Assignment 4

1. **This problem deals with digital certificates (aka public key certificates).**
   1. What information must a digital certificate contain?

The digital certificate contains the name of the user and the user’s public key. Example of how the certificate looks like:

M = (bob, bob’s public key) S = [M]CA

* 1. What additional information can a digital certificate contain?

A certificate can contain any other information that is deemed useful to the participants, example of those include emails, etc. However the more information it contains the more likely the certificate will become invalid.

* 1. Why might it be a good idea to minimize the amount of information in a digital certificate?

Because the information included in the digital certificate is public and by including too much information on the digital certificate, it might provide Trudy some information about the participants.

1. **Suppose that Bob receives Alice's digital certificate from someone claiming to be Alice.**
   1. Before Bob verifies the signature on the certificate, what does he know about the identity of the sender of the certificate?

Before Bob verifies the signature, Bob does not know anything about the sender of the certificate, assuming the certificate belongs to Alice; Trudy can simply take Alice’s certificate because it is public, and send it to Bob. Therefore, Bob does not know anything about the sender of the certificate.

* 1. How does Bob verify the signature on the certificate and what useful information does Bob gain by verifying the signature?

Bob verifies the signature by decrypting the certificate with the public key of the certificate authority that signed the certificate. By verifying the signature, Bob can ensure that the public key belongs to the person of the name inside the certificate in this case is Alice, Bob can as well ensure that Alice is the only one who has the private key, assuming her private key did not get compromised.

* 1. After Bob verifies the signature on the certificate, what does he know about the identity of the sender of the certificate?

After Bob verifies the signature on the certificate, Bob will still not know who the sender of the certificate is, because the certificate is public, the only thing Bob can know is that the owner of the certificate is Alice and only Alice has the private key.

1. **In equation (4.3) we proved that RSA encryption works, that is, we showed [{M}Alice]Alice = M. Give the analogous proof that RSA signature verification works, that is, {[M]Alice}Alice = M.**

In equation 4.3 we learn that by using the public key *e* from the equation ed = (p-1)(q-1), we can encrypt the message M, for example: C = Me mod N and in order to decrypt, Alice uses her private key d to decrypt by performing M = Cd mod N. To proof that the RSA verification works, we have to use Alice’s private key to encrypt a message and send to Bob and use Alice’s public key to decrypt. If Bob decrypts the message and then compare it to the original message, if it is the same then the signature verification works.

First Alice encrypts a message M,

S = Md mod , if we want to decrypt the S, we have to use the public key {N,e},  
M = Se mod N  
M = (Md)e mod N = Med mod N

From this, do the same as equation 4.3

Se = Med = M(ed-1)+1 = M\*Med-1 = M\*M kφ(N) = M \* 1k = M mod N

1. **Suppose that Alice's RSA public key is (N, e) = (33, 3) and her private key is d = 7.**
   1. If Bob encrypts the message M = 19 using Alice's public key, what is the ciphertext C? Show that Alice can decrypt C to obtain M.

Bob encrypts M with Alice’s public key.

C = Me mod N = 193 mod 33 = 28

**By encrypting the message M = 19 with Alice’s public key, we get ciphertext C = 28.**

Alice decrypt ciphertext C with her private key d=7

**M = Cd mod N = 287 mod 33 = 19**

* 1. Let S be the result when Alice digitally signs the message M = 25. What is S? If Bob receives M and S, explain the process Bob will use to verify the signature and show that in this particular case, the signature verification succeeds.

By digitally signing message M = 25, we get S, S = Md mod N = 257 mod 33 = 31

When Bob receives S and M, Bob decrypts S with Alice public key {33, 3} and then compare the result.

M =25

Result = 313 mod 33 = 25

After comparing, Bob can know that the verification succeeds because the result of decrypting S is equals to the original message M

1. **To speed up RSA, it is possible to choose e = 3 for all users. However, this creates the possibility of a cube root attack as discussed in this chapter.**
   1. Explain the cube root attack and how to prevent it.

By choosing e = 3 for all users, the RSA encryption becomes vulnerable, because for all messages M < N1/3, by performing the RSA encryption using e, we will get C = M3 = M3, and therefore Trudy can just cube root C, to get message M. To prevent this problem, we can pad the message with enough bits so that M will be greater than N1/3. However another type of cube root attack exist, When all users use e=3 and the same message M is encrypted by different users, yielding C0,C1,C2, Trudy can use the Chinese remainder theorem to discover message M, therefore in order to avoid a Cube Root attack, we need to pad the message with random bits or with user specific information so that the same messages can be different after encryption.

* 1. *For (N, e) = (33, 3) and d = 7, show that the cube root attack works when M = 3 but not when M = 4.*

When M=3, C = Me mod N = 33 mod 33 = 9 mod 33 = 9, Here we can see that the ciphertext is equals to the cube of M, M3 = C, and in order for Trudy to know the message M, she simply needs to take the cube root of C, 91/3 = 3, therefore the cube root attack works on message M=3

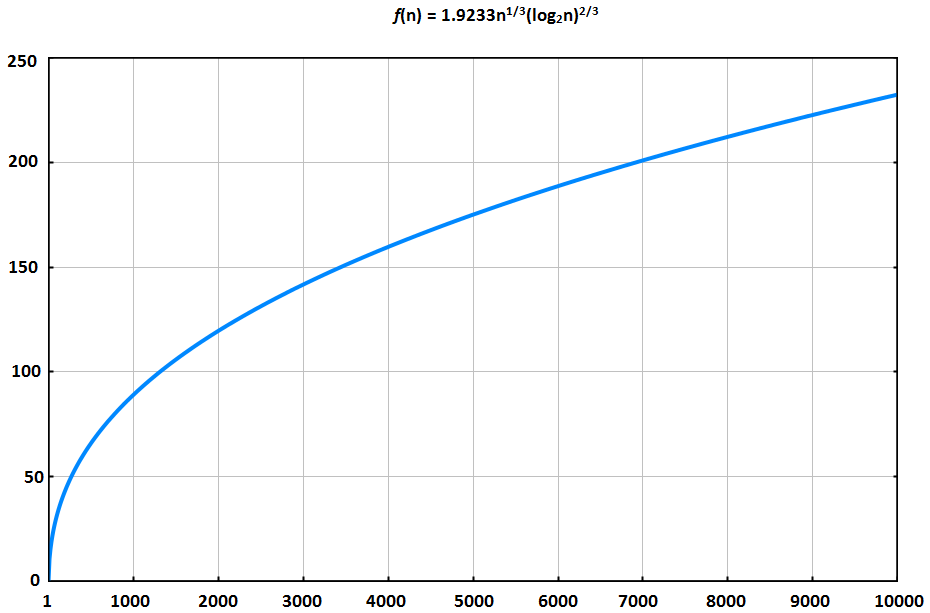
When M=4, C = Me mod N = 43 mod 33 = 64 mod 33 = 31, Here the ciphertext does not equals to M3, because from the situation mentioned on Part A, M has to be smaller than N1/3, and if we calculate N1/3=331/3 = 3.2, therefore M=4 is bigger than N1/3 and the cube root attack will not work.

1. **Consider the RSA public key cryptosystem. The best generally known attack is to factor the modulus, and the best known factoring algorithm (for a sufficiently large modulus) is the number field sieve. In terms of bits, the work factor for the number field sieve is**

***f*(n) = 1.9233n1/3(log2n)2/3**

**where n is the number of bits in the number being factored. For example, since *f*(390)≈60, the work required to factor a 390-bit RSA modulus is roughly equivalent to the work needed for an exhaustive search to recover a 61-bit symmetric key.**

* 1. Graph the function f(n) for 1 ≤ n ≤ 10,000.



* 1. A 1024-bit RSA modulus N provides roughly the same security as a symmetric key of what length?

***f*(1024)≈89.9334**

A 1024-bit RSA modulus N provides roughly the same security as a symmetric key of 90-bits or 91-bits

* 1. A 2048-bit RSA modulus N provides roughly the same security as a symmetric key of what length?

***f*(2048)≈120.742**

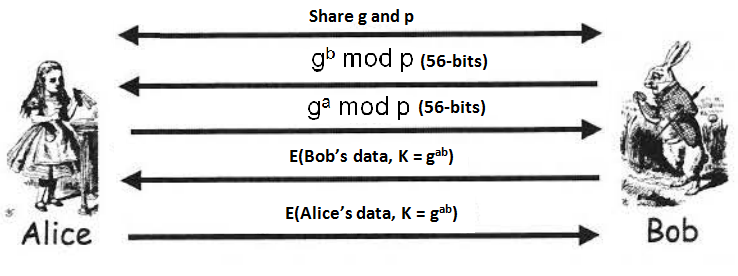
A 2048-bit RSA modulus N provides roughly the same security as a symmetric key of 121-bits

* 1. What size of modulus N is required to have security roughly comparable to a 256-bit symmetric key?

256 = 1.9233n1/3(log2n)2/3

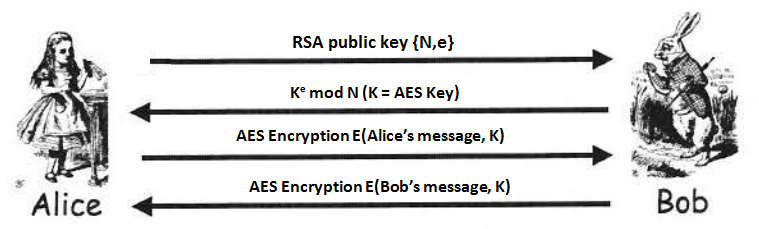
In order for RSA modulus N to provide a security roughly comparable to 256-bit symmetric key, it would require 12707-bit RSA modulus N

1. **A hybrid cryptosystem uses both public key and symmetric key cryptography to obtain the benefits of each.**
   1. Illustrate a hybrid system using Diffie-Hellman as the public key system and DES as the symmetric cipher.



In this graph, we will use Diffie-Hellman to exchange the public key of 56bits. We will use this 56bit symmetric key for DES encryption to encrypt messages. Since the key is symmetric, Alice and bob will be using the same key to encrypt and decrypt messages.

* 1. Illustrate a hybrid system using RSA as the public key system and AES as the symmetric cipher.



In this hybrid system, Alice first establish an RSA public key for bob, and bob uses the key to encrypt an AES key and send to Alice, from there on, Alice and Bob both knows the AES symmetric key, they both use the same key to encrypt and decrypt messages.

1. **Suppose that for the knapsack cryptosystem, the super increasing knapsack is (3, 5, 12, 23) with n = 47 and m = 6.**
2. Give the public and private keys.

The private key for this knapsack cryptosystem is (3,5,12,23), m-1 mod n = 6-1 mod 47 = 8

The public key is: (18, 30, 25, 44)

1. Encrypt the message M = 1110 (given in binary). Give your result in decimal.

(18, 30, 25, 44)

1 1 1 0

18 + 30 + 25 = 73, the encrypted message is **73**

1. **The man-in-the-middle attack on Diffie-Hellman is illustrated in Figure 4.2. Suppose that Trudy wants to establish a single Diffie-Hellman value, gabt mod p, that she, Alice, and Bob all share. Does the attack illustrated below succeed? Justify your answer.**

No, the attack illustrated below will not succeed. In this attack, Trudy only knows the value t, gb and ga In order to establish the value gabt, Trudy needs to know either of the value of a or b, because with the information she has in the attack illustrated below, the closest Trudy can get is gat+bt or g(a+b)t, which does not equals to gabt

1. **In the RSA cryptosystem, it is possible that M = C, that is, the plaintext and the ciphertext may be identical.**
2. Is this a security concern in practice?

Although the possibilities of M = C is very low in real life practice, there is still a security concern because if M = C, the message is exposed directly and it takes no effort for Trudy to know what the message is, and it would be a chaos if a message of high confidentiality happens to be the same as the ciphertext.

1. For modulus N = 3127 and encryption exponent e = 17, find at least one message M that encrypts to itself.

M = Me mod N = M17 mod 3127

Answers:

M = 0  
M = 1  
M = 235  
M = 295  
M = 530  
M = 531  
M = 825

1. **A digital signature or a MAC can be used to provide a cryptographic integrity check.**
   1. Suppose that Alice and Bob want to use a cryptographic integrity check. Which would you recommend that they use, a MAC or a digital signature? Why?

If Alice and Bob are only looking for cryptographic integrity check, then I would recommend them to use MAC because MAC uses symmetric keys and symmetric encryption is faster than asymmetric encryption (digital signature).

* 1. Suppose that Alice and Bob require a cryptographic integrity check and they also require non-repudiation. Which would you recommend that Alice and Bob use, a MAC or a digital signature? Why?

If Alice and Bob require a cryptographic integrity check with non-repudiation, then I would recommend them to use digital signature, because a digital signature not only offers an integrity check, but it is also non-repudiation, which is something that symmetric keys does not have.