Section 4 Lecture 28 – AES Implementation - Exercises

Q1)

(i) How many possible keywords are there for 128-bit AES?

(ii) Explain why adding one extra bit to the keyword length doubles the number of keywords that need to be tested in a brute force attack.

(ii) *Moore's Law* suggests that computing power doubles approximately every two years. If 56-bit DES was broken in 1998, when does Moore's Law suggest that AES will be able to be broken by brute force?

Q2) *As discussed in the lecture, you need to have some ability with matrices to be able to do this!*

(i) Consider the following 128-bit keyword. Split it into bytes (blocks of 8 bits) and convert each byte into hexadecimal (use the spreadsheet provided if you need to). Hence write down the initial state (as a 4 x 4 matrix) of the AES algorithm with this keyword.

10101111010111100111001110011101010100001010010010010101001010101
10101010111100100101001011101011010010101110101011001010100111101

(ii) Use the AES S-box table (supplied separately) to apply the *SubBytes* step to the matrix you created in (i).

(iii) Apply the *ShiftRows* step to the matrix you obtained in (i).

(iv) Apply the *MixColumns* step to this matrix – you should do it column by column, similarly to the example in the lecture.

And you have completed a round of AES – only nine more to go!

Q3) If you have decent computing ability, can you write programs (say in Excel or Matlab) to perform some of the operations in DES and AES?