Problem 1 Explanations

Part a - Kerckhoff's Principle

Kerckhoff's principle states that a cryptosystem should be secure if everything about the system is publicly known, except for the encryption key. This principle is important because it means that cryptosystems should be secure because they are complex, not because they are convoluted or ambiguous. Cryptosystems which are the latter are generally much weaker than systems which are publicly known such as RSA.

Part b - Perfect Security

Perfect secrecy is a cryptosystem where absolutely no information about the plaintext is leaked through the ciphertext, and every key in the keyspace has an equal probability of being produced from any plaintext. Computational secrecy is where some (although very little) information about the plaintext is leaked through the ciphertext, however there is a very low probability that the information will be determined by evesdroppers with limited computational power. Modern cryptosystems such as AES are not perfectly secret because perfect secrecy takes much more complex systems, and also because it is not required. Even with computational secrecy, it would take an insane amount of computational power to brute force the correct key for the system.

Part c - OTP

OTP or one time pad is a cryptosystem with perfect secrecy. The cipher can not be cracked without the original key since the key is always just as long if not longer than the plaintext and truly random. This cipher is not used in practice for two reasons. Firstly, the key must be as long as the plaintext, or longer. This makes keys extremely long and often too large for practicality, and true randomness can not be replicated by computers. Additionally, The original key used to encrypt the plaintext must be used to decrypt it. This means that if Alice encrypted a message with OTP and sent the ciphertext to Bob, she would also need to send Bob the key to decrypt it, which defeats the purpose of having a secure line of communication in the first place. Instead, we see cryptosystems such as the Diffie-Hellman key exchange where both parties have their own private key and don't need to exchange it with anyone.

Part d - Stream Ciphers

Stream ciphers are symmetric cryptosystems which work by utilizing bit shifts and XOR operations to create pseudo random numbers. Certain bits are selected to be XORed, determined by the key, and any number of iterations can be performed to produce different ciphertexts, up to the length of the key. Stream ciphers are very similar to OTP in the sense that they both use XOR operations to encrypt data, however the major difference between the two is

that OTP has much larger keys which are truly random, not generated from predictable and reversible bit shifts of a stream cipher.

Problem 2 RSA

(skipped)

Problem 3 Hash Functions

Part a - Properties Of Cryptographic Hash Functions

1. Preimage Resistance
    a. Hash functions should be computationally difficult to reverse
2. Second Preimage Resistance
    a. Given an input and its hash, finding a different input with the same hash should be difficult
3. Collision Resistance
    a. Two different inputs of any length should not result in the same hash

Part b - Bit Length Sufficiency

The output of 128 bits is sufficient for an output length. This means that there are 2^{128} possible hashes, providing an abundance of hash space.

Part c - Salting Passwords

The purpose of adding a salt to passwords is to enforce the second preimage resistance that given an input and its hash, finding a different input with the same hash should be difficult. Salting achieves this by appending a string onto a password then encrypting it, so the same password would have multiple different hashes depending on the salt

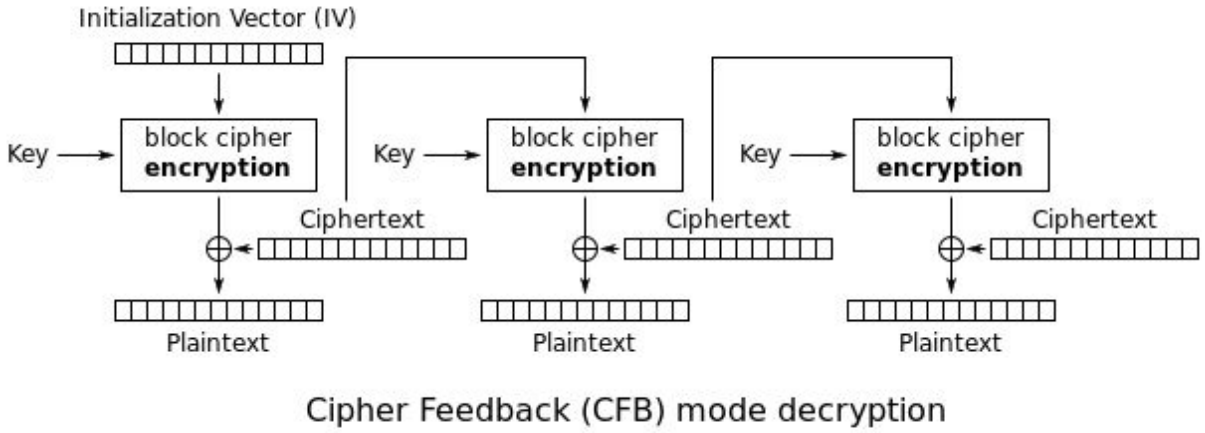
For example:

Password	Salt	Hash
Orange	book	12345678
Orange	shelf	abcdefgh
Orange	remote	!gjTEL901

Problem 4 Modes Of Operation

Part a - CFB

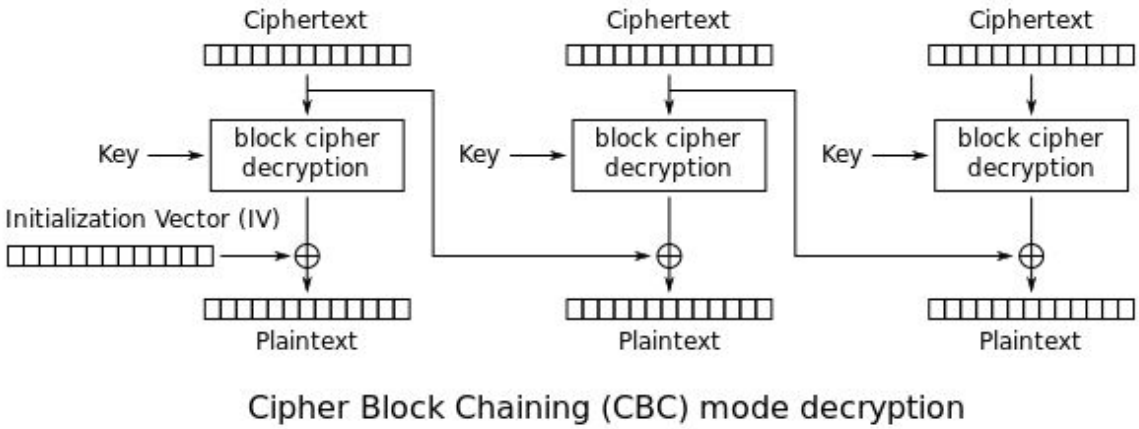
CFB takes a segment of encrypted text from length one to the block size (inclusive) and performs an XOR operation on a segment of plaintext of the same length, so if an error occurred in block y_i then Bob will notice an error in the block of plaintext that was encrypted/decrypted with CFB block y_i as well as the block right after it, y_{i+1}. This is because one ciphertext is used to decrypt two blocks, so two blocks of plaintext will have errors. Block y_i will have a singular bit error, whereas block y_{i+1} will be totally corrupted (assuming there is a single bit error in transmission)



Cipher Feedback (CFB) mode decryption

Part b - CBC

CBC uses a given ciphertext and a key to decrypt plaintext. If an error occurred in transmission, then one of these ciphertexts will be wrong (y_i). For CBC, the diagram below shows that if the first ciphertext is wrong, let's call it y_i, then the block of plaintext decrypted from y_i and the plaintext decrypted from y_{i+1} would also be wrong. Similar to CFB, this is because one ciphertext is used to decrypt two blocks, not just one. The first block y_i will be totally corrupted, while block y_{i+1} will have a singular bit error (assuming there is a singular bit error in transmission)



Cipher Block Chaining (CBC) mode decryption

Problem 5 Passwords And Keyspaces

Part a

Password has 8 characters, 128 options, 7 bits each

The number of keys for this setup would be $(128)^8$

To find the total number of bits that they keys would take up:
(number of possible characters)$^{\text{number of characters}}$ * (number of characters * bits per digit)

= $(128)^8$ * (8*7)

Part b

Password has 8 characters, 26 options, 7 bits each

In this case, the number of keys would be $(26)^8$

To find the total number of bits that they keys would take up:
(number of possible characters)$^{\text{number of characters}}$ * (number of characters * bits per digit)

= $(26)^8$ * (8*7)

Part c

Password has 8 characters, 26 options

The probability of guessing such a password such as in part b would be 1/26 chance for each letter, 8 times in a row.

$= (1/26)^8$

Part d

Because the maximum length of the password is 8 characters, using random characters would be much more secure than using words or phrases. Words are a subset of the set of 8 character long passwords created by every possible character, therefore there are less possibilities for passwords. If the password were allowed to be longer with words, then passphrases would be more secure. However with the same length of password, random characters will be more secure. If words were used, then letters that have a higher frequency in the english language will be more likely, thus making the password easier to guess.

Problem 6 Evaluating Using Algorithms

Part a - Extended Euclidean Algorithm

Find 32^{-1} mod 101 using extended euclidean algorithm

$t = t_1 - gt_2$
r = remainder of r_1/r_2
g = quotient of r_1/r_2

g	r_1	r_2	r	t_1	t_2	t
3	101	32	5	0	1	-3
6	32	5	2	1	-3	19
2	5	2	1	-3	19	-41
2	2	1	0	19	-41	101
				-41		

Thus, 32^{-1} mod 101 = -41 mod 101 = 60

Part b - Square And Multiply

We can use fermat's little theorem to rewrite $7^{45} = 32$ mod 101 as: $7^{101-45-1} = 32^{-1}$ mod 101

This means we need to calculate 7^{55} mod 101 using the square and multiply algorithm. We do this by converting the exponent to binary and squaring the number where there is a 0, and squaring and multiplying by the original number when there is a 1.

55 in binary: 110111

Bit 1 (1):			= 7
Bit 2 (1):	$7^2 * 7$	= 343 mod 101	= 40
Bit 3 (0):	40^2	= 1600 mod 101	= 85
Bit 4 (1):	$85^2 * 7$	= 50575 mod 101	= 75
Bit 5 (1):	$75^2 * 7$	= 39375 mod 101	= 86
Bit 6 (1):	$86^2 * 7$	= 51772 mod 101	= 60

And we get the same answer, 60!

Problem 7 Elliptic Curve Cryptography

Given curve: $y^2 \equiv x^3 + 2x + 1 \mod 17$
Given point: $\alpha = (1, 2)$

Part a - Find 2α and 3α

Finding 2α:
$2\alpha = \alpha + \alpha = (1, 2) + (1, 2) = (x_3, y_3)$

Find gradient:
(Since P = Q we use this equation)
$\lambda = (3x_1^2 + a)/(2y_1)$
$= (2*2)^{-1}(3*1^2 + 2)$
$= 4^{-1}(5)$
$= 13(5)$
$= 65 \mod 17$
$= 14$

Find x_3 and y_3:

$x_3 = \lambda^2 - x_1 - x_2$
$= 14^2 - 1 - 1$
$= 194 \mod 17$

= 7

$y_3 = \lambda(x_1 - x_3) - y_1$
= (14)(1 - 7) - 2
= -86 mod 17
= 16

$2\alpha = (x_3, y_3) = (7, 16)$

Finding 3α:
$3\alpha = \alpha + 2\alpha = (1, 2) + (7, 16) = (x_3, y_3)$

Find gradient:
(Since P ≠ Q we use this equation)
$\lambda = (y_2 - y_1)/(x_2 - x_1)$
$= (7 - 1)^{-1}(16 - 2)$
$= 6^{-1}(14)$
= 3(14)
= 42 mod 17
= 8

Find x_3 and y_3:

$x_3 = \lambda^2 - x_1 - x_2$
$= 8^2 - 1 - 7$
= 56 mod 17
= 5

$y_3 = \lambda(x_1 - x_3) - y_1$
= (8)(1 - 5) - 2
= -34 mod 17
= 0

$3\alpha = (x_3, y_3) = (5, 0)$

Part b - Find 4α

$\alpha + 3\alpha = (1, 2) + (5, 0) = (x_3, y_3)$

Find gradient:
$\lambda = (y_2 - y_1)/(x_2 - x_1)$
$= (5 - 1)^{-1}(0 - 2)$
$= 4^{-1}(-2)$
= 13(-2)

$= -26 \bmod 17$
$= 8$

Find x_3 and y_3:

$x_3 = \lambda^2 - x_1 - x_2$
$= 8^2 - 1 - 5$
$= 58 \bmod 17$
$= 7$

$y_3 = \lambda(x_1 - x_3) - y_1$
$= (8)(1 - 7) - 2$
$= -50 \bmod 17$
$= 1$

$4\alpha = (7, 1)$

Verify:
$2\alpha + 2\alpha = (7, 16) + (7, 16)$

$\lambda = (3x_1^2 + a)/(2y_1)$
$= (2*16)^{-1}(3*7^2 + 2)$
$= 32^{-1}(149)$
$= 8(149)$
$= 1192 \bmod 17$
$= 2$

$x_3 = \lambda^2 - x_1 - x_2$
$= 2^2 - 7 - 7$
$= -10 \bmod 17$
$= 7$

$y_3 = \lambda(x_1 - x_3) - y_1$
$= 2(7 - 7) - 16$
$= -16 \bmod 17$
$= 1$

$4\alpha = (7, 1)$

Part c - Find All Points On Curve

Simply plug in all integers within the modulo range to get y values, and continue to find all multiples and sums of integer points on the curve.

Points: (0, 1) (0, 16) (1, 2) (1, 15) (2, 8) (2, 9) (3, 0) (5, 0) (6, 12) (6, 5) (7, 1) (7, 16) (8, 6) (8, 11) (9, 0) (10, 1) (10, 16) (12, 6) (12, 11) (14, 6) (14, 11) (16, 7) (16, 10)