|  |
| --- |
| SET 8D |
| Crack4Beginner Python Application |
| COMP7481 – Carly Orr |

|  |
| --- |
| Danny Lieu  April 7, 2018 |

Table of Contents

[Completed tasks 2](#_Toc510899858)

[To-do lists 2](#_Toc510899859)

[Risks and Bottlenecks 2](#_Toc510899860)

# Completed tasks

The application can crack the encrypted passwords using MD5 encryption. The application uses 8 processes which two processes run on the same processor. The reason that I have to make 8 processes is to reduce the time for one processes to iterate through the list. The word list has 1 million words which makes each process to handle 125,000 words. The app also uses multiprocessing module running in parallel. If one process finds the password, it will notify all other processes to stop to enhance the efficiency.

The time to crack a password which contain only letter is around 30 seconds (both uppercase and lowercase) and the time to crack a password which contain letters and numbers is around 1.5 minutes. If the password is not in the list, the application will exit after 2 minutes.

# To-do lists

The final project should have:

* A GUI for better user experience
* The app will allow the user to choose different kind of encryption: md5, sha256, sha512.

# Risks and Bottlenecks

At first, I tried to encrypt the password using sha256 algorithm. However, the time it took to find a simple password which contained only lowercase letters is around 7 minutes. For the purpose of presenting in class, I prefer md5 to reduce the time. I planned to allow the user to choose which algorithm they want to do.

The application will find how many available processors and create the number of processes which is two times the number of processors. Therefore, the number of processes will differ for different machines depending on how many cores it has.