The University of Melbourne
School of Computing and Information Systems
COMP90043 Cryptography and Security

Assignment 2, Semester 2 2022

Due Date: September 22, 23:59

Objectives

To improve your understanding of RSA, hash functions and MAC.
To develop problem-solving and design skills. To improve written communication skills.

Questions

1. [10 marks] Use two primes 204550067093074444840216461869475183610920960 1 and 129040916265035916988720502542985710751370690432000 01 to construct a RSA key. Determine the smallest valid RSA public exponent and its corresponding private key. **Show the detailed workings with an explanation justifying your answer**. You need to attach any code you implemented in appendix (do NOT use screenshot for code), or mention the tools you used.

2. [25 marks] We discussed in the subject that a message M (encoded as an integer) can be signed using RSA private key by $S = M^d \bmod n$ and verified by the corresponding public key by $M' = S^e \bmod n$ and check whether $M == M'$. This mechanism can also be applied to sign a digital document, such as a PDF file. When signing a digital document, the message to be signed is the hash value of that document $H(\mathit{document})$.

   A common way to store an RSA (public) key is to use a certificate. We will discuss more about certificates later in the semester. A certificate typically includes an entity name (identity of the owner), an RSA public key, as well as some additional information (valid date range, intended key usage, etc.). This assignment specification document is signed by two certificates using RSA signature we discussed.

   (a) Extract both certificates and retrieve their public keys. Document the process and show the two RSA public keys (in decimal). You may include screenshots (with short explanation) to show steps. Any code and/or command used should be included in appendix (do NOT use screenshot for code and command).

   (b) A confidential message M is encrypted using these two RSA public keys, respectively. Show how to recover the original message. The encrypted messages are given below.

   Encrypted by Lianglu's public key:

   17405716623205410298052814800766685925161506190140393230628812783475352
   49471997136630021146401220661431626180478500131883548775588627375593995

0508993627974045126735178142188390038900681004381463746766413654184790 1
3839882701711496909746177516251886942701434388688795991614210587867930 0
6875156365762501799386044660730313435792034217845801469670789658625672 0
8940386202179381507500467360891079403050331426140878985497212431247495 7
6502211298289541472246359263828161906455036354779494649746596695420246 7
4536203715023164703467981879769720972912074037784936554187975471316779 5
3988352654382895730729385944489296226777559540186

Encrypted by Wenjun's public key:

2489100810653493688494556938827808045637182027973197822819795686671620 5
5771212409240890676588040607515953220047794543925590611881896918909011 7
6926660455877982155423149513916342129303570772675407229476844504394095 1
8869010746691449873984540914208509474336948806199732750753528043892009 9
2533517783154505477964135439767925587459452336193274787998448313054708 0
2286930239296404660771988911333110044990456187016422077826603199251336 8
5489814651335199264961658109162491215398084085814059241676649697510469 9
1911121267088510487217177079124890651646616558592658947644994750944801 7
643006419555223982191765093717651555884181845734 3

3. [12 marks] The Diffie-Hellman (DH) key agreement protocol is vulnerable to a man-in-the-middle attack. Is it possible to secure the key agreement protocol against this attack by using each of the following primitives? **If yes, sketch the method. If no, give reasons.**

   (a) Public Key Digital Signatures (where the public key of each entity is known to everyone in the system)

   (b) Hash functions (where the hash function is secure and known to everyone in the system)

   (c) Message Authentication Codes (with a prior agreed key)

4. [15 marks] Consider the following hash function. A message M is represented by a sequence of integers $\{M_1, M_2, M_3, ..., M_m\}$ such that $0 \leq M_i < n$. We can assume that n is large enough. The hash value is calculated by:

$$H(M) = (\textstyle\sum_{i=1}^{m}(M_i)^3) \bmod n$$

Does this hash function satisfy each of the following requirements? Justify your answers (with examples if necessary).

   (a) Fixed output size

   (b) Efficiency (easy to calculate)

   (c) Preimage resistant

   (d) Second preimage resistant

   (e) Collision resistant

5. [13 marks] Suppose user A has a message to share with n recipients. To ensure integrity, A decides to use message authentication code (MAC).

   (a) Briefly justify why integrity cannot be guaranteed if A and all recipients share the same secret key.

   (b) Suppose the recipients will not share their secret key(s) with each other. One way to ensure integrity is for A to hold n distinct secret keys, and each key is shared with only one of the recipients. When A transmits a message, each key is used to calculate a MAC tag, and all n MAC tags are attached to the message so that all recipients can verify the integrity of the message. However, to ensure integrity, it is unnecessary to use n keys if recipients are allowed to hold multiple keys. If a recipient holds multiple keys, all these keys are used to validate the message. The message is only deemed valid by this recipient if all MAC tags corresponding to these keys are valid. For n recipients, what's the minimum number of keys needed to ensure integrity? Which key(s) should each recipient get? Justify your answers.

   (c) If two users are allowed to share their secret keys with each other, how does this affect the scheme you proposed in (b)? Does it still guarantee integrity of the message? Briefly justify your answer.

Submission and Evaluation

- You must submit a PDF document via the COMP90043 Assignment 2 submission entry on the LMS by the due date. Handwritten, scanned images, and/or Microsoft Word submissions are not acceptable — if you use Word, create a PDF version for submission.

- Late submission will be possible, but a late submission will attract a penalty of 10% available marks per day (or part thereof). Requests for extensions on medical grounds will need to be supported by a medical certificate. Any request received less than 48 hours before the assessment date (or after the date) will generally not be accepted except in the most extreme circumstances.

- This assignment will be marked out of 75 marks, and will contribute to 7.5% of your total marks in this subject. Marks are primarily allocated for correctness of your thinking and clarity of your communication, rather than (only) the correct result without sufficient justification.

- We expect your work to be neat — parts of your submission that are difficult to read or decipher will be deemed incorrect. Make sure that you have enough time towards the end of the assignment to present your solutions carefully. Time you put in early will usually turn out to be more productive than a last-minute effort.

- You are reminded that your submission for this assignment is to be your own individual work. For many students, discussions with friends will form a natural part of the undertaking of the assignment work. However, it is still an individual task. You are welcome to discuss strategies to answer the questions, but not to share the work (even draft solutions) on social media or discussion board. It is University policy that cheating by students in any form is not permitted, and that work submitted for assessment purposes must be the independent work of the student concerned.

  Please see `https://academicintegrity.unimelb.edu.au`

If you have any questions, you are welcome to post them on the Ed discussion board *so long as you do not reveal details about your own solutions.* We encourage you to make your questions public, so that your classmates may also benefit from the discussion should they have a similar concern.