Security and Risk Assessment1

Due: Week 17 (26 February, Friday, 4pm)

Instructions

1) This exercise covers security policy, cryptography, malware, and password cracking.
2) Some of the material may not be covered by the time the exercise is released. You will be expected to work on the parts that have been covered first.
3) Only a single attempt is allowed for both quizzes so only press submit when you are happy with your answers).

Part 1: Security Policy and Cryptography quiz (26%)

On moodle, complete the quiz titled “Assessment 1: Policy and Crypto Quiz”. This is made up of a mix of multiple choice questions and free form answer questions.

Part 2: Malware (25%)

On moodle, complete the quiz titled “Assessment 1: Malware Quiz”. This is a mix of multiple choice questions, and longer free form answers.

Part 3: Password Security (49%)

Download the password file “password.txt” from the course moodle page. This file contains a number of password hashes in SHA-256 format. Your task is to produce a password cracker using Java. If you wish to use another language then please speak to one of the TAs ASAP. You should design your software such that it matches as many passwords as possible in 10 minutes. Your program should output the found passwords to a file, “output.txt”, where each line is of the format “username:password”. For example:

```
user1:password
user2:monkey
user3:hjjdfl
```

We will test your software on a new file, and the majority of the mark will depend on how many of the passwords are found from this second file in 10 minutes (the mark will be based on the range of password counts across the class).

For the best results, take advantage of as many processor cores available on the machine. Your code will be tested on a i7 quad-core linux (Centos) desktop with 32GB of RAM.

The software should make use of a dictionary. You will need to decide what to include within the dictionary. You may not use any dictionaries included with John the ripper. You can then use this dictionary as input to your cracker, and apply transformations on the dictionary contents to generate further guesses. You will lose marks if you only apply guesses from the dictionary without applying transformations.

The marks will be broken down as follows:

30% - The number of passwords cracked from the assessment file (if you crack 50% of the passwords you will earn 15%).
10% - Construction of the dictionary file.
30% - Strategies employed to guess passwords.
15% - Use of multithreading to maximise CPU usage in the time allowed.
5% - Crack a single password
10% - Effective commenting of code to explain operations
You should submit a zip file called “submission.zip”. Inside, include a jar executable of your code “cracker.jar” and any extra files (dictionaries) required. Also include a folder contain your source code. Do not submit the test password file. Also include a txt file explaining how you constructed the dictionary, and what strategies you use for generating password guesses.

The lecture on Thursday 4th Feb will be an tutorial on Java hashing, threads, file IO and sockets (required for the second assessed exercise). It is optional, but will be useful if you are unfamiliar with these concepts.