Assignment 2: COMP90043
Due Date: September 23, 2016
Assignment is worth 7.5% of the total marks

1. Answer all the questions.
2. A Discussion forum thread Assignment 2 has been created on LMS. Any clarification offered on this forum will be considered as a part of the specification of the Assignment.
3. The total number of points for this assignment is 30. This contributes to 7.5% of the total.
4. Answers must be submitted as a PDF file via the comp90043 Assignment 1 submission form on LMS by September 23, 2016. Late submissions will attract a penalty of 10% per day (or part thereof). Please ensure your name and login name are clearly presented.

Questions

1. (9 points) This question is concerning properties of Textbook RSA cryptosystem.

   a. RSA in small parameters: Assume that Alice chooses two primes 196065871 and 102305491 to construct her RSA keys. Determine the smallest valid RSA public key and its corresponding private key for Alice. Show the detailed workings and not just the solution. You can use magma calculator from http://magma.maths.usyd.edu.au/magma/ If you use algorithms such as EEA or magma, show the workings.

   ```
   > p;
   196065871
   > q;
   102305491
   > n;
   20058615200997661
   > phin;
   ```

```
20058614902626300
> Factorization(phin);
[ <2, 2>, <3, 2>, <5, 2>, <7, 2>, <11, 1>, <13, 1>, <17, 1>, <6529, 1>, <28657,

1> ]
>
Smallest possible e is 19.


2 marks for correct calculations.
```

b. This question is about the multiplicative property of the textbook RSA algorithm.

We showed in the workshop that basic RSA is not secure for chosen ciphertext attack. The same idea can also be applied to create blind signatures. Assume that Alice's public keys are $[n, e]$ and her private key is d. Explain how Bob could create Alice's signature on a message of choice m using the concept of blinding. Note that that Bob will not have access to private key d, but can request Alice to sign a blinded message.

Your solution should also show the workings of the above blinding procedure using a random RSA key for Alice. Your answer here should include the following:

i. Your selection of two random primes, each of length at least 100 digits.
ii. the public key e be smallest valid public key.
iii. Determine the private key d.
iv. A random message m of length at least 100 digits.
v. A blinded message m_b.
vi. Signature of m through blinded process.
vii. Direct signature of m using the private key.

Note: the last two items should be identical. Any code written for the above should be included as an appendix.

```
Any example would do; This question is mainly for students to realize the issue

Review the section on blinding in RSA where an adversary sends random mes-
sages for the receiver to decrypt producing random outputs, thereby allowing
an adversary to decipher the key after interpreting the produced outputs over
a period of time.
// Code
p := RandomPrime(400);
q := RandomPrime(400);
n := p * q;
phin := (p-1) * (q-1);
e := 1;
repeat
e := e + 1;
```

```
until (GCD(e, phin) eq 1);
// Private key d
d := InverseMod(e, phin);

// Choose m
m := RandomBits(400);

// Blinded m
r := RandomBits(400);
mb := Modexp(r, e, n) * m mod n;

// Blind signature
rinv := InverseMod(r, n);
sb := Modexp(mb, d, n) * rinv mod n;

// Direct signature
s := Modexp(m, d, n);

// Output
> p;
538714886480360602243593408403891003203191872956756462862430371358599028569874362437258939898803884774925789645533741099
> q;
109118348465657696456581281486188509358096529566356856937666471789750246517301124476514805628971894773470491868728277441 9
> e;
3
> d;
391891191377341442976129378823571020673013268030030466387821307885875972217810042903523550070108899585540192989264667541275062886305531373843764641947036070136574535536931792208564581903093254713769270104196691518457542187470885509566420643
> m;
256772650355665707308497650280887952717825248847502010567185580709491522439887292598385596704831516751181135941690824580
> mb;
501927806663129100128512097345893172168283469685229211863080362639026578470600084234753226475758162945483637688820655414901434412397150671874315846962143570344589541065718383590511401556113514132799277828964213564555560718787737815171718267
// These should be equal
> s, sb;
102879706073536180528263644352655047823138535468359383928839403687674493489452804457542064263476094285913421420470434676795769181603593714416893723702856712 48
```

```
288508545279575849462466691879237191720840382168806695505133389866008470872676540
8
102879706073536180528263644352655047823138535468359383928839403687674493489452804
4457542064263476094285913421420470434676795769181603593714416893723702856712482885
08545279575849462466691879237191720840382168806695505133389866008470872676540
8
```

2 marks for correct determination of e, d and m.
2 marks for correct calculation of blinding process.

c. Assume that Alice has chosen a large RSA modulus n such that factorization is impossible with reasonable time and resources. She also then chooses a large random public exponent $e < n$ for which the RSA problem is also not practical. However Bob decides to send a message to Alice by representing each alphabet character as an integer modulo 26 and then encrypting each number separately using Alice's public address n, e. Is this a secure method? If not describe the most efficient attack against this method. Also, suggest a countermeasure to this attack.

3 marks = 1.5 marks for the quality of the proposed explanation.
1.5 mark for explaining with relevant examples and proposed alternative.

We can mount a known-plaintext attack against this cryptosystem. Compute $C_i \equiv (i)^e \bmod n$ for $0 \leq i < 26$. Now decrypt a ciphertext $\mathrm{CT} = (Z_1, \ldots, Z_l)$ by computing $D(Z_j) = i : Z_j = C_i, 0 \leq i < 26$ for each $j = 1, \ldots, l$.

2. (4 points) A variant of ElGamal cryptosystem over the prime field $GF(q)$ given as follows. Assume the parameters as given in the ElGamal.pdf. Let $y_A = a^{x_A} \mod q$, be the public address of Alice, where $x_A, 1 < x_A < q - 1$, is Alice's private key. Encryption function is defined as follows:

$$E(M) = C_1, C_2,$$

where $C_1 = a^k \bmod q$, where k is a random integer $1 \leq k \leq q-1$, $C_2 = K \oplus M$, where $K = y_A^k \bmod q$ and $\oplus$ is binary exclusive or function applied to binary representation of K and M.

a. Describe the Decryption Function $D(C_1, C_2)$ that Alice can use to recover the message.

b. Show how the security of the encryption function is based on Computational Diffie-Hellman (CDH) problem.

CDH Problem: Let q be a prime number and a be a generator of the cyclic multiplicative group of modulo q. Given a^x, a^y, the CDH problem computes a^{xy}.

For the given encryption function, the decryption function D(C1;C2) that recovers the message is defiened as follows:
Key K = C1^XA 1 modq and
Plaintext M = (C2 XOR K) mod q

For recovering the plaintext, the operation done is the XOR of the ciphertext C2 and the recovered key K modulo q. This is because, the function used here for obtaining the cipher text C2 is XOR and we know that for XOR operation, if c = a XOR b, we have b = a XOR c

2 marks for correct decryption function.
2 marks for explaining how CDH is used.

Suppose that there exists a probabilistic polynomial time attacker $\mathcal{A}$ that breaks the CDH problem with probability ϵ. Then attacker $\mathcal{B}$ breaks the cryptosystem as follows.

(a) $\mathcal{A}$ receives ciphertext (C_1, C_2) as input.
(b) Run $\mathcal{A}(1^\lambda, \langle a \rangle, C_1, y_A) \to y_A^k$.
(c) Output $M = C_2 \oplus y_A^k$.

$\mathcal{B}$ runs in polynomial time and succeeds iff $\mathcal{A}$ succeeds, thus has success probability ϵ. By assumption this is non-negligible, therefore the CDH problem is a necessary condition for the security of the cryptosystem.

3. (4 points) A question on HASH functions.

a. The textbook lists seven requirements of Hash functions. Out of these, one-way property, second image resistance and collision resistance are the three key requirements. Describe these three requirements.

Preimage, second image and collision resitance-elaborate.

1.5 marks = 3 * 0.5 marks.

Let X be the message space. For all probabilistic polynomial-time attackers $\mathcal{A}$,

Preimage resistance $\Pr_{x \leftarrow X}[\mathcal{A}(h(x)) = y : h(x) = h(y)]$ is neligible.

Second-image resistance $\Pr_{x \leftarrow X}[\mathcal{A}(x, h(x)) = y : x \neq y, h(x) = h(y)]$ is negligible.

Collision resistance $\Pr[\mathcal{A}(1^\lambda) = (x, y) : x \neq y, h(x) = h(y)]$ is negligible.

b. Now consider the following hash function. Here, messages are represented as series of numbers from Z_n, integers modulo n: $M = \{a_1, a_2, \cdots a_t\}$ for some integer $t \geq 1$. The hash function is defined as follows:

$$h(M) = (\sum_{i=1}^{t}(a_i)) \; mod \; n,$$

where n is a number agreed in advance.

Does the above hash function satisfy any of the three key requirements mentioned in [a].? Explain your answer.

```
None- show how the properties are violated.

1 mark for correct answer.
```

Preimage resistance Let the output of h on M be y. Then $(y, 0, \ldots, 0)$ is a valid pre-image.

Second-image resistance Same as above.

Collision resistance Same as above.

c. Now, consider a variation of the hash function for the messages represented as sequences of integers modulo n. The function is defined as follows:

$$h(M) = (\sum_{i=1}^{t}(a_i)^2) \ mod \ n,$$

where n is a large number whose factorization is unknown. Does the modified hash function satisfy any of the three key requirements mentioned in [a].? Explain your answer.

```
Satisfies preimage resitance and not others.
1.5 marks = 3 * 0.5 marks.
```

Preimage resistance It is difficult to invert h assuming the hardness of computing square roots modulo a composite.

Second-image resistance Suppose we are given $M = (a_1, \ldots, a_t)$ and $y = h(M)$. Then $M' = (n - a_1, a_2, \ldots, a_t)$ satisfies $M \neq M', h(M') = y$.

Collision resistance Same as above.

4. (4 points) A question MAC and signatures.

a. What is the main difference between message authentication codes and digital signatures?

b. Explain how Diffie-Hellman(DH) key agreement protocol is vulnerable to man-in-the-middle attack. Is it possible to secure DH key agreement protocol against this attack by using each of the following primitives? If your answer is yes, sketch the method. If the answer is no, give reasons.

i. Message Authentication Codes

ii. Public Key Digital Signatures.

iii. Hash functions.

```
1 mark part a).
```

1 mark for each of parts i) ii) and iii)

Mac is an authenticator that is a cryptographic function of both the data to be authenticated and a secret key.

Digital signature of a message is a digital tag created by the originator, the verification of the tag assures that the message was indeed created by the originator.

MAC	Signatures
is a symmetric key primitive	Generally is a public key primitive
Provides only authentication to Parties sharing the key	
	Provides authentication and non repudiation property

b)

MAC:
Since MAC is a keyed hash function, we can thwart MITM attack in DH protocol. Both Alice and Bob use a priorly agreed secret key to calculate a message digest, which can be used for authenticating each other. Since this key is assumed to be known to no one else other than Alice/Bob, the attacker Darth cant intercept the message and play with it.
NB: The emphasis for this question is on the primitive. The need for a key exchange prior is assumed. Answers which have taken both sides into consideration have been awarded marks.

Digital Sig:
If they use digital signatures, both Alice and Bob can user their private keys to sign their public keys Y s before sending them to each other. Since Darth does'nt have access to the private keys, he wont be able to intercept the message.

Hash functions:
Hash functions cannot be used to secure DH protocol from MITM attack because, a hash function, by itself, cannot provide user authentication.

5. (4 points) A question about standard ElGamal signature considered in lectures.

   (a) What are the consequence of using same k for signing two different messages

`secret key can be recovered-show the steps`
`2 marks for correct calculations.`

Let the messages be $m \neq m'$. Then the equations

$$\sigma_1 = (r, s) : r = \alpha^k \bmod q, s = k^{-1}(m - x_A r) \bmod (q-1)$$
$$\sigma_2 = (r, t) : r = \alpha^k \bmod q, t = k^{-1}(m' - x_A r) \bmod (q-1)$$

imply

$$k^{-1}(m' - m) \equiv (s - t) \bmod (q-1)$$

Let $g = \gcd(m'-m, q-1)$. Let $e = (m'-m)/g, f = (s-t)/g, v = (q-1)/g$. Then $k^{-1}e \equiv f \bmod v$ and $\gcd(e, v) = 1$. So there exists x and y with $ex + vy = 1$. Then x is the multiplicative inverse of e modulo v, hence $k^{-1} \equiv fx \bmod v$, i.e. k is recovered as $k = (xf)^{-1}$.

(b) ElGamal signature algorithm given in the lectures involves computing the inverse of a number mod $(q-1)$, where q is a prime number. Can you modify the signature algorithm avoiding the inverse computation while signing? Please explain your answer with a possible modification.

`A rearrangement of signing equation is needed;`
`show the steps;`

`2 marks for correct signing equations.`

Let $m = H(M)$, where M is the message to be signed.
Signing:

$$S_1 = \alpha^K \bmod q$$
$$S_2 = K + mx_A \bmod (q-1)$$

Verification:

$$S_1 \cdot {y_A}^m \stackrel{?}{=} \alpha^{S_2} \bmod q$$

Other possibilities exist, for example reversing the roles of K and m.

6. (5 points) This question is about Protocols. An alternative key distribution method suggested by a network vendor is illustrated in the figure below: (Fig. 14.18 of the textbook).

Figure 1: Fig. 14.18 of the Textbook

a. Describe the scheme.

b Compare this scheme to that of the scheme discussed in lectures (Fig 14.3 of the textbook-Given below).

c. Comment on the security of the new scheme.

d. What is the advantage of this scheme? Discuss the pros and cons.

e. Give an estimate of the memory requirements of KDC and the users with respect to storing key information.

1 mark part a)

1. A sends the message includes its identity and the encrypted unique identfier, Na
2. B issues a request to the KDC for a session key to protect a logical connection to A. The message includes the identity of A and B and the encrypted nonce of A and B using their own master keys Ka and Kb respectively.

The KDC responds with a message encrypted using Kb. Thus, B is the only one who can successfully read the message, and knows that it originated at the KDC. The message contains two elements intended for B:
*The one-time session key, Ks, to be used for the session.
* The original request message; include the nonce to enable B to match this response with the appropriate request.
In addition, the message includes three items intended for A:
* The one-time session key, Ks, to be used for the session.
* An identifier of B, IDB
* Nonce, Na , to allow A to correctly match the response with the request.
These last three items are encrypted with the master key that KDC shares with A, Ka.
At this point, A and B know the session key and each other from the identifiers, also know that the information originated at the KDC. They could now begin their protected exchange.

b)
1 mark for atleast 3 differences.
c) Security
1 mark for an appropriate level of explanation.

The nonce is secured; since Na and Nb are encrypted with their master key when the requests are sent over the network assure that the original request was not altered before reception by KDC.
* The information replied from KDC is encrypted with master keys; it is

protected from eavesdropping. A knows the session key, assure that the other party is B, and also knows that the information originated at the KDC.
* Even if there is no authentication function between A and B, A still be assured that the original message it received was not a replay.

d)
1 mark.
pros and cons
Both schemes achieved the same level of security.
* One of them is more efficient, requiring only 3 steps.
* Replay attack is detected differently. In one the attack is detected in the beginning while in the other is detected at the end of the protocol.
Cons
* KDC can become a performance bottleneck.
* It the KDC is compromised, all communications are insecure.
* KDC can impersonate anyone.
* KDC is a single point of failure when it is not available
e)
1 mark.
Memory Requirements
2 marks for an appropriate level of explanation.
Suppose there are N entities that want to communicate in pairs. The keys that need to be stored in KDC are N master keys plus [N(N-1)]/2 session keys. Hence, the memory requirement of KDC is N+[(N(N-1))/2] For each entity, the keys that need to be stored in each of them are 1 master key plus N-1 session keys. Thus, memory requirement of each entity is N.