Huanjie Guo 10655496

1.

a) Give four examples of security mechanisms. (2 marks)

Encipherment, digital signatures, access controls, authentication exchange.

b) Explain how the working of a rotor machine like Enigma relates to the ideas of substitution and permutation. (2 marks)

Rotor machines change the interconnecting wiring with each key stroke. The wiring is placed inside a rotor, and then rotated with a gear every time a letter is pressed. Every letter pressed on the keyboard increments the rotor position and get a new substitution, implementing a polyalphabetic substitution cipher.

c) Why is triple DES (even with two keys) better than double DES? (2 marks)

Namely, 2DES uses 112 key bits (two 56-bit DES keys) but offers a security level of about 2^57, not 2^112, because of a "meet-in-the-middle attack". Similarly, 3DES uses 168 key bits but offers "only" 2^112 security (which is quite sufficient in practice). This also explains why 3DES is sometimes used with a 112-bit key (the third DES key is a copy of the first): going to 168 bits does not actually make things more secure.

d) What is the hard problem used in elliptic curve cryptography? (2 marks)

The elliptic curve discrete logarithm is the hard problem.

e) What is the difference between the BB84 and the B92 quantum key distribution algorithms? (2 marks)

The key difference in B92 is that only two states are necessary rather than the possible 4 polarization states in BB84.

2.

a) Describe the structure of a round of a Feistel cipher. (6 marks)

In each round, the right half of the block, R, goes through unchanged. But the left half, L, goes through an operation that depends on R and the encryption key. First, we apply an encrypting function 'f' that takes two input — the key K and R. The function produces the output f(R,K). Then, we XOR the output of the mathematical function with L.

b) In a Feistel cipher, the same algorithm is used for encryption and for decryption, and yet it remains secure. Explain the main reason for this. (4 marks)

In short, because XOR is its own inverse operation. a XOR b XOR b is a again. So we can use the same algorithm to encrypt and decrypt and it remains secure. The only difference is that, in decryption, we use the round keys in reverse.

c) Various modern padding and similar schemes use a Feistel structure in their closing stages, despite the fact that all the data needed for Feistel decryption is readily visible in the ciphertext. Why is this nevertheless effective? (4 marks)

The padding and similar schemes just use a Feistel structure for the purpose of extending their size as the Feistel structure. So it doesn't matter that the decryption data is visible in the ciphertext.

d) Describe the Extended Euclid Algorithm for finding not only the GCD of two numbers x and y, but also the coefficients a and b such that GCD(x, y) = $ax + by$. (4 marks)

define a function exgcd(a,b)

if b!=0

then gcd(a,b) = gcd(b, a mod b)

assume that we get the solution of (b, a mod b), which is (x',y')

then ax+by = bx'+(a mod b)y'

then ax+by = bx'+(a–b*[a/b])y'

then ax+by = ay' + b(x' – [a/b]y')

so x=y' y=x'–[a/b]y'

The question becomes how to calculate exgcd(b, a mod b)

recur until b=0 we get the solution.

e) Why is DES encryption inadequate as a secure hash function? (2 marks)

DES has a key; the secrecy of the key is what the cipher security builds on. On the other hand, a hash function has no key at all, and there is no "secret data" on which security of the hash function is to be built.

DES is reversible: if you know the key, you can decrypt what was encrypted. Technically, for a given key, a block cipher is a permutation of the space of possible block values. Hash functions are meant to be non-reversible, and they are not permutations in any way.

a) Describe the structure of AES. (4 marks)

Encryption Process

- **Byte Substitution**: The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.
- **Shiftrows**: Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row.
- **MixColumns**: Each column of four bytes is now transformed using a special mathematical function.
- **Addroundkey**: The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key.

Decryption Process (simiar to the encryption process in the reverse order)

- Add round key
- Mix columns
- shift rows
- Byte substitution

b) In an AES round, which phases use linear mathematics and which do not? Why is it important that there should be nonlinear phases? (3 marks)

The MixColumns step uses a linear algorithm, and the SubBytes step does not use linear mathematics. Nonlinear phases can avoid attacks based on simple algebraic properties.

c) In the Diffie–Hellman key agreement protocol, why is it important to base the algorithm on a prime number that is large? (3 marks)

The security of Diffie–Hellman key agreement protocol relies on the difficulty of computing discrete logarithms. Prime numbers don't break down into smaller factors, making cracking the code or hash much harder.

d) Describe the Chinese Remainder Theorem and what it can be used for. (5 marks)

The Chinese remainder theorem states that if one knows the remainders of the Euclidean division of an integer n by several integers, then one can determine uniquely the remainder of the division of n by the product of

these integers, under the condition that the divisors are pairwise coprime. We can use it to speed up modulo computations. If working modulo a product of numbers, the Chinese Remainder theorem lets us work in each modulus separately. Since computational cost is proportional to size, this is faster than working in the full modulus.

e) In the context of block ciphers, what is an S-box? (3 marks)

An S-box (substitution-box) is a basic component of symmetric key algorithms that performs substitution. In block ciphers, they are typically used to obscure the relationship between the key and the ciphertext.

f) What is the difference between a hash and a MAC? (2 marks)

While hashes are used to guarantee the integrity of data, a MAC guarantees integrity and authentication.