CSCU9T4 Practical (27th February) Spring 2017
Security

Download the code for this practical from http://cs.stir.ac.uk/~nve/CSCU9T4/secu ritylab.zip The zip file contains Java source code within a BlueJ project. You may use any other IDE such as Eclipse. Note that a library (jar) for reading CSV files is included in the +*libs* folder which you will have to handle properly if you use something other than BlueJ.

Brute force attacks try all possible combinations of some set of characters to find a password. Dictionary attacks look through a list of passwords. Rainbow table attacks look through a list of precomputed hashes. MD5 is a popular hashing algorithm which is vulnerable to attack[1]. Databases of MD5 hashes can be found online.

1. Open your favourite search engine and look for the text strings that correspond to the following MD5 hashes:
   a. `5F4DCC3B5AA765D61D8327DEB882CF99`
   b. `E10ADC3949BA59ABBE56E057F20F883E`
   c. `B3EA2220280EC43C31C7F6723F85904C`
2. Use the `tryMD5()` method in the provided source code to generate the MD5 hashes for a few other possible passwords and look up the hashes online to check whether they are known.
   Warning. If you wish to try your own passwords to check whether they have been compromised, be aware that the `Scanner` object does not hide what you input so people around you may see what you type in.
   Password input may be hidden using the `readPassword()` method of the `Console` object (see commented code within the `tryMD5()` method), however this will generate a `NullPointerException` if you try to use it within an IDE (BlueJ, Eclipse, etc). To use the code, open the command prompt, navigate to the appropriate folder and call
   `java <NameOfTheClassThatContainsTheMainMethod>`

As a developer, it is important not to unwittingly divulge your secrets (passwords, keys), for example when you host your source code on platforms such as GitHub.

3. Go to https://gitleaks.com/ and look for “password”. This brings up Amazon, Facebook or Google passwords and keys that developers have failed to remove before publicly sharing their code.

[1] http://www.zdnet.com/article/md5-passwordscrambler-no-longer-safe/

Leakeddata.csv contains some fictitious usernames and password hashes (MD5). In its original state the provided source code can be used to crack MD5 hashes but only for strings that only contain lowercase letters.

4. By simply looking at the content of Leakeddata.csv, what is the minimum number of hashes you need to crack to know the passwords for all the usernames?
5. Examine the `bruteForceRecursive` method and understand how it works.
6. Make suitable changes to the `bruteForce` method to crack the password hashes. You can assume that they are weak passwords, no longer than 5 characters. You will have to modify the code to include numbers and common symbols to crack all of them.

CHECKPOINT - Display the list of usernames and passwords.

In the source code, you will find the `hashPassword` method which uses the “PBKDF2WithHmacSHA512” algorithm to generate a strong hash.

7. Examine the `hashPassword` method and understand how it works.
8. Write a `generateHash(String password)` method that calls `hashPassword` at some point and that results in a strong hash by
   a. generating a 32-byte salt using an appropriate random number generator,
   b. running for about half a second,
   c. producing a hash that is 512 bytes long,
   d. returning a string formatted as “<iterations>, <salt>, <hash>” where <salt> and <hash> are hexadecimal strings (look at the `encodeMD5()` method for how to do this).

CHECKPOINT - Show the `generateHash` code. Show that it runs properly.

In the lecture, we have seen a number of secure coding practices (with respect to parameters, overflow, exceptions).

9. Identify the parts of the source code that can be improved and carry out the necessary modifications. For parameters, you only need to perform callee validation within the public methods.

The deadline for checkpoints is 2 April 2017. You may send your source code to nve@cs.stir.ac.uk to validate your checkpoints (the message should be sent from your University email address).