**Assignment 1**

* This assignment is worth 40% of your final mark.
* Due Friday the 27th of September at 23:59.
* The word limit for this assignment is 2,100 words (+/- 10%). This does not include tables, graphs, code etc.

**Audience:**You should write as if for a non-specialist audience, but not a completely general one. Imagine that you are writing for IEEE Computer or the BCS IT Now magazine, or perhaps Scientific American or New Scientist. Alternatively (if you’re not familiar with any of those publications), imagine you are writing for a computer club newsletter.

More specifically, you may assume that your readers have some general knowledge of computing and information technology, but no specialised knowledge of cryptography, algorithms, hashing etc. Think about what you need to explain and what your readers will already know. Also, think about the type of language you use in your report.

Please see detailed instructions over-leaf.

**Instructions:**Please complete the following two tasks, presenting your submission in clearly numbered sections.

**Task 1: Developments in Hashing (20 Marks):**Hashing is a critical process in computer science that ensures data security, integrity, and efficient retrieval. Recent studies have brought to light both the vulnerabilities and advancements in various hashing algorithms, emphasising their importance in applications such as cryptocurrencies, password storage, and digital signatures. These studies have also highlighted the emerging threats from quantum computing and the need for quantum-resistant algorithms.

You are required to write a detailed report (1,050 words) on the role of hashing in modern computing, based on these studies. Your report should:

Critically analyse the role of computer science in developing and maintaining robust hashing algorithms. Evaluate the key findings of the following recent studies, you may wish to consider collision vulnerabilities of older algorithms, the robustness of new algorithms, the potential threats posed by quantum computing, or any other area relevant to developments in hashing.

Cherckesova, L., Safaryan, , Lyashenko, G. and Korochentsev, D. (2022) *Developing a New Collision-Resistant Hashing Algorithm*. Available at: <https://www.mdpi.com/2227-7390/10/15/2769> (Accessed: 01 July 2024).

Kushwaha, P. and Shirsat, G. (2024) *JETIR*. Available at: <https://www.jetir.org/papers/JETIR2402024.pdf> (Accessed: 02 July 2024).

**Task 2: Time & Space in Hashing (20 Marks):**Your task is to carry out a further investigation into the average time and space complexity of different password forms. Note, you do not need to explain any of the internal details of different hashing algorithms, these are beyond the scope of the assignment.

You must provide an introduction to this topic and then answer the following questions:

1. What does it mean to hash passwords? How does authentication, using a password hash file work? Why is this more secure than storing the passwords unencrypted? What are the important characteristics of cryptographic hash functions? **(4 Marks)**
2. What different sorts of attacks can hackers mount against password-protected systems? You should at least mention dictionary and brute force attacks. Compared with just attacking a web site’s login page directly, how does it help the attackers to have a list of password hashes? **(4 Marks)**
3. What is salt? What does it mean to salt the passwords? In what way does this help protect against hackers? In what way does it not help at all? **(4 Marks)**

You will now complete a worked example (case study) involving actual data, calculations and experiments.

1. Prepare a table listing the total number of different possible passwords for each of the following password protocols. **(6 marks)**

* Write your results using no more than two significant figures.
* Write out your answers using the most appropriate units, e.g. seconds, days, months, years, centuries, etc., kB, MB, GB, TB and so on. (You may need to look up the correct names and abbreviations for some of the larger units).
* You must explain how you have arrived at the numbers in your table.
  1. 8-digit passwords that represent birth dates of people between 15 and 65 years of age, in the format DDMMYYYY, e.g. 27071990, 09061977, 29021996.
  2. 4 lower case letters followed by 4 digits, e.g. dhxr3584, aprt9683, gyje2957.
  3. 12-character passwords that are only digits, e.g. 784392956014, 326822897940, 738846554663.
  4. 12-character passwords that are only lower-case letters, e.g. vrkuvwbedykp, scrmkceshzac, dzcpykfunfnf.
  5. 12-character passwords that are case sensitive alphanumeric (that is, each character can be an upper or lower case letter or a digit), e.g. B57vTjZg8xDQ, 8UG94Bak2wKk, 6NYt68hB9yd4.

* 1. 12-character passwords where each character could be a letter, digit or punctuation character (precisely, any printable ASCII character), e.g. 6$4b#t0CN&4x, 8\*7u#u^gUIf%, $YOd0lnI45%w.  
       
     The following table can be used to summarise time and space requirements for these password protocols:

|  |  |  |  |
| --- | --- | --- | --- |
| **Password Protocol** | **Number of possible passwords** | **Temporal for a SHA1 calculation and disk write** | **Storage space requirements of password/hash table** |
| 8-digit passwords representing birth dates (ddmmyyyy) | Possible combinations? | Microseconds etc. | Bytes, Exabytes or maybe upwards? |

1. Assume a machine takes approximately 5x10–6 seconds (i.e., 5 microseconds) to compute a SHA-1 hash of a password and write the result to a file on disk. Use this to add a column to your table for the time required to compute a dictionary of all passwords and their hashes for each type of password in Q4. **(1 Marks)**
2. Each hash, together with the associated 12-character password, takes up exactly 54 bytes of disk space when written to a plain text file. Use this to add a column to your table for the disk space required to store a dictionary of all passwords and their hashes, for each type of password in Q4. **(1 Marks)**

Write a short conclusion to part 2 in which you discuss the results of your calculations and why the more complex passwords are more difficult to crack.

**See page 6 for full marking grid.**

**File Format & Layout:**Your report should be presented in a standardised format:

* 11pt or 12pt font-sizing.
* Use of a standard, plain font: Arial, Calibri, Aptos, Garamond etc.
* One-and-a-half line spacing used throughout.
* Margins of 2.5cm on all sides.
* Your student number should be clearly visible on all pages.
* Every page should be numbered in the footer.
* The official UHI cover page must be included as the first page, or as a separate file.
* All sources must be fully referenced using the UHI referencing standards.

**Submission:**Submit your assignment via Brightspace. Please place your report, coversheet and any other required files (i.e., spreadsheets, code etc.) in a ZIP archive following the naming convention:

Student-number-SOC-assignment-1-v1.zip

If you need to re-submit, please increment the version number at the end of filename. The submission with the most recent version number will be selected for marking.

**Late Submissions:**The standard scale will be used to penalise any late submissions:

* Less than 24-hours late: 5% penalty.
* 2-5 days late: 10% penalty.
* 6-10 days late: 20% penalty.
* 10 days or more: Zero grade awarded.

**Study skills:**If you are not confident about academic writing, you may find the academic writing section of the UHI induction web site helpful: <http://induction.uhi.ac.uk/Core-Skills/induction-academic-writing/index.html>. If you have issues with spelling and/or grammar, please seek help from the study skills advisers at your college.

**Plagiarism and academic misconduct:**This is an individual assessment. Copying from or collaborating with other students is not acceptable. Everything you submit for assessment must be your own work. If you quote, refer to, or use the ideas of others, you must include an in-text citation and a full reference in your reference list. Penalties for plagiarism and academic misconduct can be severe.

**Marking Scheme:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria/Mark** | **A** | **B** | **C** | **D** | **Fail** |
| **Task 1: Developments in Hashing (20 Marks)** | Provides an in-depth and critical analysis of the role of computer science in developing and maintaining robust hashing algorithms. Demonstrates a deep understanding of both theoretical and practical aspects.  Thoroughly evaluates key findings from the specified studies. Demonstrates insightful connections and implications of these findings.  Report is exceptionally well-structured, with clear, logical flow and coherent arguments. Writing is clear, concise, and free of grammatical errors.  All references are correctly formatted in the UHI reference style. Reference list is complete and accurate. | Provides a strong analysis of the role of computer science in developing and maintaining robust hashing algorithms. Demonstrates good understanding with minor gaps in theoretical or practical aspects.  Evaluates key findings from the specified studies. Shows good connections and implications of these findings.  Report is well-structured, with clear, logical flow and coherent arguments. Writing is clear with minor errors.  Most references are correctly formatted in the UHI reference style. Reference list is mostly complete and accurate (one or two minor errors). | Provides a basic analysis of the role of computer science in developing and maintaining robust hashing algorithms. Demonstrates adequate understanding with some gaps in theoretical or practical aspects.  Provides a minimal evaluation of key findings from the specified studies. Shows some connections and implications of these findings.  Report is adequately structured, with mostly coherent arguments. Writing is clear but with numerous errors.  Some references are correctly formatted in the UHI reference style. Reference list is partially complete and accurate with noticeable inconsistencies. | Provides a minimal analysis of the role of computer science in developing and maintaining robust hashing algorithms. Demonstrates limited understanding with significant gaps in theoretical or practical aspects.  Provides a lacking evaluation of key findings from the specified studies. Shows minimal connections and implications of these findings.  Report is poorly structured, with unclear flow and weak arguments. Writing is unclear with consistent errors.  Few references are correctly formatted in the UHI reference style. Reference list is incomplete and inaccurate with major inconsistencies. | Provides no meaningful analysis of the role of computer science in developing and maintaining robust hashing algorithms. Demonstrates no real understanding of theoretical or practical aspects.  Fails to evaluate key findings from the specified studies. Shows no connections or implications of these findings.  Report is very poorly structured, with no clear flow or arguments. Writing is very unclear with pervasive errors.  Reference list is presented in clear contrast to UHI referencing guidelines. |
| 14-20 | 12-13 | 10-11 | 8-9 | 0-7 |
| **Task 2: Time & Space in Hashing (20 Marks)** | See inline mark  distribution. | See inline mark distribution. | See inline mark distribution. | See inline mark distribution. | See inline mark  distribution. |
| 14-20 | 12-13 | 10-11 | 8-9 | 0-7 |