Practical 2

This practical is worth 45% of the continuous assessment portion of this module and is due 21:00 on Thursday November 17th. Submit a report in PDF with answers to the questions, and where applicable include any programs.

Diffie-Hellman key exchange

We have seen how the Diffie-Hellman key exchange can be used to generate symmetric keys in lecture 6.

Q1 (10%) In the context of Slide 16 of Lecture 6, suppose that a third party Carol is able to intercept the conversation between Alice and Bob. That is, Carol has the ability to read messages sent between them, and also send messages to Alice or Bob while posing as the other party.

Describe how this poses a threat to the security of the message transfer between Alice and Bob. Include in your answer Carol's actions in sabotaging the conversation.

Identify a procedure from the lectures that could mitigate this.

RSA

In lecture 8, we looked at how we could generate public and private keys for RSA. Suppose that Bob wishes to send a message x to Alice using RSA.

Q2 (10%) **Which** of $n, p, q, i, j, \varphi(n)$ from Slide 2 of Lecture 8 should be known to Bob or Alice, and which of them should be kept secret from everyone?

For each variable that should be known, **justify** its use in RSA.

For each variable that should be kept secret, **describe** how one could use it to break the encryption.

Modern day implementation of RSA uses the Chinese Remainder Theorem Euler's theorem for more efficient decoding. In the remaining questions of the section, we shall see how these number theory results allow us to work with smaller numbers.

The Chinese Remainder Theorem (CRT) states that if n_1 and n_2 are coprime, and then there exists a unique b with $0 \leq b < n_1 n_2$ such that $b = a_1 \pmod{n_1}$ and $b = a_2 \pmod{n_2}$. Moreover, $b = a_1 k_2 n_2 + a_2 k_1 n_1 \pmod{n_1 n_2}$, where k_1 and k_2 are integers such that $1 = k_1 n_1 + k_2 n_2$.

Q3 (30%) **Discuss** how we can use CRT to store $m = y^j \pmod{n}$ as $m_p = y^j \pmod{p}$ and $m_q = y^j \pmod{q}$.

Let $\bar{q}$ be the inverse of q in $\mathbb{Z}_p$. That is, $0 \leq \bar{q} \leq p-1$ with $\bar{q}q = 1 \pmod{p}$. Further let $h = \bar{q}(m_p - m_q) \pmod{p}$.

Show how m can be recovered from p, q, m_p, m_q, h and $\bar{q}$.

[*Hint*: what are k_1 and k_2 in $\mathbb{Z}_{n_2}$ and $\mathbb{Z}_{n_1}$ respectively?]

There is another optimisation which uses Euler's theorem. The theorem states that if n, a are coprime positive integers, then

$$a^{\varphi(n)} \equiv 1 \pmod{n},$$

where $\varphi(n)$ is Euler's totient function. Note that for any prime number p, we have $\varphi(p) = p - 1$.

Q4 (15%) **Show** how we can use Euler's Theorem to compute $m_p = y^j \pmod{p}$ by computing $y^{j_p} \pmod{p}$ where $j_p = j \pmod{p-1}$.

[Hint: $j \pmod{p-1}$ is the remainder when dividing j by $p-1$]

You may wish to write a simple program to answer the following question.

Q5 (20%) Using the notation of Lecture 8, let $p = 157$, $q = 163$ and $i = 7$.

Find a suitable j.

Decode $y = 2373$ using optimisations with CRT and Euler's theorem, showing all intermediate values of $m_p, m_q, j_p, j_q, \bar{q}, h$.

State the decoded messages of 20360, 19129 and 23043.

Huffman codes

Q6 (15%) **Construct** a tertiary Huffman code with code alphabet $\{0, 1, 2\}$ for the following information source

symbol	A	B	C	D	E	F	G	H
probability	0.14	0.3	0.12	0.03	0.1	0.15	0.05	0.11

Find the average length of the code words.

Rubric

There are 6 questions/assignments in this practical. The percentages approximately indicate the proportion of time you could be expected to spend on each of the parts. Your submission will be marked as a whole however, and an exceptional achievement in one or more of the five parts could make up for small mistakes in other parts. Marking is according to the standard mark descriptors published in the Student Handbook at:

`https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/feedback.html#General_Mark_Descriptors`

You should submit your work on MMS by the deadline. The standard lateness penalties apply to coursework submitted late, as indicated at:

`https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/assessment.html#lateness-penalties`

I would remind you to ensure you are following the relevant guidelines on good academic practice as outlined at:

`http://www.st-andrews.ac.uk/students/rules/academicpractice`

Revisions

Modified on 2/11/22: Fixed typo in Q3. $b = a_1k_2n_2 + a_2\mathbf{k_2}n_1 \pmod{n_1n_2}$ becomes $b = a_1k_2n_2 + a_2\mathbf{k_1}n_1 \pmod{n_1n_2}$.