Furkan Reha TUTAŞ 21036
CS 535 HW2-Part A

Table for Pre-Processing and Post Processing:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	8	4	C	2	A	6	E	1	9	5	D	3	B	7	F
F	7	B	3	D	5	9	1	E	6	A	2	C	4	8	0

Actual Message That Sender Wants to Send: AB0D291AA74D

Pre-processing rules before CRC: For the first 4 bytes starting from the most significant; Read backwards and take 1's complement (third row). For the rest; Read backwards and take the corresponding hex from the second row.

Post-processing rules after CRC: Read backwards and 1's complement (third row).

What Sender Does:

Pre-process the message: AB 0D 29 1A A7 4D = BA D0 92 A1 7A D4 = 2A 4F 6B A7 E5 B2

Pre-processed message (M') = 2A4F6BA7E5B2

CRC-32(M') = FB20A685

Post-process the output of CRC-32: FB 20 A6 85 = 58 6A 02 BF = 5E 9A FB 20

CRC (M) = 5E9AFB20

Pads the message and CRC output: AB0D291AA74D | 5E9AFB20

XOR with k (1^n) = M|CRC (M) $\oplus$ k = AB0D291AA74D5|5E9AFB20 $\oplus$ FFFFFFFFFFFF|FFFFFFFF = 54F2D6E558B2|A16504DF

Sends ciphertext (54F2D6E558B2|A16504DF) to receiver.

What Attacker Does:

Creates ΔM as follows: Creates an empty ΔM that has the same length of the sender's message and put 1's at positions (4,6,10,20,24 and 36) and all 0's at the remaining positions.

ΔM = 000800880228

Pre-process ΔM: 00 08 00 88 02 28 = 00 80 00 88 20 82 = FF EF FF EE 40 14

Pre-processed ΔM (ΔM') = FFEFFFEE4014

CRC-32 (ΔM') = 68583A42

Post-process the output of CRC-32: 68 58 3A 42 = 24 A3 85 86 = BD A3 E5 E9

CRC (ΔM) = BDA3E5E9

Pads ΔM and CRC output: 000800880228| BDA3E5E9

XOR with ciphertext that sender tries to send: ΔM |CRC (ΔM) $\oplus$ ciphertext =

000800880228| BDA3E5E9 $\oplus$ 54F2D6E558B2|A16504DF = 54FAD66D5A9A|1CC6E136

And sends 54FAD66D5A9A|1CC6E136 instead of ciphertext (that sender wants to send) to receiver.

Furkan Reha TUTAŞ 21036
CS 535 HW2-Part A

What Receiver Does:

Receives 54FAD66D5A9A|1CC6E136 and XOR it with k (1^n).

Plaintext = ΔM |CRC (ΔM) $\oplus$ k = 54FAD66D5A9A|1CC6E136 $\oplus$ FFFFFFFFFFFF|FFFFFFFF =

AB052992A565|E3391EC9

Actual Message that sender wants to send: AB 0D 29 1A A7 4D

Received Message that attacker sends: AB 05 29 92 A5 65

Bits in actual and received message to check whether the bits are correctly flipped:

AB 0D 29 1A A7 4D = 101010110000110100101001000110101010011101001101

AB 05 29 92 A5 65 = 101010110000010100101001100100101010010101100101

In order to do cross check;

Pre-process plaintext-message: AB 05 29 92 A5 65 = BA 50 92 29 5A 56 = 2A 5F 6B B6 A5 A6

Pre-processed plaintext-message (plaintext-message') = 2A5F6BB6A5A6

CRC-32 (plaintext-message') = 6C876338

Post Process the output of CRC-32: 6C 87 63 38 = 83 36 78 C6 = E3 39 1E C9

CRC (plaintext-message) = E3391EC9

Since CRC (plaintext-message) and plaintext-CRC are equal to each other; Attack Succeeds..