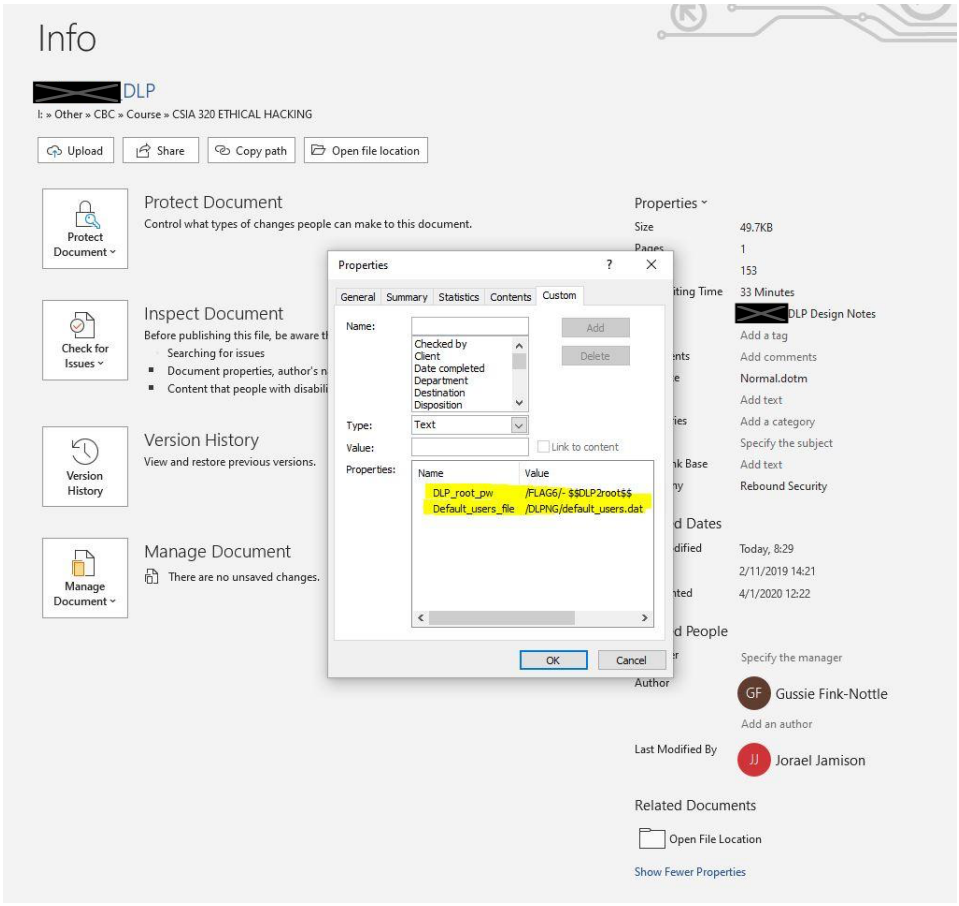
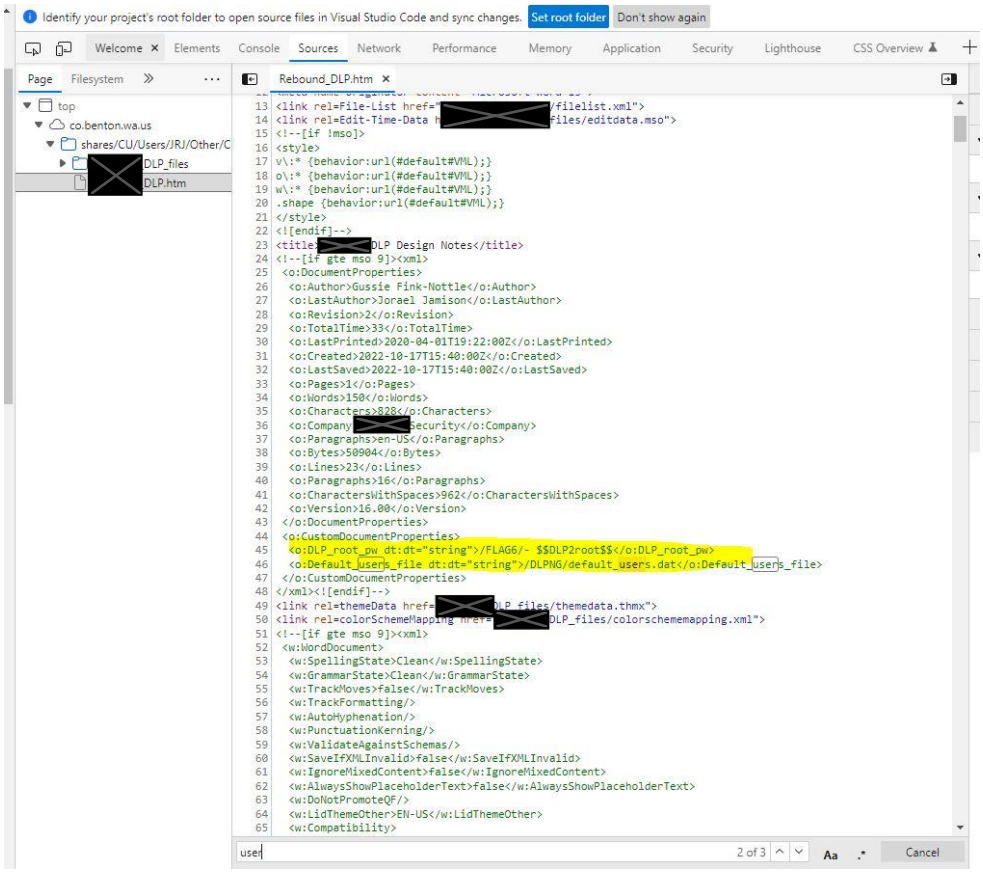
Jorael Jamison
CSIA320
Professor Eric Robinson
11/20/2022

Exam – Capture the Flags 2

**/FLAG6/** -

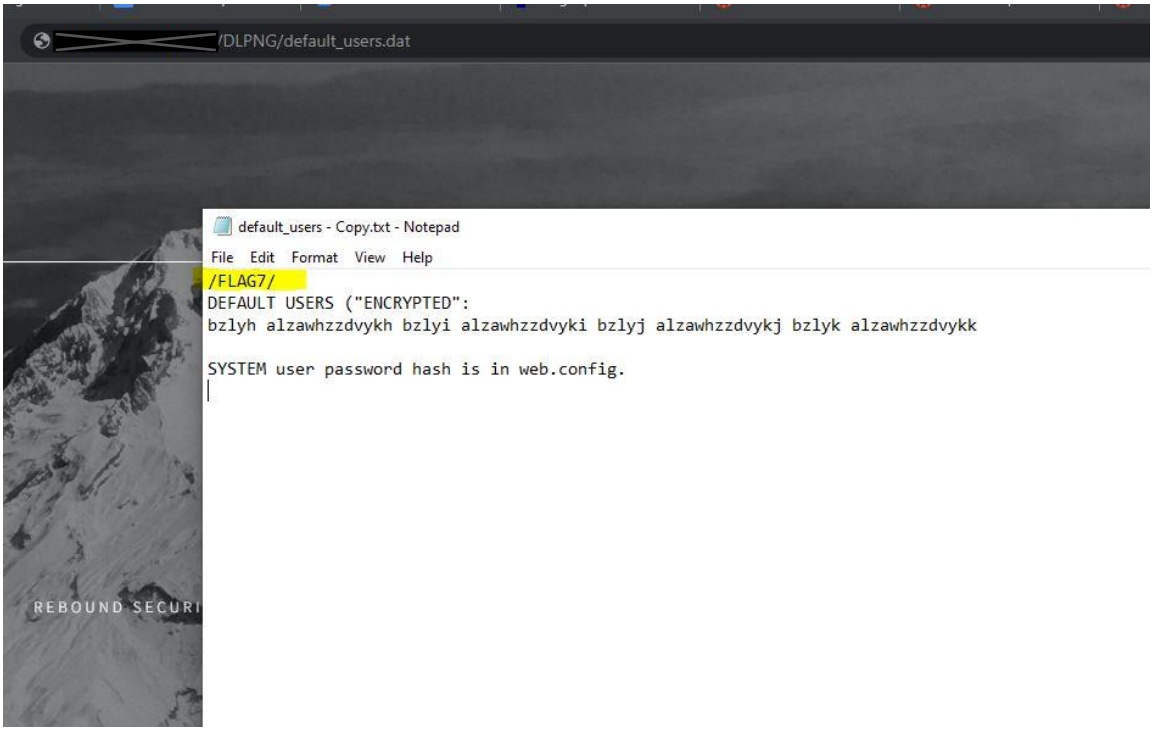
1. On the [redacted] website I clicked on DLP and selected "[redacted] DLP Design Notes" which downloaded a [redacted] DLP.docx file. I opened the Word document and viewed its advanced properties which displayed the DLP_root_pw and Default_users_file information. I saved the file as HTML and viewed in the browser using devtools to see the source code data. I found the root password **"DLP2root"** displayed with **/FLAG6/.**






**/FLAG7/** -

2. Within the source code right under the root password for Flag 6 I found the path to a file /DLPNG/default_users.dat and trailed that at the end of the URL for [redacted]security as shown below. This downloaded a .dat file and I opened within a .txt file to view contents. It displayed **/FLAG7/** with the encrypted default user's hash. It also mentioned the SYSTEM user password is in the web.config file.




bzlyh alzawhzzdvykh bzlyi alzawhzzdvyki bzlyj alzawhzzdvykj bzlyk alzawhzzdvykk – This was discovered to be a shift cipher. I cracked it using a +7 shift:

**usera testpassworda userb testpasswordb userc testpasswordc userd testpasswordd**

**/FLAG8/** -

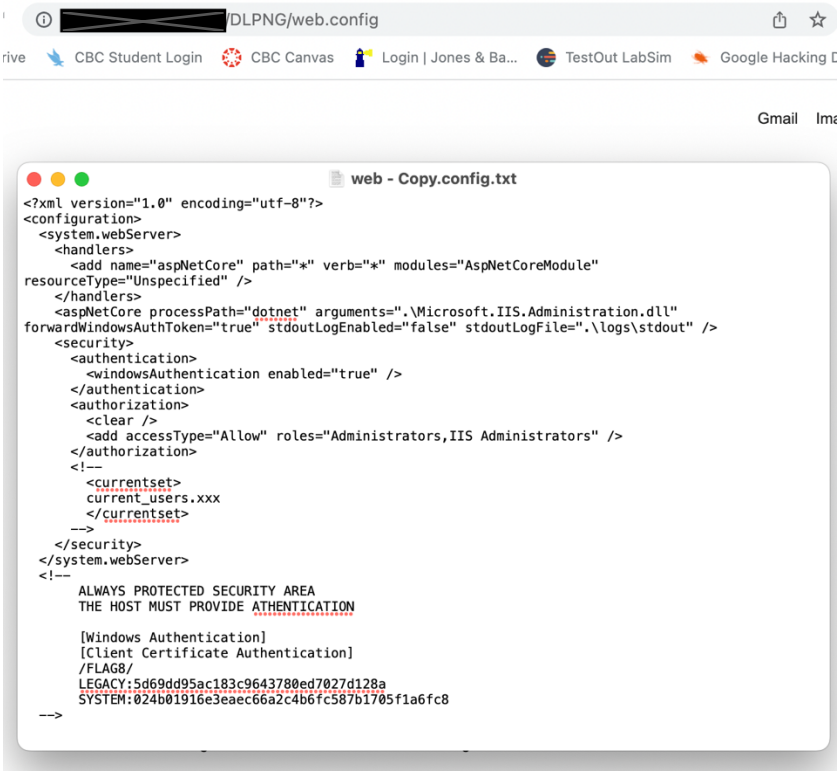
3. As shown in the default_users file above, I took the web.config and added that to the end of the [redacted] URL as well which downloaded the file. I opened in as a .txt file and displayed the contents showing the SYSTEM user's password hash as **/FLAG8/.**

   I took each hash below and ran them on Linux Hash ID tool to find the type of hash:

   LEGACY:5d69dd95ac183c9643780ed7027d128a **MD5 HASH**

   SYSTEM:024b01916e3eaec66a2c4b6fc587b1705f1a6fc8 **SHA1 HASH**

   I ran each using Hashcat on Linux and found the passwords for each as **password9**.




The SHA1 hash was stored inside the SecretFile.txt

```
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 100 (SHA1)
Hash.Target......: 024b01916e3eaec66a2c4b6fc587b1705f1a6fc8
Time.Started.....: Sun Nov  6 21:18:09 2022 (0 secs)
Time.Estimated...: Sun Nov  6 21:18:09 2022 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:  2413.1 kH/s (0.05ms) @ Accel:256 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests
Progress.........: 5632/14344385 (0.04%)
Rejected.........: 0/5632 (0.00%)
Restore.Point....: 5120/14344385 (0.04%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: allison1 → katana

Started: Sun Nov  6 21:18:08 2022
Stopped: Sun Nov  6 21:18:11 2022
```

```
┌──(parallels㉿kali-linux-2022-2)-[~]
└─$ sudo cat crack.txt
024b01916e3eaec66a2c4b6fc587b1705f1a6fc8:password9
```

```
┌──(parallels㉿kali-linux-2022-2)-[~]
└─$ sudo hashcat -m 0 -o cracked "5d69dd95ac183c9643780ed7027d128a" /usr/share/wordlists/rockyou.txt
hashcat (v6.2.5) starting

OpenCL API (OpenCL 3.0 PoCL 3.0+debian  Linux, None+Asserts, RELOC, LLVM 13.0.1, SLEEF, POCL_DEBUG) - Platform #
1 [The pocl project]
=============================================================================
* Device #1: pthread-0×000, 1439/2943 MB (512 MB allocatable), 2MCU
```

Next, for MD5 rather than saving hash in a file I tried it directly on the command line.

```
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 0 (MD5)
Hash.Target......: 5d69dd95ac183c9643780ed7027d128a
Time.Started.....: Sun Nov  6 21:30:50 2022 (0 secs)
Time.Estimated...: Sun Nov  6 21:30:50 2022 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:  2884.8 kH/s (0.07ms) @ Accel:256 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests
Progress.........: 5632/14344385 (0.04%)
Rejected.........: 0/5632 (0.00%)
Restore.Point....: 5120/14344385 (0.04%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: allison1 → katana

Started: Sun Nov  6 21:30:43 2022
Stopped: Sun Nov  6 21:30:52 2022
```

```
┌──(parallels㉿kali-linux-2022-2)-[~]
└─$ sudo cat cracked
5d69dd95ac183c9643780ed7027d128a:password9
```

## /FLAG9/ - NOT SOLVED

4. **Find the current set of default users and their passwords. This data will be marked with "/FLAG9/". Show all work for full credit.**

I remember you mentioning about steganography in the Top2 (inception) photo and the narnia.jpg images, I focused on that for a while assuming the flag would be embedded in a file within one of the images on the site, but I spent a long time playing around with each trying to extract any files or analyzing the metadata using various tools for hashes but cannot seem to crack this one.

In the Top2 Image I ran strings command to find: `$3br`
`%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz.`

I brought down the site with wget -r and combed through each file, photo, and source code on the site using tools like Hashcat and John password crackers trying to brute or dictionary crack any hashes within the files, also used exiftools to extract metadata and used steghide to try and extract any secret files within images but could not locate any valid hashes to crack. Not sure what I am missing here but have put a lot of thought into trying to solve this flag.