAL ANALYSIS | Virtual Machine Detection [B0009]                                                  |
| DISCOVERY               | System Information Discovery [E1082]                                                |
| EXECUTION               | Command and Scripting Interpreter [E1059]                                           |
| PROCESS                 | Allocate Thread Local Storage [C0040]                                               |
|                         | Terminate Process [C0018]                                                           |
+-------------------------+-------------------------------------------------------------------------------------+

+-------------------------------------------------+-------------------------------------------------------+
| CAPABILITY                                      | NAMESPACE                                             |
|-------------------------------------------------+-------------------------------------------------------|
| reference anti-VM strings targeting VMWare      | anti-analysis/anti-vm/vm-detection                    |
| contain a resource (.rsrc) section              | executable/pe/section/rsrc                            |
| extract resource via kernel32 functions         | executable/resource                                   |
| accept command line arguments                   | host-interaction/cli                                  |
| check OS version                                | host-interaction/os/version                           |
| allocate thread local storage                   | host-interaction/process                              |
| terminate process                               | host-interaction/process/terminate                    |
| link many functions at runtime                  | linking/runtime-linking                               |
+-------------------------------------------------+-------------------------------------------------------+
```

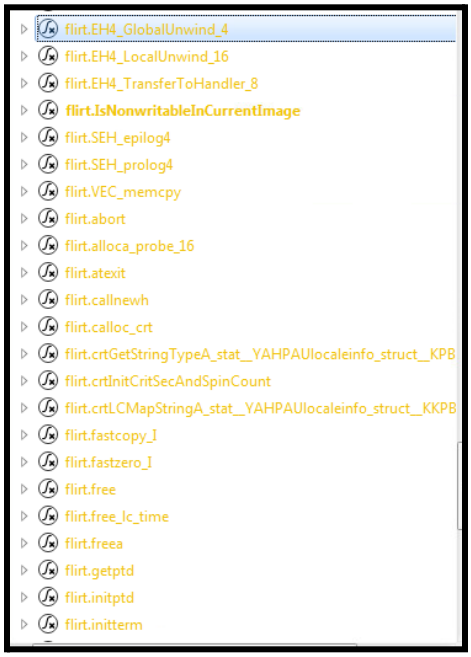
The capabilities of the malware include virtual machine detection, discovering information about the host system, controlling thread allocation, and terminating running processes. Given that the sample is ransomware, this makes sense.

```
contain a resource (.rsrc) section
namespace  executable/pe/section/rsrc
author     moritz.raabe@mandiant.com
scope      file
section: .rsrc @ 0x37D0000

extract resource via kernel32 functions
namespace  executable/resource
author     william.ballenthin@mandiant.com
scope      function
function @ 0x401100
  or:
    and:
      or:
        api: kernel32.LockResource @ 0x4026C0

accept command line arguments
namespace  host-interaction/cli
author     moritz.raabe@mandiant.com, anushka.virgaonkar@mandiant.com
scope      function
att&ck     Execution::Command and Scripting Interpreter [T1059]
mbc        Execution::Command and Scripting Interpreter [E1059]
function @ 0x401100
  or:
    api: GetCommandLine @ 0x402504

check OS version
namespace  host-interaction/os/version
author     michael.hunhoff@mandiant.com, johnk3r
scope      function
att&ck     Discovery::System Information Discovery [T1082]
mbc        Discovery::System Information Discovery [E1082]
function @ 0x401100
  and:
    match: get OS version @ 0x401100
      or:
        api: GetVersion @ 0x402454
    or:
      and:
        instruction:
          and:
            mnemonic: cmp @ 0x4027B5
```

The -vv flag for capa returns how it believes each capability is being executed. For example, one capability of the malware is to accept command line arguments. This is done through the GetCommandLine function as listed above. This is useful because we can see what tools are being used by the malware to achieve its goal.

Cutter

In Cutter, we see the malicious code calls GetCommandLine, which we suspect is accepting command line arguments and parsing them. Throughout the assembly code, the malware often calls these “flirt” functions. After some research, I found that FLIRT stands for “Fast Library Identification and Recognition Technology” and is a database that contains signature patterns for identifying known functions and libraries. Malware often uses a technique called FLIRT signature evasion to hide malicious code in functions from these known libraries. This could be one way the malware tries to hide itself from being spotted and analyzed.

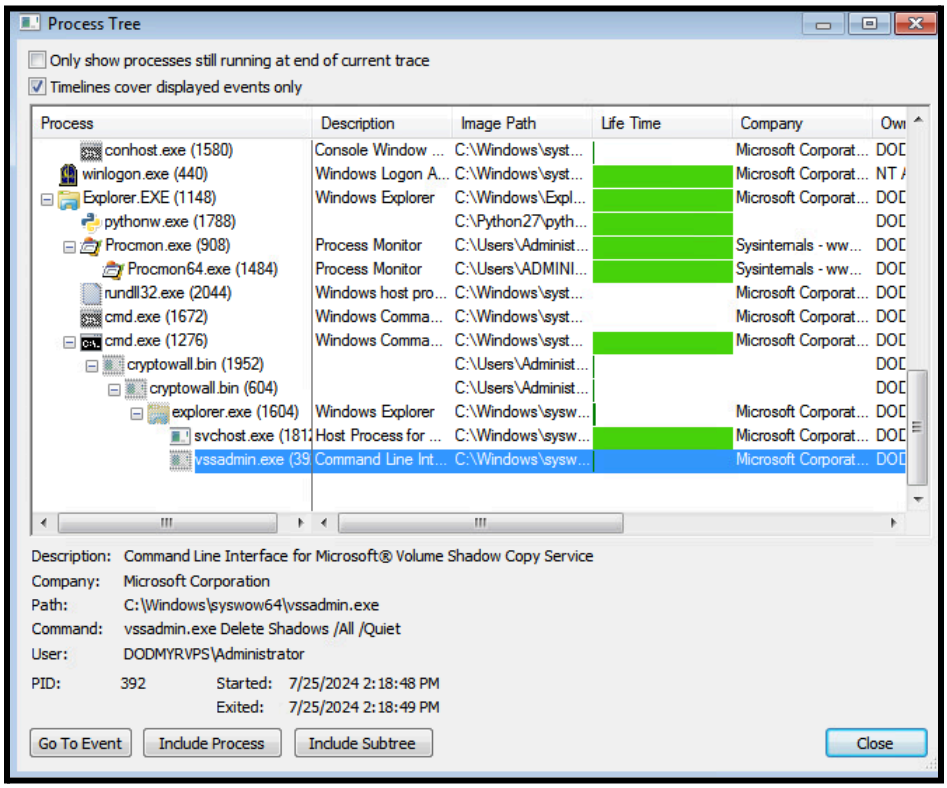


Looking these functions up on Google gave me no results, leading me to believe these are malicious functions the malware created that are disguised as legitimate.

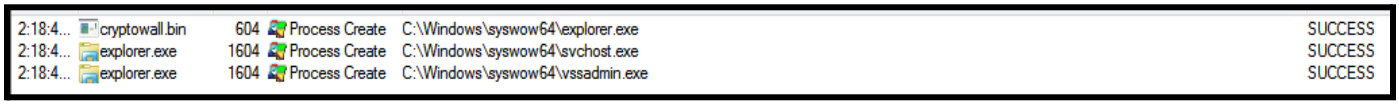
Dynamic Analysis

Onced executed, the malware immediately deletes itself (cryptowall.bin).

Process Monitor



In process monitor, we can see that the child processes of cryptowall.bin, svchost.exe and vssadmin.exe were executed as a result of detonating the malware. The malware used the command line to run the command, “vssadmin.exe Delete Shadows /All /Quiet”. This command deletes all shadow copies of the infected machine, making it so there are no stable backups to revert back to. This ensures that the victim must pay the ransom in order to recover their files.



The malware creates these processes as well as a lot of DLLs in the directory “syswow64”. Svchost.exe (Service Host) is a legitimate process that hosts the services and files that Windows needs to run efficiently. However, it’s possible that the malware is exploiting svchost.exe to run its own malicious code within a legit system process of the service host, making it harder to detect.

cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisableUserModeCallbackFilter	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\CWDIllegalInDLLSearch	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisableUserModeCallbackFilter	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\CWDIllegalInDLLSearch	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions\(Default)	SUCCESS
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\SafeDllSearchMode	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\GRE_Initialize\DisableMetaFiles	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Compatibility32\cryptowall	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Windows\LoadAppInit_DLLs	SUCCESS
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\AuthenticodeEnabled	SUCCESS
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisableLocalOverride	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale\EMPTY	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale\EMPTY	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\Language\InstallLanguageFallback	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\MUI\UILanguages\en-US\Type	SUCCESS
cryptowall.bin	1952	RegQueryValue	HKLM\System\CurrentControlSet\Control\MUI\UILanguages\en-US\AlternateCodePage	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKCU\Control Panel\Desktop\PreferredUILanguages	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKCU\Control Panel\Desktop\MuiCached\MachinePreferredUILanguages	BUFFER OVEF
cryptowall.bin	1952	RegQueryValue	HKCU\Control Panel\Desktop\MuiCached\MachinePreferredUILanguages	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisableUserModeCallbackFilter	NAME NOT FO
cryptowall.bin	1952	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\GRE_Initialize\DisableMetaFiles	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\CWDIllegalInDLLSearch	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisableUserModeCallbackFilter	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\CWDIllegalInDLLSearch	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions\(Default)	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\SafeDllSearchMode	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\GRE_Initialize\DisableMetaFiles	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Compatibility32\cryptowall	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Windows\LoadAppInit_DLLs	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Microsoft\OLE\PageAllocatorUseSystemHeap	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Microsoft\OLE\PageAllocatorSystemHeapIsPrivate	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\System\CurrentControlSet\services\crypt32\DebugHeapFlags	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Internet Settings\DisableImprovedZoneCheck	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\Internet Settings\Security_HKLM_only	NAME NOT FO
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name	SUCCESS

cryptowall.bin	604	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider	SUCCESS
cryptowall.bin	604	RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\Type	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\Image Path	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\Image Path	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\Image Path	SUCCESS
cryptowall.bin	604	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\Image Path	SUCCESS

Lastly, we can see what the malware changes in the registry. For example, “DisableImprovedZoneCheck” is a registry setting related to Windows security that once disabled, will restrict security warning boxes from appearing on the screen. Again, this is a way the malware is trying to go undetected. The malware attempts to do similar things with “DisableUserModeCallbackFilter” and “DisableMetaFiles” that work to make the machine even more vulnerable. The malware also uses “Microsoft Strong Cryptographic Provider” to likely encrypt its own files to avoid detection and analysis.

Conclusion

In this analysis, we learned how the Cryptowall Ransomware sample carries out its attack and why it is ransomware in the first place. According to capa, this malware has the ability to accept command line arguments, detect virtual environments, learn information about the host machine, and more. In PeStudio, there were various API calls that were used to detect a virtualized environment and explained how it is important to make sure the system settings are accurate to that of a legitimate machine to avoid detection. We saw in Cutter that it likely created malicious functions disguised as legitimate via FLIRT signatures. Once the malware was detonated, it spawned new processes such as vssadmin.exe, that deleted all stable backups to force the user to pay the ransom to get their files back. Finally, the malware changes many values in the registry to bring down the security of the victim machine, making it easier to carry out the attack. The malware also makes it a priority to avoid detection by encrypting itself and disabling security flags in the registry. In conclusion, this is a very powerful piece of malware because once executed, it changes and deletes so many key components that it makes it extremely hard to fix, leading most victims to simply pay the ransom.