CAB340 Assessment 2
Stream ciphers, block ciphers and hash functions

1 LFSRs

The *Content Scrambling System* is a protocol which was used for digital rights management for DVDs. Its design consists of a synchronous stream cipher built from two LFSRs, one 17 bits long and one 25 bits long. The 40-bit key is loaded directly into the LFSRs, 2 bytes into the first LFSR and 3 bytes into the second LFSR with the remaining bits in each set to 1. The outputs of the LFSRs are then combined to produce the keystream. (Note: I've left out some details which are not important for this discussion.)

a. What is the maximum possible period of each LFSR? **(1 mark)**

b. If we have two periodic functions $f(t)$ and $g(t)$ and consider the function $h(t) = (f(t), g(t))$, then h will have period

$$\frac{pq}{gcd(p,q)}$$

where p and q are the period of f and g, respectively, and gcd is the greatest common divisor. Apply this principle to explain why CSS was chosen to have different lengths for the two LFSRs. **(1 mark)**

c. What is the brute force attack complexity for this cipher? **(1 mark)**

d. Imagine a similar cipher which uses the same method of loading the key and simply XORs the outputs of each LFSR to produce the keystream. Explain how this fictitious cipher leaks easy information about the key into the keystream **(1 mark)** and suggest a known plaintext attack that retrieves the key with complexity around 2^{24} or less **(1 mark)**.

e. CSS was an epic security fail. Do some reading and write **one** paragraph about the decisions around CSS which led to its eventual break. **(1 mark)**

2 Linearity in block ciphers

The Hill cipher is essentially a block cipher operating on blocks of length d where the Hill cipher key is a $d \times d$ matrix. The known plaintext attacks considered in the previous assignment are essentially attacks on the Hill cipher operating in ECB mode, but in principle the Hill cipher could be used in other modes, and it could be used on a binary alphabet.

a. Imagine using a Hill cipher as a block cipher in CTR mode. Write out the explicit formulas for the first three ciphertext blocks in terms of the key and first three plaintext blocks. Assume that d is even and the nonce and counter both have length $d/2$. **(1 mark)**

b. Explain how to launch a known plaintext attack against a Hill cipher used in CTR mode. More specifically, show how you can reduce this problem to the known plaintext attack against Hill cipher in ECB mode. **(1 mark)**

c. Imagine using a Hill cipher as a block cipher in CBC mode. Write out the explicit formulas for the first three ciphertext blocks in terms of the key and first three plaintext blocks. **(1 mark)**

d. Explain how to launch a known plaintext attack against a Hill cipher used in CBC mode. More specifically, show how you can reduce this problem to the known plaintext attack against Hill cipher in ECB mode. **(1 mark)**

3 Using block ciphers in hash functions

Recall the Merkle-Damgard construction, which uses a *compression function* $f : A \times A \to A$. A basic version is given by:

$$\begin{aligned} W_0 &= IV \\ W_1 &= f(W_0, m_1) \\ W_2 &= f(W_1, m_2) \\ &\vdots \\ W_n &= f(W_{n-1}, m_n) \end{aligned}$$

where W_n is the output of the hash function, $m_1 m_2 \ldots m_n$ is the message and IV is a constant.

a. Discuss the simplest way you can think of to use AES-128 as a compression function in such a construction. **(1 mark)**

b. To achieve security, f must be a *one-way function*, meaning that given $f(m_1, m_2)$ it should be very difficult to find any new information about (m_1, m_2). For example, if m_1 is known, it should still be difficult to find m_2 and vice versa. Does your suggestion for the previous question satisfy this requirement? Why or why not? **(1 mark)**

c. Modern hash functions have 256-bit outputs. Discuss how the security of your construction compares to modern hash functions with regards to collision resistance. **(1 mark)**

4 Message authentication codes

a. It is in principle possible to create a hash function by using CBC-MAC with a constant, public key.

i. Discuss how to find a single-block message m that hashes to a given digest d. I.e. show that this hash function is not pre-image resistant. **(1 mark)**

ii. Extend your attack to find a message of a given length n. **(1 mark)**

b. Padding can affect cryptography in surprising and subtle ways. Consider the following padding scheme for a CBC-MAC. We are given a message $m_1 m_2 \ldots m_n$ where m_j is a full block for $n < 1$, and the length of m_n is at most the length of a full block. If m_n is a full block then it is left alone. Otherwise we add 0's to the end of m_n until it is a full block. In either situation we apply CBC-MAC to the resulting (padded) message.

i. Suppose that an adversary obtains a message, which is not an integer number of blocks in length, and the CBC-MAC tag for that message assuming the above padding scheme. Explain how it is possible to easily construct another message which gives the same tag for the same key. **(1 marks)**

ii. Now suppose that the adversary is instead given a message which is an integer number of blocks in length. Explain a similar attack to above, and state any additional conditions that the message has to satisfy for the attack to succeed. **(1 mark)**

iii. Suggest a different padding method that is secure against attacks similar to those you have given above. **(1 mark)**

c. The block cipher mode XCBC was proposed to defeat the length attacks against CBC-MAC discussed in the tutorial. XCBC is defined by:

$$\begin{aligned} m_n' &= \begin{cases} m_n \oplus k_2 & |m_n| = d \\ P(m_n) \oplus k_3 & |m_n| < d \end{cases} \\ tag &= CBC - MAC(m_1 m_2 \ldots m_n', k_1) \end{aligned}$$

where the message is $m_1 m_2 \ldots m_n$, P is a padding function, $|m_n|$ is number of bits in the last block, d is the size of the blocks for the block cipher, and k_1, k_2, k_3 are secret keys.

i. Explain why this modification defeats the attack described in question 3b of tutorial 6. **(1 mark)**

ii. Explain what would happen to security of we instead had $k_2 = k_3$, so that m_n was modified in the same way, regardless of whether it was padded. **(1 mark)**