Shweta Prasad March 10, 2021

CS 2362 Problem Set 2

Collaborators: *Saloni Mehta*

Problem 1-1

(a)

Including Round 0 of ARK, we have a total of 5 complete rounds just before starting round 5. Per round, 4 words are consumed at the ADD ROUND KEY step. Thus, a total of 20 words from the key schedule have been consumed at this point.

(b)

At the end of round 8, a total of $9 \times 4 = 36$ words have been consumed. So the words $\{w_{32}, w_{33}, w_{34}, w_{35}\}$ are the ones that have been consumed at the 8^{th} round.

Problem 1-2

DES uses 64 bit blocks and a 56 bit master key from which 48 bit keys are drawn to make the key schedule. When using 16−bit or 8−bit chaining modes, we break the plaintext up into 16 bit chunks and feed it into each stage of the chaining mode, and after doing the necessary XORing, if any, pad it using PKCS7 to expand it to 64 bits in size, and then feed it to the encryption function corresponding to each block.

Another assumption we make here is that the error mentioned in this question did not occur during encryption, but during transmission of the encrypted blocks.

(a)

In 16−bit Cipher Block Chaining Mode, we find that one bad block of cipher text received can affect give 2 wrongly encrypted blocks. If C_i was the bad block received, then using it to decrypt the plaintext block ($P_i = D_k(C_i) \bigoplus C_{i-1}$) will give erroneous decryption

of P_i. Since the bad block has only one error (assuming it to be a single bit flip), we will also get an erroneous P_{i+1}, where the error will be in a single bit and the position of this error will correspond to the position of the bit flip in the bad block. This is because $P_{i+1} = D_k(C_{i+1}) \bigoplus C_i$.

Thus, blocks 2 and 3 will be affected in CBC mode.

(b)

In Electronic Code Book Mode, there is no inter-block dependence, and an error in the transmission of one block, here the second block, will affect only its decryption.

Problem 1-3

The Output Feedback Mode first encrypts a public Initialization Vector (IV) with a public encryption function E_k, where only the k is secret. The encryption of the IV is XORed with the plaintext block to give the corresponding ciphertext block. The $K_1 = E_k(IV)$ is used as the IV in the next round of encryption and so on.

If one uses the same IV in 'every execution of the encryption operations', which I assume here to mean 'Every new message encrypted', then a problem arises if the same k is used with the encryption function E_k. If this is the case, we will thus obtain the same pseudorandom sequence of keys in the sequence of encryptions, and every block at the same index will be encrypted using the same K_i. This makes the system susceptible to a chosen-plaintext attack, and one can uncover the K_i used at every block and decrypt (using the XOR property) all messages sent using this pair of k/IV.

Problem 1-4

Image `(b)` is the result of encryption using ECB mode, while Image `(c)` is the result of encryption using CBC mode.

ECB mode has the problem of plaintext-ciphertext block dependency, so similar patterns of plaintext repeated anywhere as a block would have the same ciphertext block in the encryption. therefore, patterns like edges will be encrypted in the same manner and we will end up with a result similar to what we see in Image `(b)`. Since there's only one option for what mode was used to get the third image, CBC mode for Image `(c)`. Moreover, Cipher

Block Chaining mode breaks the P-C dependency found in ECB mode, so is good for image encryption. I would argue that modes other than ECB will result in a similar encrypted output.

Problem 1-5

AES−128 has 10 rounds and an additional $0th$ round with just the ARK step. Additionally, the last round does not have a MIX COLUMNS operation. This is per block of data. Thus for 3 blocks:

a . 33 ARKs

b . 30 SBs

c . 30 SRs

d . 27 MCs

Problem 1-6

(a)

$1111 \implies f$
$0101 \implies 5$

$f5 \implies e6$ (from the table in the slides)

$e \implies 1110$
$6 \implies 0110$

Final answer: 11100110

(b)

The state array is:

95	6c	5c	56
ee	50	35	cf
66	77	b5	12
42	16	a7	0b

(c)

The second and fourth blocks in the plaintext are the same. If you break the ciphertext given into chunks of 32 characters each, you will be able to observe this. Since AES uses 128−bit blocks, and since we know that 1 hexadecimal number represents a nibble, we know that the given ciphertext corresponds to 4 AES blocks. Thus, we arrive at the above conclusion.

Problem 1-7: What's so special about S-Boxes?

DES was designed with resilience to differential cryptanalysis in mind. The S-Boxes are the only components in DES that provide non-linearity to the system. The tables used within the S-Boxes were designed with the intention of **providing non-linearity** to the s-p network, and several criteria were specified that had to be satisfied by the S-Boxes so that non-linearity could be attained. The criteria used are mentioned in the paper titled "The Data Encryption Standard (DES) and its strength against attacks" by D. Coppersmith ref.

The designers of DES developed some measure that showed how resilient a given substitution table was to differential cryptanalysis. They made a large number of such tables and measured their performance against the set criteria (as listed in the paper cited) and chose the set of encodings that performed the best.

Problem 1-8

Submitted in the folder.