Tutelr Infinity Internships - Assignment Week 3

Team: Penetration Testing and Red Teaming

# SYNGSONG

Syngsong is a tool that generates password guesses based on song lyrics using the Genius API. It uses the lyrics of a song to create a wordlist and then applies Hashcat-style mask rules to create password guesses. This tool is useful for generating passwords that are easy to remember but still meet the complexity requirements of a strong password.

**Some of the key features of Syngsong include:**

1. Customizable Masking: Syngsong allows users to specify a masking rule to generate passwords that meet specific complexity requirements. Users can specify the length of the password, the types of characters to include, and the position of those characters in the password.

2. Multiple Song Selection: Syngsong allows users to select from a large database of songs on the Genius platform. This means that users can generate password guesses based on songs they are familiar with, making them easier to remember.

3. Password Complexity Requirements: Syngsong allows users to set requirements for the complexity of the generated passwords. Users can specify a minimum length, minimum number of uppercase and lowercase letters, numbers, and special characters.

Syngsong can be used to generate password guesses that meet password complexity requirements by selecting a song that the user is familiar with and then applying a masking rule to generate a password that meets the specified requirements. The tool can be used by security professionals to test the strength of passwords used by employees and clients of a company, or by individuals who want to create strong, memorable passwords.

Installation process and workthrough :-

1. Clone the Syngsong repository from Github: https://github.com/DeBos99/syngsong
2. Install the required Python modules by running the command

   “pip install -r requirements.txt"
3. Obtain an API key from the Genius API website: https://genius.com/api-clients
4. Set our Genius API access token as an environment variable called "GENIUS_ACCESS_TOKEN".

**Usage:**

1. Open a terminal window and navigate to the Syngsong directory.
2. Run the command "python **main**.py artist_name" where artist_name is the name of the artist we want to generate password guesses from.
3. We can also use the following optional arguments:

- --songtitle: the title of the song to use (default: the most popular song by the artist).
- --minsize: the minimum length of the password guesses (default: 8).
- --maxsize: the maximum length of the password guesses (default: 16).
- --mask: the Hashcat-style mask to use for the password guesses (default: ?l?u?d).
- --topsongs: the number of top songs to select from when choosing a song by the artist (default: 10).
- --debug: enables debug mode.

4. Syngsong will generate a list of password guesses based on the artist and song lyrics. By default, it will generate 10 password guesses.
5. We can save the password guesses to a file by adding "> filename.txt" to the end of the command. For example: "python **__main__**.py artist_name > passwords.txt"

Cmd Commands to generate the results:--

**Example**

Python __main__.py artist_name --songtitle "song_title" --geniuskey "genius_api_key"

**Run this code**

python __main__.py --artist "Arjit singh" --minsize 12 --mask "?l?u?d?s" --topsongs 50 -g <API_KEY> -o output.txt

Note that <API_KEY> should be replaced with our actual API key for the Genius API, and output.txt is the desired name of the output file.

Artist – Arjit singh

Hashcat mask - ?l?u?d?s

The generated password is saved in google drive and the link is :

https://drive.google.com/file/d/1znHPVK5GP-YGsjXu7GmXVjEtvhxvDA1o/view?usp=share_link

## PASSWORD COMPLEXITY ANALYSIS TOOL

**ZXCVBN**

A realistic password strength estimator.

This is a Python implementation of the library created by the team at Dropbox. The original library, written for JavaScript, can be found here.

While there may be other Python ports available, this one is the most up to date and is recommended by the original developers of zxcvbn at this time.

**Features**

- Tested in Python versions 2.7, 3.6-3.9
- Accepts user data to be added to the dictionaries that are tested against (name, birthdate, etc)
- Gives a score to the password, from 0 (terrible) to 4 (great)
- Provides feedback on the password and ways to improve it
- Returns time estimates on how long it would take to guess the password in different situations

Installation

Install the package using pip: **pip install zxcvbn**

**Usage:**

Pass a password as the first parameter, and a list of user-provided inputs as the user_inputs parameter (optional).

**Code :**

```
from zxcvbn import zxcvbn

results = zxcvbn('JohnSmith123', user_inputs=['John', 'Smith'])

print(results)
```

save this code in the same directory where the zxcvbn is downloaded as a python extension file like

**analyze_passwords.py**

for generating the result, we can use this command to get the password complexity result that we have generate using the **syngsong** tool.

**Python analyze_passwords.py**

Here are some results of the password complexity analysis results

1st Password - **Dil mera dekhoWj9**

**https://drive.google.com/file/d/1AsOVua0Btr0ZIK8iOEPXzvLyb3LAcsIY/view?usp=share_link**

2nd password - **MausamkaigujaareFr8**

**https://drive.google.com/file/d/1TG2g_xpIDFNxX1t-6zyx9AGEWYep3sLk/view?usp=share_link**

3rd password - **SAU SAAL HAII**

**https://drive.google.com/file/d/1aZ7Y3A28KvVabXfum0-lvT7VYZpd8XPb/view?usp=share_link**

4th password - **DIL MERA DEKHOBy7**

**https://drive.google.com/file/d/11bOPzgGia143vIVeollK50CsPnElqVNO/view?usp=share_link**

5th password - **Khairiyat puchhoMd3**

**https://drive.google.com/file/d/1xMA1CpLZGFOOkq_4Oc4MoQ8e9FSbL1Ju/view?usp=share_link**

password recommendations for improving the password policies and practices at the company.

**Examples:**

1. Implement a strong password policy: The password policy should require employees to create complex passwords that include a combination of upper- and lower-case letters, numbers, and special characters. Passwords should also be changed frequently, at least every 90 days, and employees should not use the same password for multiple accounts.

2. Use Multi-Factor Authentication: Multi-Factor Authentication adds an extra layer of security to the sign-in process by requiring users to provide additional authentication factors, such as a fingerprint or a one-time code sent to their phone. in addition to their password. This makes it harder for attackers to access sensitive information.

3. Train Employees in Password Best Practices: Employees should be trained on how to create and manage strong passwords, recognize phishing scams and other social engineering attacks that attempt to steal passwords, and report suspicious activity.

4. Regularly monitor and audit passwords: Regularly monitor and audit passwords to identify any weak or compromised passwords, and enforce password changes as needed.

5. Use a password manager: Encourage employees to use a password manager to securely store and generate strong, unique passwords for each account. This can help prevent employees from reusing passwords or using weak passwords.

# Assessment

1. What is Syngsong?

Ans -> B. A tool to generate password guesses based on song lyrics via the Genius API.

2. How does Syngsong generate password guesses?

Ans -> C. By using the Genius API to extract lyrics and then applying Hashcat style masking.

3. What is Hashcat style masking?

Ans -> D. A technique used to crack passwords

4. Can Syngsong generate passphrases that meet password complexity requirements?

Ans -> A. Yes

5. Which API does Syngsong use to extract song lyrics?

Ans -> C. Genius API

6. How can Syngsong be useful for password cracking?

Ans -> A. By generating a list of potential passwords based on song lyrics

7. What are some password complexity requirements that Syngsong can generate?

Ans -> D. All of the above(Length requirements, Character set requirements, Combination of uppercase and lowercase letters, numbers and symbols)

8. Is Syngsong a legal tool?

Ans -> A. Yes

9. How can Syngsong be used ethically?

Ans -> A. To test the strength of one's own passwords

Ans -> B. To test the strength of passwords for others with their consent

10. What are some potential risks associated with using Syngsong?

Ans -> D. All of the above (It can generate weak passwords, it can violate the terms of service of the Genius API, and it can be used for illegal activities).