Password cracking

Vulnerabilità e Difesa dei Sistemi Internet

Caliandro Pierciro

University of Rome Tor Vergata

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password - overview

Password - overview

- Passwords are one of the most employed method for authentication
- There are several problems related with passwords, most of them have one common source: human factor!
- Common passwords choices
  - 25% similar to TOP-20
  - 16% first name
  - 4% "password" variant
- Common passwords length
  - 26% length of 6 byte
  - 19% length of 7 byte
  - 20% length of 8 byte

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password - low entropy

- **Entropy** can be seen as a way to measure how much a given quantity is ***random***
- Password chosen by users usually have low entropy
  - They are not generated considering all the possible characters
  - Often are related to known words or personal information
  - etc...
- Examples (using KeepassXC):
  - Example of a low entropy password: **ciaomamma123** (entropy $\approx$ 27.52 bit)
  - Example of a high entropy password: **<';Uu?}rk?xx>E sN:/W** (entropy $\approx$ 123.82 bit)

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password - reuse

- People tend to reuse **the same password** for several websites
- If someone somehow gains possess of the password, a **password spray** attack may be employed
- Test the same password on different application and see how many of them guarantee access

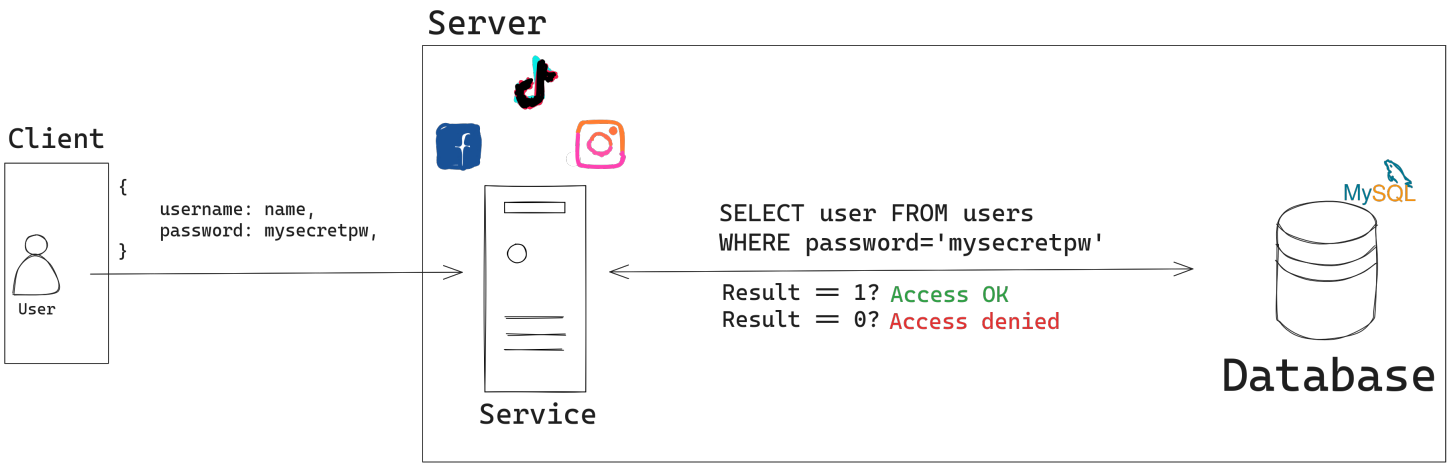
TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password usage

- A **password** may be seen as a shared information between two parties
  - Someone offering a service
  - ourselves
- Password need to be saved
  1. **where?**
  2. **how?**

Password - authentication

1. User sends the information to the service server
2. The server checks if there exists an entry in the database corresponding to the user
3. If so, user authenticates correctly



Password - authentication

- Services **shall NEVER** save user credentials in cleartext
- What happens if there is a data breach and someone gains possess of the exfiltrated data?
- Impersonification of a legitimate user by means of stolen credentials!

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

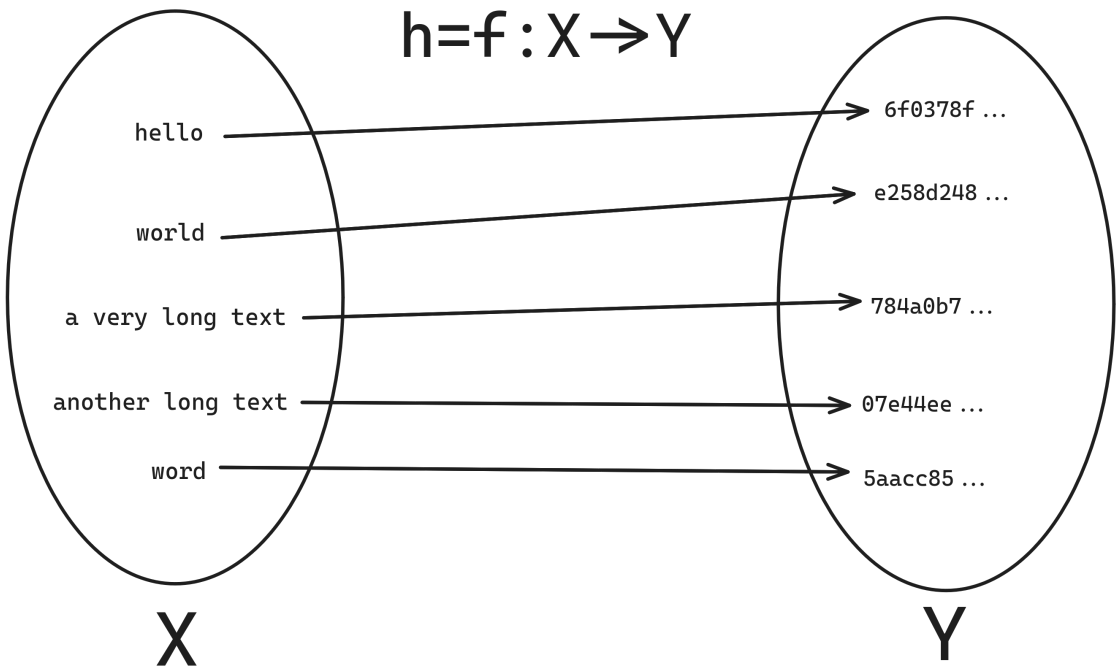
Password - hash functions

- Password is stored using **hash functions**
- Now, if an attacker gains access to the data, it will not be possible to authenicate as the user

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password - cryptographic hash functions

- Cryptographic hash function are surjective functions with some interesting properties such as:
  1. given an hash value, it's **impossible** to find the corresponding input string
  2. it's **impossible** to find two input strings which **collide** (i.e. have the same hash output)

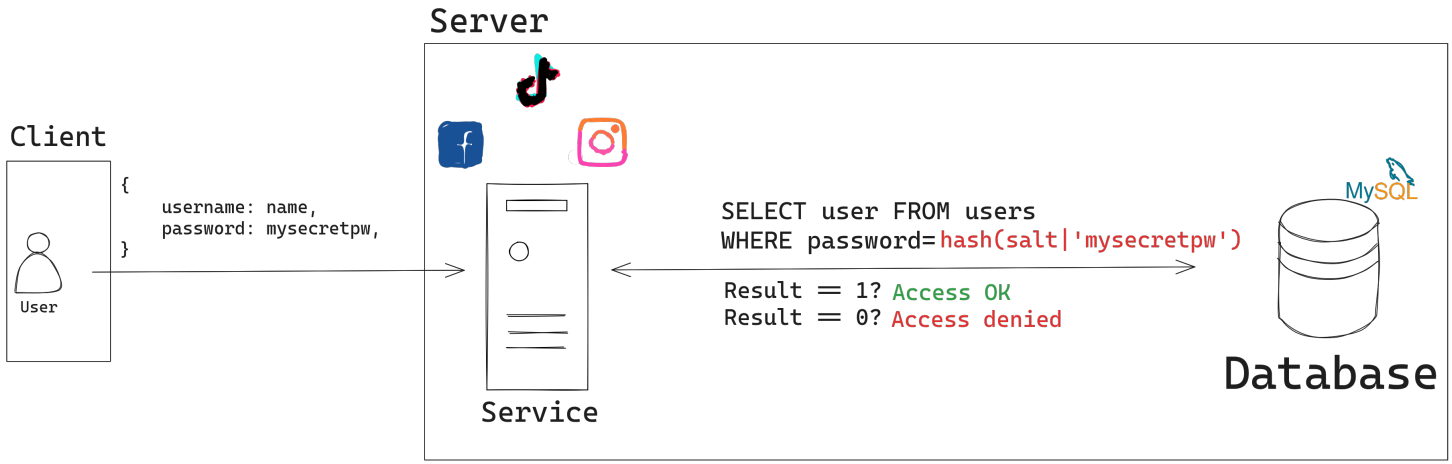


Password - cryptographic hash functions

- There are several standardized hash function
  1. `md5`: was widely used, now is deprecated since it's not considered secure anymore
     - See this paper
  2. `sha1`: same as above
  3. **`sha256`**
  4. `sha384`
  5. ...

Password - cryptographic hash functions + salt

- We still have a problem: users that share the same password will have the same resulting hash!
- This is solved by adding **salt**
- Salt is chosen at random and saved in the database



Linux hashes

- Let's have a look at a `/etc/shadow` entry:

```
kali:$y$j9T$ufXTBpN1QpgwlgqRFmb/B0$/.y0ybAF4iNQXniErsDWf9QSl2HZH7LnBeRHB4ZiQa9:...
```

- We can dissect it as: `user : $id $salt $hash : other-stuff : ...`, where
  - $id is the hash function used
  - $salt is the salt value
  - $hash is the product of $H_{id}(\text{salt}|\text{password})$

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Linux hashes

- The `mkpasswd` command allows us to generate an hash for a password
- `mkpasswd -m help`: shows the list of available hashing algorithms
- `mkpasswd -m sha256crypt` generates the sha256 of the password, which is asked as input
- Result: `$5$4pkPTTnWqX8Mh0ub$VX7Pt5NxkJCr.ETdjkt93dyiyIw9NeEMOX51z0RCBS9`
- `mkpasswd -m sha256crypt -S provasal` generates the sha256 of the password with salt
- Result: `$5$provasal$Teu.Q6SD0/wqLoTw3kqPKv1jMjPywAoxALAAOrb97S.`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking

Password cracking - bruteforcing

There are several methods one can use to crack a password

- **Online attack**: a user tries to authenticate to a service by trying several possible password
- Pros:
  - Requires no knowledge on the password
- Cons:
  - Generates a lot of requests
  - Easy to design countemeasures

- **Offline attack**: a user succesfully obtained a list of hashes
- Pros:
  - No need to interact with the service
  - Can make use of more sophisticated tools
  - If a password gets cracked $\rightarrow$ can try a password spray attack !
- Cons:
  - Needs to know some information about the password

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - bruteforcing

- How much time do we need to crack a password that is:
  - 8 caracters long
  - composed by lower case letters (tot $= 26$) and numbers (tot $= 10$)
- For each character, the probability to guess its value is

$$\frac{1}{36}$$

- Hence, the probability to guess the entire password is

$$\frac{1}{36^8} = \frac{1}{2.821.109.907.456}$$

- Mean time to crack the password (using a CPU which does $66\frac{\text{M}}{\text{s}}$ guesses)

$$\frac{2.8 \cdot 10^{12}}{6.6 \cdot 10^7} \cdot \frac{1}{2} \approx 5.9h$$

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - bruteforcing

- How much time do we need to crack a password that is:
  - 8 caracters long
  - composed by any possible character value (i.e. 256)
- For each character, the probability to guess its value is

$$\frac{1}{256}$$

- Hence, the probability to guess the entire password is

$$\frac{1}{256^8} = \frac{1}{1.8 \cdot 10^{19}}$$

- Mean time to crack the password (using a CPU which does $66\frac{\text{M}}{\text{s}}$ guesses)

$$\frac{1.8 \cdot 10^{19}}{6.6 \cdot 10^7} \cdot \frac{1}{2} \approx 4431 \textit{years}!$$

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - tools

Password cracking - online tools

- There are several online tools that can be used for password cracking and hash generation
- Most of the are based on **hash databases** found online after some data breach
- Some examples:
  - MD5 Hash Generator
  - Crack Station
  - Cyber chef

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper

- Command line utility for password cracking
- Can crack several hash functions, using different modes of operation
- To install on Debian derivatives: `sudo apt-get install john`
- You can browse documentation here

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Hashcat

- Command line utility for password recovery
- Can leverage specialized hardware, such as GPUs or FPGAs
- Docs here
- To install on Debian derivatives: `sudo apt-get install hashcat`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - methods for cracking

- **Wordlist**
  - Check collisions using a dictionary as input
  - Attack succeed if dictionary contain password
- **Single-Crack**
  - Will try to crack the password using the login info
- **Incremental**
  - Brute-Force attacks (try almost all combinations)
- **External**
  - User can define it's own mode and rules
- Each mode uses rules defined in: `john.conf`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Dictionaries

- When dealing with target enumeration, password cracking etc..., **wordlists** are a fundamental tools
- One of the best and most up to date source is SecLists
- It's a collection of wordlists that can be used for a wide range of scopes
  - Web enumeration (you will use them a lot when solving HTB/THM machines!)
  - DNS enumeration
  - **Password craking**
  - Login bruteforcing
  - etc...
- It is FOSS (we ❤️ foss!)
- On kali, install with `sudo apt-get install seclists`
- You will find it under `/usr/share/seclists`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Dictonaries

- Another gem from the web: **rockyou.txt**
- *"In December 2009 the social networking site 'RockYou' experienced a data breach, resulting in the exposure of over 32 million user accounts"* (source article)
- You can find the `tar` archive by default on Kali Linux
- How to install it:
  - `gunzip /usr/share/wordlists/rockyou.txt.gz`
  - `sudo apt install wordlists`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - wordlist mode

- Run with `john -w=<dictionary> <file>`
- John will try to hash each password in the wordlist and match the hashes in file
- If a match is found, it will eventually be reported
- Cracked hashes will be found under `$HOME/.john/john.pot`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - wordlist mode

- What if a user tries to strengthen the password by
  - character substitution
  - character append/prepend
  - etc...
- The result will be something like this: `password` → `P@55w0rd1234!`
- The latter seems a password with more entropy, right 😁?

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - wordlist mode

- With `john`, we can specify a **rules** file
- Rules are expressed line by line
- The syntax seems very similar to **regexp**
- Rules documentation can be found here

Password cracking - lab

John the Ripper - lab

- Let's try to crack some hashes, in different formats
- We will use two dictionaries for this task:
  - `/usr/share/seclists/Passwords/Cracked-Hashes/milw0rm-dictionary.txt`
  - `rockyou.txt`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - crack md5

- First, let's try to recognize the hash format: `john --show=formats <file>`
- After we get which the hash format is, we can procede to crack it
- `john --format=md5crypt --fork=$(nproc ) --wordlist=/usr/share/wordlists/rockyou.txt md5_hashes`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

John the Ripper - crack sha256

- We can also try to crack sha256 hashes
- `john --format=Raw-SHA256 --fork=$(nproc ) --wordlist=/usr/share/seclists/Passwords/Cracked-Hashes/milw0rm-dictionary.txt --rules:All sha256_hashes`

Password cracking - online

- We can leverage DVWA login bruteforce service to test an online password crack
- The tools we will use: Burp + Hydra
- `Burp` is a tool widely used for web VAPT (you will see more later on during this course!)
- `Hydra` is a tool for online login bruteforcing

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - online

- We can attempt to bruteforce login
- We will use two wordlists, one for the username and one for the password
  - `/usr/share/seclists/Usernames/top-usernames-shortlist.txt`
  - `/usr/share/wordlists/rockyou.txt`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - online

- First, we need to understand what HTTP method is used and what parameters
- We can leverage burp to do so

Request

Pretty Raw Hex

```
1  POST /dvwa/login.php HTTP/1.1
2  Host: 192.168.122.153
3  User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5  Accept-Language: en-US,en;q=0.5
6  Accept-Encoding: gzip, deflate, br
7  Content-Type: application/x-www-form-urlencoded
8  Content-Length: 41
9  Origin: http://192.168.122.153
10 Connection: keep-alive
11 Referer: http://192.168.122.153/dvwa/login.php
12 Cookie: security=low; PHPSESSID=0da925126bb64f2a126495947d616bb9
13 Upgrade-Insecure-Requests: 1
14 Priority: u=0, i
15
16 username=prova&password=prova&Login=Login
```

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Password cracking - online

- We then build our Hydra command
- `hydra -L /usr/share/seclists/Usernames/top-usernames-shortlist.txt -P /usr/share/wordlists/rockyou.txt 192.168.122.153 http-post-form "/dvwa/login.php:user=^USER^&password=^PASS^&Login=Login:Login failed" -V`

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

Take home message

- Password are still crucial out there, choose them wisely!
- When possible, try to use a password manager (such as KeepassXC
- Also, use built-in functions from password managers to generate high entropy passwords for the most crucial apps (such as home banking etc...)

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

References

- Breaking MD5
- Hydra blog post

Coffee break!

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA