School of Computing & Information Technology
CSCI262 Systems Security
SIM-2024-S4

Assignment 1 (14 marks, worth 14%)

Due 11:59pm Singapore time 6th November 2024

Part One: Short answer questions: 4 Marks

This assignment is about **authentication** and **access control**.

1. A phonetic password generator picks two segments randomly for each six-letter password. Each segment has 3 English letters. The form of each segment is Δ Φ Δ (consonant, vowel, consonant), where Φ is an element in $\{a, e, i, o, u\}$ and Δ is an English letter which is not in $\{a, e, i, o, u\}$. Note that Δ for the first and third letters refer to consonants, but they do not have to be the same letters. The two segments could be different. For example, "pampam", "pabbom" and "bacdef" can be possible output of the generator. However, "iamiam" and "bombcd" are not.

   How many possible passwords the generator can generate? 1 Mark

2. Assume that Alice has registered with the server Bob to use Lamport's one-time password scheme. Alice's password is Alice1234567. If n = 5 initially, what are the first 3 one-time password transmitted by Alice? Use MD5 as the hash function. 1 Mark

3. Consider the BLP level relationship diagram in A1-Q3.pdf, and the associated explanation of the notation, and answer the subsequent questions.
   a. Does the diagram define a lattice? Justify your answer. 0.5 Mark
   b. If the diagram does not define a lattice, assume you have corrected it so that it does, while maintaining the relationships between the existing levels. Some of the domination relationships in the diagram are redundant. Identify two such relationships and explain why they are unnecessary. 0.5 Mark

4. Consider the following statements and answer the subsequent questions:

```
Alexis can kick balls and throw sticks.
Boris can catch balls, kick balls, and throw sticks.
Catherine can snap sticks and roll balls.
Duggy the Dog can chew balls, fetch sticks and chew Boris.
```

   a. What are the subjects, objects, and actions for this scenario? 0.25 Mark

   b. Draw an access control matrix representing this scenario. 0.25 Mark

   c. Write a set of access control lists for this situation. 0.25 Mark

   d. Write a set of capability lists for this situation. 0.25 Mark

Part Two: Two factor Authentication 10 Marks

Write two separate programs to simulate two-factor authentication. These programs must not communicate through sockets, files, or any other means. The implementations should be consistent and capable of operating on different computers.

Requirements:

- The programs must be written in either Java, C/C++, or Python.
- Compilation environment instructions will be provided separately.
- Include a Readme.txt file with detailed instructions on how to compile the programs within the specified environment.

The programs are as follows:

1. **Device:** This represents the device which can provide one–time pins that need to be used along with a password.
2. **Connect:** This represents the device which a user intends to connect to, and that the user needs to provide authentication to.

Device should run as:
Device username password
for example, Device Alice 1wdcasFga

Device does not check that the user or password is valid. It starts generating ”one–time” pin values, 6 digit pins. As far as being one–time you do not need to check that they haven’t been used before.

Device: 107283
Device: 837226
Device: 012123
Device: 492833
Device: 217281
...

Each pin should be valid for 15 seconds, although the first pin does not need to be displayed for 15 seconds before the next is displayed. Every 15 seconds, other than for the first pin, a new now-valid pin should appear. So, 15 seconds is how long a user must use that pin in attaching to Connect. The details of pin generation inside Device are up to you. but it needs to be a function of the user and their password in such a way that the Connect can check it is correct during the appropriate time window.

Device should keep running until it is broken by the user, with Ctrl-C for example.

You should include, with your answers to the first part of the assignment, an explanation of the relationship you have chosen to use. You should also explain why you consider the pin values will not leak information about the password despite being a function of the password and username.

Connect should run as:

Connect username new
or
Connect username password pin

The first form is for new users. A new user should be prompted to enter a password twice for confirmation. There should be basic checks to ensure the appropriateness of the password. If the password meets the criteria, it should be stored in a file called Passwords.txt, along with the username. This file should be in plain text, meaning the contents will be human-readable, with the password visible. Although this is not a secure practice in real scenarios, it simplifies the assessment. You can decide the file's format, but it should allow for easy reading and searching of users within the file.

In the second form, the program should verify the username, password, and pin. It should read from the Passwords.txt file to check the consistency of the username and password.

Notes on Submission:

1. Submit your work via Moodle.
2. Include compilation instructions with your submission (i.e., provide a readme.txt file).
3. Late submissions will incur a 25% penalty for each day late, including weekends.
4. Submissions more than three days late will not be marked unless an extension has been granted.
5. If you need an extension, apply through SOLS, preferably before the assignment deadline.
6. Plagiarism is taken seriously. Students found to be involved will likely receive a zero.