09CS 416 Assignment 2
13D070003 | 13D070058 | 120070009

Question 1 -- 4.2

Hill cipher with m = 3

(a) Plaintext = (V O W) = (22 15 23)

c = pK mod 26; where c is ciphertext and p is plaintext

pK = [22 15 23] $\begin{bmatrix} 25 & 3 & 7 \\ 5 & 9 & 21 \\ 11 & 8 & 13 \end{bmatrix}$ = [878 385 768] mod 26 = [20 21 14]

c = [T U N]

Ciphertext is (T U N)

(b) Ciphertext = (T Q X) = (20 17 24)

c = pk

p = cK^-1

K X K^-1 = I (mod 26)

K^-1 = d^-1 * Adj(K)

Adj(K) = $\begin{bmatrix} -51 & 17 & 0 \\ 166 & 248 & -490 \\ -59 & -167 & 210 \end{bmatrix}$

Det(K) = -1190

We want (-1190)*X = 1 (mod 26) => 1190*X +1 = 26*Y

This is not possible since LHS id odd and RHS is even.

Hence K^-1(for Hill cipher) doesn't exist. No plaintext!

Question 2. 4.10

(A) Stream ciphers typically operate on bits unlike block ciphers, ex one time pad.
Practical stream ciphers typically generate a pseudo random keystream as a function of fixed length key and per message bit string and key is known to both sender and receiver. Moreover stream ciphers are usually faster than block ciphers.

(B) Block ciphers use more complicated circuits than stream ciphers.

Question 3. 5.6

Cipher Block Chaining (CBC) mode adds a feedback mechanism to the encryption scheme. In CBC, the plaintext is exclusively-ORed with the previous ciphertext block prior to encryption. This has the effect of 'Randomizing' the input. In this mode two identical blocks of plaintext never encrypt to the same ciphertext.

But if positions of XOR gate and the encryption box are swapped then two identical blocks of plaintext cannot guarantee different ciphertext owing to encryption before XORing. Randomizing of the output takes place instead of input which causes security threat.

Question 4. 7.2

CRC is a linear fn. as a result, even if the CRC is encrypted with astream cipher that uses XOR as its combining operation (or mode of block cipher which effectively turns it into a stream cipher, such as OFB or CFB), both the message and the associated CRC can be manipulated without knowledge of the encryption key; this was one of the well-known design flaws of the Wired Equivalent Privacy (WEP) protocol.

Here is no authentication, an attacker can edit a message and recompute the CRC without the substitution being detected. When stored alongside the data, CRCs and cryptographic hash functions by themselves do not protect against intentional modification of data. Any application that requires protection against such attacks must use cryptographic authentication mechanisms, such as message authentication codes or digital signatures.

Unlike cryptographic hash functions, CRC is an easily reversible function, which makes it unsuitable for use in digital signatures.

Q5 DES
Message : 'The quiter you become , The more you are able to hear'
Changed Message : 'The quiter you become . The more you are able to hear'
Key : 5SaalKejriwal
Changed Key : 50SaalKejriwal

DES

Original Message

ECB Mode :
5361 6c74 6564 5f5f 25a5 c10a f100 9272
8f0e 2003 e960 f863 63eb ce9e b77f 1441
775f c90e e67d f048 ea68 4be2 98ba a3f2
5745 7b56 3173 7156 4e0c fdd2 9d61 7cbc
ba99 a789 e0f1 d0eb 3fa2 8284 7495 d285
CBC Mode :
5361 6c74 6564 5f5f c798 fd6f 5d04 6ebe
2a8c 3a2f cb15 f259 ee51 649e 8539 5c40
cfae 28aa 496b b304 f55c aa00 8b8d b952
21d3 d8fe e9eb d2ba 1390 8529 6c45 78a3
14dc c46f fd2e 8518 1be0 9bd3 7ce5 497f

Changed Message
ECB Mode :
5361 6c74 6564 5f5f 2211 6bd3 53c6 454e
ed37 90da 92dc b94c 41f1 029b 1355 02b8
6fe0 ae9a bd97 7c63 eb1c ff86 6753 58bb
a498 1018 139c 5fc0 dd50 6904 3acc aa4b
bf78 cf40 ca1e 0aa1 ca9a 245a 2ebc e25b

CBC Mode :
5361 6c74 6564 5f5f 96bc dadc 004a 278e
9810 d2b2 b461 47cb 5c66 f831 2dff a118

af5d 7e82 0f21 9608 f25f 5ea5 afab 2483
2352 42c4 f9e6 cf26 35bd 58ee e138 ecb0
4b6b 2840 6c56 4f19 96fd 7cca 92c1 849c

Observation and Explanation :
There are several Bits that have changed in the ECB and CBC Mode and the change is more vivid and evident in the CBC mode as the ECB mode is quite a raw mode of ciphering the text compared to a professional and chained form of CBC which leaves very few chances of plaint-text guessing with the same key and changed text.

Key Changed (Message Same as Original) :
ECB :
5361 6c74 6564 5f5f b55d 4396 c27e b3c0
36a1 5ca6 d30b 26a3 ea90 364b 9ac2 6cb5
23f9 d210 d851 2425 a6bb a4b3 1996 2263
073b 7014 4bbf 8a46 d867 0206 bbbf 39db
bb46 be98 e99f 2eea 92c4 7e78 19cc f04b
CBC :
5361 6c74 6564 5f5f f147 8cb6 bb38 55d1
6144 ead3 7501 fa23 f317 a256 8575 0b97
8b2f cdcd ad31 6c8d 31e3 3fe6 9038 13b0
8171 9aa0 2278 99fd 3c4d 4510 463e 4868
75f3 4af5 d9d4 34da d0ee 93d9 a5ed b453

Observation : The change observed in the cipher text compared after changing one bit of the key is quite more than changing the message altogether. The avalanche effect is quite more(bits changed in output / bits changed in input) when key is changed by a bit. Also the effect is quite more evident in the CBC mode due to its chained architecture.

Q6 SHA1 Hashing :

Unchanged Message :
Hash : 6059eed6dad3c57ab928a37a6f0f7075b104d012

Changed Message :
Hash : ac63ec61cb1c73e0205680a2b46a3ffebabf9b7c

Observation :
Due to Superior Design of the SHA-1 Hashing algorithm , the output with 1 bit changed input shows an desirable avalanche effect. ie The output changes dramatically with a minor change in the input.

Explanation :
As the SHA-1 Algorithm involves several <<<n steps and a 2^23 modulo adder , it is quite obvious that a single bit change in the input can cause the output change to this magnitude.

Q7a DVWA

Vulnerability : The functions in the DVWA application that serve as its hardcore security feature are **is_numeric() and stripslashes()** function in the DVWA source will render it helpless to all the attacks from several universally true strings.

Attack Vector : '1' OR '1'='1'
This same vector works now as the feature of is numeric and slashstripping will not detect the vexctor as it did in the earlier case and hence we can break into the system easily if this 2 functions weren't present . Reason

being that when we dont have the isnumeric check even string matching the universal case return true and when strip slashes is absent the effect of similar OR and its conditions can be over-ridden (magic Quotes).

Q7b SQL Injection

1 - Get TableName :

Vector : **' OR EXISTS(SELECT 1 FROM dual WHERE database() LIKE '%<YOUR GUESSED CHAR HERE>%') AND ''**

2 - Get Number of Rows :

We first find an entire column by guessing a sample character from its name (see below) and then use the function SQL_COUNT() for that column to
Vector : **SELECT table_name, column_name FROM information_schema.columns WHERE table_schema = '<POSSIBLE COLUMN NAME OR CHARACTER>'**

Followed by #row = SQL_COUNT(<columnName here>)

3 - To get Column Names and Number

For Name :
Vector : **SELECT table_name, column_name FROM information_schema.columns WHERE table_schema = '<POSSIBLE COLUMN NAME OR CHARACTER>'**
For Number :
Vector : *http://www.w3schools/SQLInjection.php?id=1 ORDER BY **n—***

Note that the n-- in the end is the determining factorin finding the column number . changing it from time to time in seeral iterations helps us find the number of columns by hit and trial method as it gives true for the value of n which is == the value of number of columns.