Cybersecurity

Brute Force Offline Lab

Contributed by Dr. David Raymond, Virginia Tech University

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY CISA

CYBER.ORG
THE ACADEMIC INITIATIVE OF THE CYBER INNOVATION CENTER

Brute Force Materials

- Materials needed
  - Kali Linux Virtual Machine

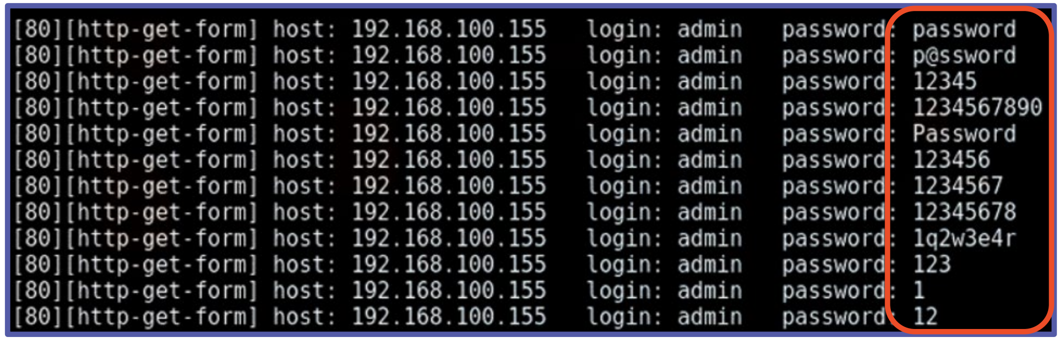
- Software Tool used
  - JTR (John the Ripper)
    - Password cracking tool (pre-installed on Kali OS)

Objectives Covered

- Security+ Objectives (SY0-701)
    - Objective 2.4 - Given a scenario, analyze indicators of malicious activity.
        - Password Attacks
            - Brute force

What is a Brute Force Attack?

- A brute force attack is a form of password attack where the attack attempts to guess a password by trying many passwords in the attempt to guess the correct password

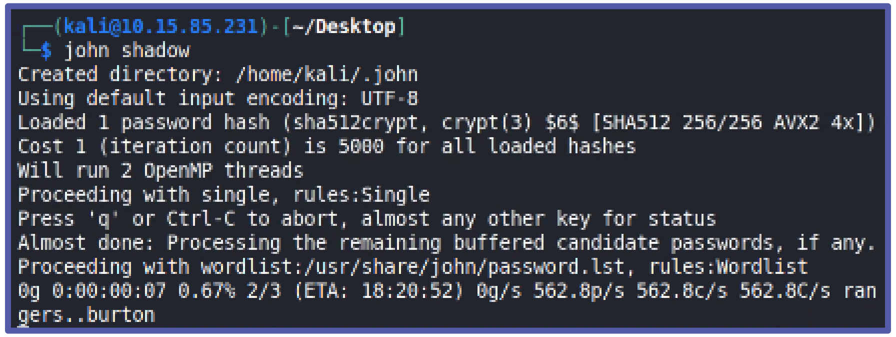


```
[80][http-get-form] host: 192.168.100.155   login: admin   password: password
[80][http-get-form] host: 192.168.100.155   login: admin   password: p@ssword
[80][http-get-form] host: 192.168.100.155   login: admin   password: 12345
[80][http-get-form] host: 192.168.100.155   login: admin   password: 1234567890
[80][http-get-form] host: 192.168.100.155   login: admin   password: Password
[80][http-get-form] host: 192.168.100.155   login: admin   password: 123456
[80][http-get-form] host: 192.168.100.155   login: admin   password: 1234567
[80][http-get-form] host: 192.168.100.155   login: admin   password: 12345678
[80][http-get-form] host: 192.168.100.155   login: admin   password: 1q2w3e4r
[80][http-get-form] host: 192.168.100.155   login: admin   password: 123
[80][http-get-form] host: 192.168.100.155   login: admin   password: 1
[80][http-get-form] host: 192.168.100.155   login: admin   password: 12
```

Notice all the passwords being used in hopes of finding the right password for the system

Brute Force Lab Overview

1. Set up Environment
2. Create example users
3. Set example passwords
4. Locate password file
5. Change Permissions
6. Launch the Attack
7. More Hashes
8. Observe results



```
┌──(kali@10.15.85.231)-[~/Desktop]
└─$ john shadow
Created directory: /home/kali/.john
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
0g 0:00:00:07 0.67% 2/3 (ETA: 18:20:52) 0g/s 562.8p/s 562.8c/s 562.8C/s ran
gers..burton
```

Set up Environment

- Log into your range
- Open the Kali Linux Environment
  - You should be on your Kali Linux Desktop

Create Users

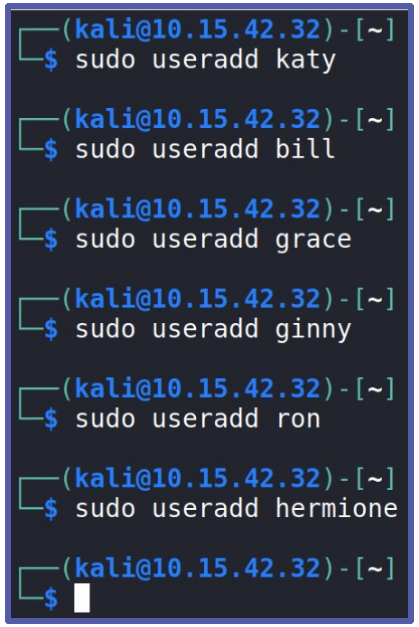
- In your Kali VM open a terminal by clicking on the terminal icon at the top left corner
- Create a user on the system:

  ```
  sudo useradd katy
  ```

  - This command creates a user named “katy”
- Create additional users by using the following command:

  ```
  sudo useradd bill
  ```

- Create at least 3 users
- Remember the users' names - you will need these to set passwords for them

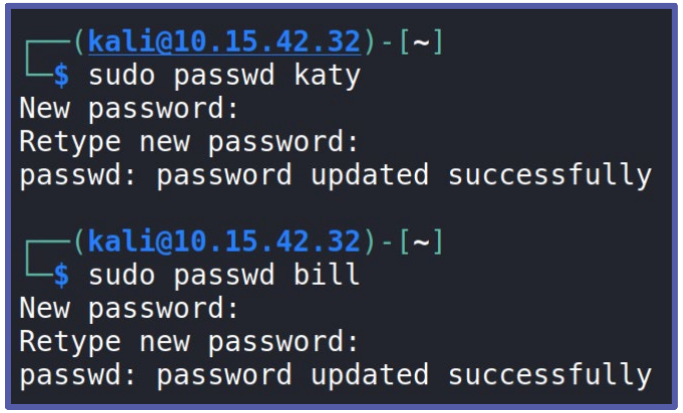


```
┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd katy

┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd bill

┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd grace

┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd ginny

┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd ron

┌──(kali@10.15.42.32)-[~]
└─$ sudo useradd hermione

┌──(kali@10.15.42.32)-[~]
└─$ 
```

Set Passwords

- Use the following command to set a password for each account:
  - The following command starts the prompt to set a password for the user katy

    `sudo passwd katy`
- Enter the password at the prompt "New password:"
  - Set the password to be one from the list of the names you added to the dictionary file earlier!
- Repeat this step for all user accounts you created.

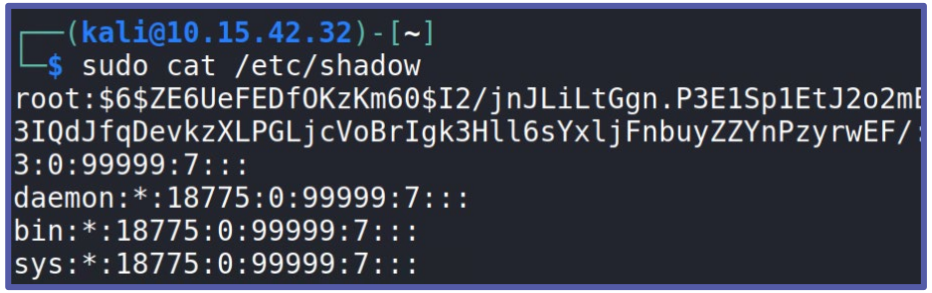


```
┌──(kali@10.15.42.32)-[~]
└─$ sudo passwd katy
New password:
Retype new password:
passwd: password updated successfully

┌──(kali@10.15.42.32)-[~]
└─$ sudo passwd bill
New password:
Retype new password:
passwd: password updated successfully
```

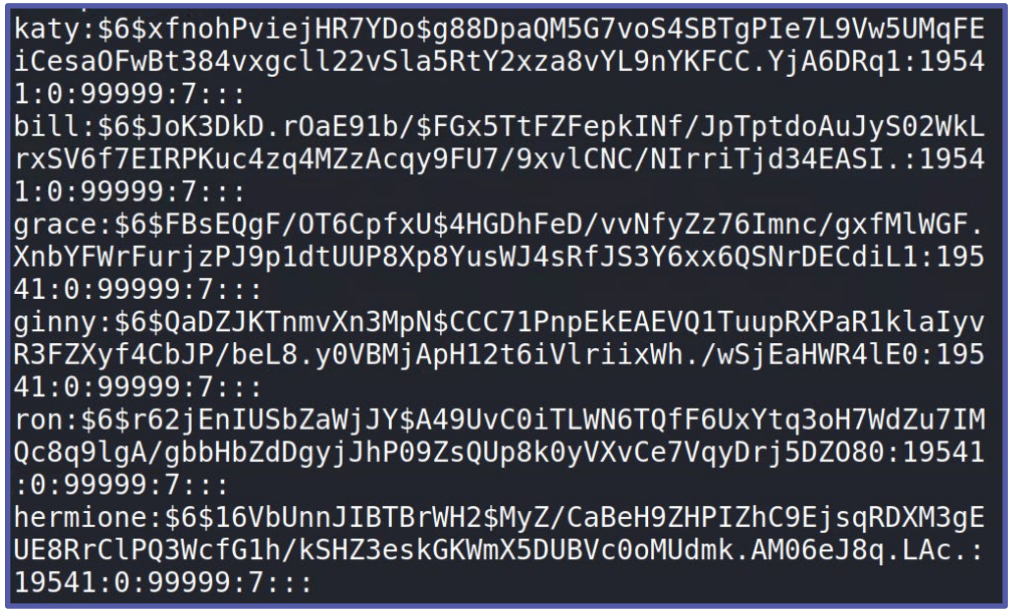
Locate Hashed Passwords

- Display the hashed passwords:

  `sudo cat /etc/shadow`



```
┌──(kali@10.15.42.32)-[~]
└─$ sudo cat /etc/shadow
root:$6$ZE6UeFEDfOKzKm60$I2/jnJLiLtGgn.P3E1Sp1EtJ2o2mE
3IQdJfqDevkzXLPGLjcVoBrIgk3Hll6sYxljFnbuyZZYnPzyrwEF/
3:0:99999:7:::
daemon:*:18775:0:99999:7:::
bin:*:18775:0:99999:7:::
sys:*:18775:0:99999:7:::
```

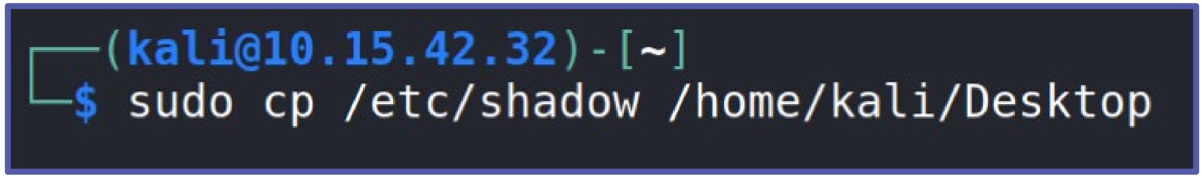


```
katy:$6$xfnohPviejHR7YDo$g88DpaQM5G7voS4SBTgPIe7L9Vw5UMqFE
iCesaOFwBt384vxgcll22vSla5RtY2xza8vYL9nYKFCC.YjA6DRq1:1954
1:0:99999:7:::
bill:$6$JoK3DkD.rOaE91b/$FGx5TtFZFepkINf/JpTptdoAuJyS02WkL
rxSV6f7EIRPKuc4zq4MZzAcqy9FU7/9xvlCNC/NIrriTjd34EASI.:1954
1:0:99999:7:::
grace:$6$FBsEQgF/OT6CpfxU$4HGDhFeD/vvNfyZz76Imnc/gxfMlWGF.
XnbYFWrFurjzPJ9p1dtUUP8Xp8YusWJ4sRfJS3Y6xx6QSNrDECdiL1:195
41:0:99999:7:::
ginny:$6$QaDZJKTnmvXn3MpN$CCC71PnpEkEAEVQ1TuupRXPaR1klaIyv
R3FZXyf4CbJP/beL8.y0VBMjApH12t6iVlriixWh./wSjEaHWR4lE0:195
41:0:99999:7:::
ron:$6$r62jEnIUSbZaWjJY$A49UvC0iTLWN6TQfF6UxYtq3oH7WdZu7IM
Qc8q9lgA/gbbHbZdDgyjJhP09ZsQUp8k0yVXvCe7VqyDrj5DZO80:19541
:0:99999:7:::
hermione:$6$16VbUnnJIBTBrWH2$MyZ/CaBeH9ZHPIZhC9EjsqRDXM3gE
UE8RrClPQ3WcfG1h/kSHZ3eskGKWmX5DUBVc0oMUdmk.AM06eJ8q.LAc.:
19541:0:99999:7:::
```

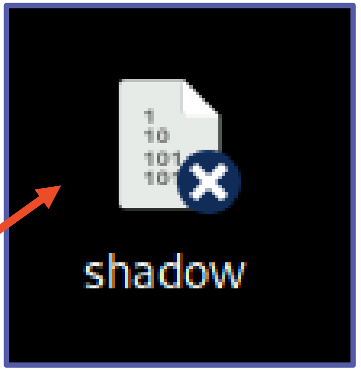
- Passwords are stored in the **`shadow`** file located in the **`/etc`** directory

Move Hashed Passwords

- Copy the **`shadow`** file to your Desktop using the following command:
`sudo cp /etc/shadow /home/kali/Desktop`



```
┌──(kali@10.15.42.32)-[~]
└─$ sudo cp /etc/shadow /home/kali/Desktop
```



Verify the shadow document appears on the Desktop

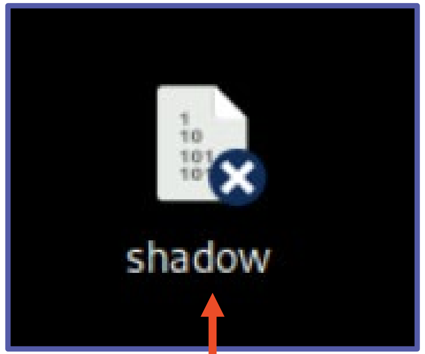
Change Permissions

- Navigate to the Desktop

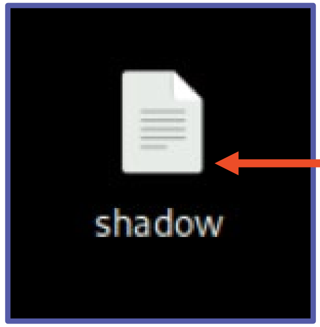
```
cd Desktop
```

- Change the permissions on the shadow file

```
sudo chmod 777 shadow
```



Verify the shadow document appears on the Desktop



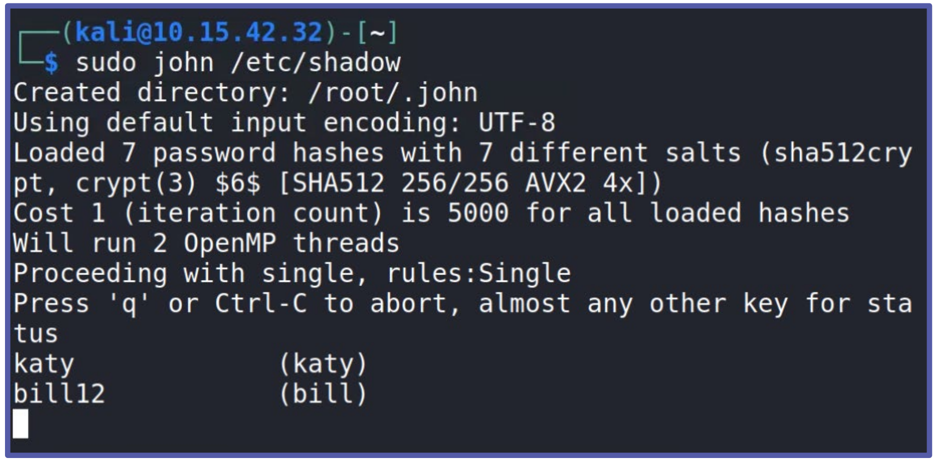
Verify the blue icon is removed from the shadow document

Launching the JTR Attack

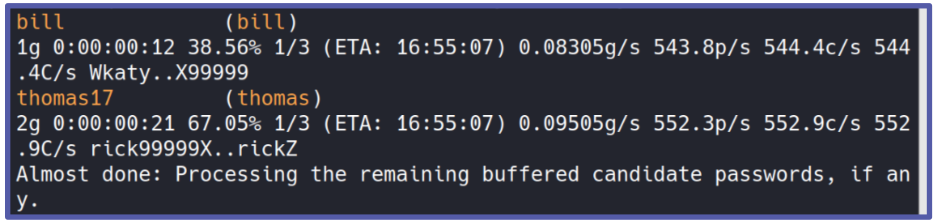
- In order to launch the attack, use the following command:

  `john shadow`

- This will run *John the Ripper* on the **`shadow`** file and start working to crack the passwords
- Press **space** while the attack is working to see what passwords *John the Ripper* is currently trying
- Note this will take some time, depending on the strength of the passwords



```
┌──(kali@10.15.42.32)-[~]
└─$ sudo john /etc/shadow
Created directory: /root/.john
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (sha512cry
pt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for sta
tus
katy             (katy)
bill12           (bill)
█
```

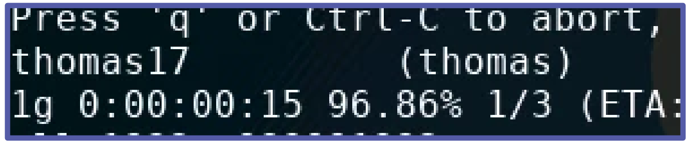


```
bill             (bill)
1g 0:00:00:12 38.56% 1/3 (ETA: 16:55:07) 0.08305g/s 543.8p/s 544.4c/s 544
.4C/s Wkaty..X99999
thomas17         (thomas)
2g 0:00:00:21 67.05% 1/3 (ETA: 16:55:07) 0.09505g/s 552.3p/s 552.9c/s 552
.9C/s rick99999X..rickZ
Almost done: Processing the remaining buffered candidate passwords, if an
y.
```

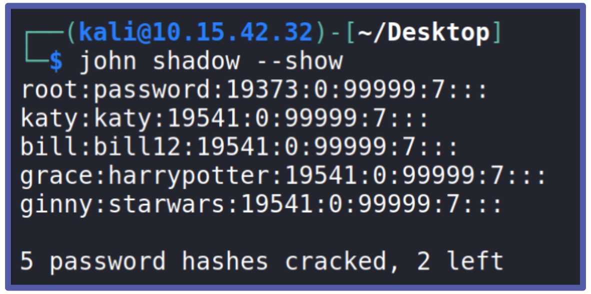
Seeing the Results

- Notice that a found password will display the result while JTR is running
  - The following example found “**`thomas17`**” to be the password for the user “**`thomas`**”
  - Not a very secure password was it?
- You can wait for JTR to finish or press **CTRL+C** to stop the attack.
- The following command will show all the passwords that have been solved

  `john shadow --show`



```
Press 'q' or Ctrl-C to abort,
thomas17          (thomas)
1g 0:00:00:15 96.86% 1/3 (ETA:
```



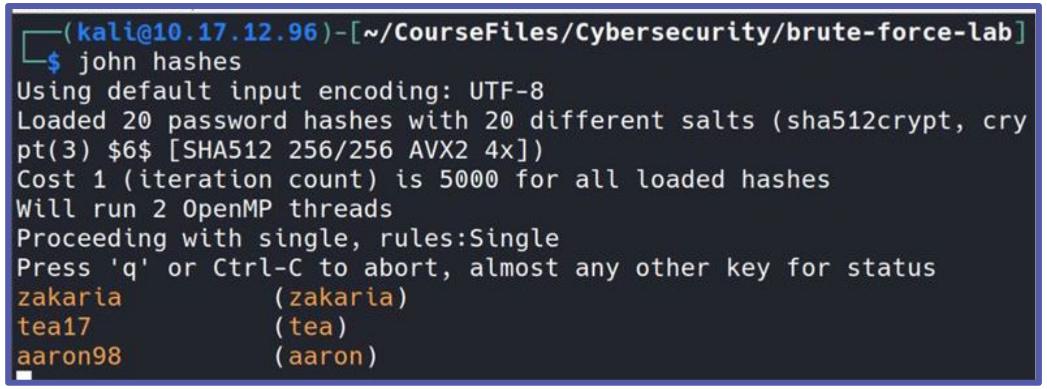
```
┌──(kali@10.15.42.32)-[~/Desktop]
└─$ john shadow --show
root:password:19373:0:99999:7:::
katy:katy:19541:0:99999:7:::
bill:bill12:19541:0:99999:7:::
grace:harrypotter:19541:0:99999:7:::
ginny:starwars:19541:0:99999:7:::

5 password hashes cracked, 2 left
```

More Hashes

- Open a new Terminal and navigate to the lab folder

```
cd /home/kali/CourseFiles/Cybersecurity/brute-force-lab
```

- Display the hashes
  - `cat hashes`
    - Notice there are 20 password hashes
- Crack the hashes
  - `john hashes`



```
┌──(kali@10.17.12.96)-[~/CourseFiles/Cybersecurity/brute-force-lab]
└─$ john hashes
Using default input encoding: UTF-8
Loaded 20 password hashes with 20 different salts (sha512crypt, cry
pt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
zakaria          (zakaria)
tea17            (tea)
aaron98          (aaron)
```

How to Defend Against a Brute Force Attack?

- Strong Passwords
  - Why is a longer password stronger? (D0e5 w31rd sp3LLing M4tt3r?)
  - Why were some passwords solved before others?
- Increasingly longer delay between failed attempts
  - Slow down the attacker. (10s, 15s, 30s, 45s, 1minute between attempts.)
- Lockout after __ failed attempts
  - Account will eventually lock. User will need contact support to regain access.
- Two-Factor Authentication
  - Why would these help secure your password?
- What are some other ways of defending against a brute force attack?