National University of Singapore

School of Computing

CS2105 **Assignment 3** (3 Marks) Sem 1 AY24/25

Submission Deadline

15th Nov (Friday) 11:59 pm . 1 mark penalty will be imposed on late submission (Late submission refers to submission or re-submission after the deadline). The submission folder will be closed on **22th Nov (Friday) 11:59 pm** .

Objectives

In this assignment, you will implement a program to check for the integrity of a file using the "Message authentication code". After completing this assignment, you should have a good understanding of

- Cryptographic Hash functions and
- verification of message integrity using Message Authentication Codes.

Exercise: Message Integrity

You will be implementing *Message Authentication Code* using SHA256 as the hash function ((Refer to Lecture 11). Given a file, you need to verify if the file content is tampered with. The content of the file is as in Fig 1.

Message Authentication Code (32 Bytes)	Data

Figure 1: File content

Input

The program takes 2 command line inputs

- The file containing the 32 byte "authentication key"
- The data file

Output

If the file contents are not tampered with, print "yes"; otherwise, print "no".

Notice

- The stu server has the package pycryptodomex with version 3.11.0 (link)
  - The package name is "Cryptodome" instead of "Crypto"
  - Some other syntax may be different in 3.11.0 as compared to the most up-to-date version.
  - SHA256: link
- We have provided some template code.
- The outputs are case-sensitive.
- There is a low possibility that the Server gets overloaded. So start the assignment early to avoid "congestion" during the last few days.

Grading Rubric

- Message Integrity: 3 Marks (equally divided among test cases)

Grading

We will test and grade your program on the stu server. Please make sure that your program runs properly on stu. Moreover, you are allowed to use libraries installed in public folders of stu (e.g. /usr/lib) only.

We accept submission of only Python 3 (in particular, 3.10) program. For Python 3, we use the python3 program installed in folder /usr/bin on stu for grading. We will **deduct 1 mark** for every type of failure to follow instructions (e.g. wrong program name).

We will grade your program based on its correctness only. A grading script will be used to test your program and no manual grading will be provided.

Testing Your Program

To test your program, please use your SoC UNIX ID and password to log on to stu as instructed on Assignment 0 paper.

- To use the grading script, please upload your program along with the test folder given in the package to stu. Make sure that your program and the test folder are in the same directory. Then, you can run the following commands to test your server program:

  ```
  bash test/Integrity.sh
  ```

- By default, the script runs through all test cases. You can also choose to run a certain test case by specifying the case number in the command:

```
bash test/Integrity.sh 3
```

To stop a test, press `ctrl-c`. If pressing the key combination once does not work, hold the keys until the script exits.

- If you ever encounter this error: **tput: unknown terminal "xterm-256color"** when testing your program using script provided, run the command:

  `export TERM=xterm` once after you log in and before you run the test scripts.

Program Submission

For individual submission,

- The file name should be `Integrity-<Matric number>.py` where <Matric Number> is your matriculation number which starts with the letter A. An example file name would be Integrity-A0165432X.py.
- Submit it to the `Assignment 3` assignment on Canvas.
- If multiple submissions are needed, submit the file with the name as stated above, and allow Canvas to append numbers to the name.
- All file names are case-sensitive.

You are not allowed to post your solutions to any publicly accessible site on the Internet.

Policy on AI Tools

Caution should be used when using AI tools (eg. ChatGPT). Usage of such tools may be considered plagiarism. Please see NUS' policy on AI for academic works:

https://libguides.nus.edu.sg/new2nus/acadintegrity

In general, the use of AI is acceptable so long as it does not involve code (eg. clarifying concepts taught during lecture, in English or other non-programming languages).

If AI is used for code (generation, or touching up of original code), the relevant sections of code should be marked accordingly with comments, but take note that excessive use of AI for code can be considered plagiarism in the same manner as use of code written by another person, as this assignment is meant to be your own work.

Example comments:

```
#AI Usage: Code generated by ChatGPT using prompt "reading standard input in Python"

(code)

#End section
```

or:

```
#AI Usage: Code touched up by ChatGPT using prompt "reading standard input in Python"
```

```
(code)

#End section
```

Any section of code not marked as such shall be considered a declaration that the code did not use AI for generation or touching up in any way.

Plagiarism Warning

You are free to discuss this assignment with your friends. However, you should refrain from sharing your program, program fragments, or detailed algorithms with others.

We employ zero-tolerance policy against plagiarism. If a suspicious case is found, student would be asked to explain his/her code to the evaluator in face. Confirmed breach may result in zero mark for the assignment and further disciplinary action from the school.

Question & Answer

If you have any doubts on this assignment, please post your questions on `piazza` before consulting the teaching team. However, the teaching team will NOT debug programs for students and we provide support for language-specific questions as a best-effort service. The intention of Q&A is to help clarify misconceptions or give you necessary directions.