TTM4135 Worksheet 5: RSA

Tjerand Silde and Emil August Hovd Olaisen
{tjerand.silde, emil.august.olaisen}@ntnu.no

Spring 2026

1. Review the definitions of the following concepts:
   (a) trapdoor one-way function;
   (b) RSA equations;
   (c) prime number theorem;
   (d) square-and-multiply algorithm;
   (e) Håstad's attack;
   (f) Miller's theorem;
   (g) discrete logarithm problem;
   (h) generator of $\mathbb{Z}_p^*$;
   (i) Diffie–Hellman key exchange.

2. Suppose that an RSA public key is chosen with primes $p = 13$ and $q = 17$. Suppose that the public key is $e = 5$.
   (a) Find the value of d.
   (b) Find the encryption of $m_1 = 4$ and $m_2 = 13$.
   (c) Decrypt the ciphertexts and verify that the correct value is recovered.

3. Suppose that RSA encryption uses a modulus n of 3000 bits. Assuming that the square-and-multiply method is used for exponentiation, compare the computational cost of encryption, measured in the number of squarings and the number of multiplications, in the following cases:
   (a) $e = 3$
   (b) $e = 2^{16} + 1$
   (c) e is chosen randomly between 0 and n.

   How much computation is required for decryption in each case?

4. Suppose that the same message m has been encrypted for three recipients with different RSA moduli: 205, 319 and 391. Each recipient uses public exponent $e = 3$. Suppose also that no random padding has been added. The three ciphertexts found are: 180, 43 and 218 respectively.

   Demonstrate Håstad's CRT attack by finding the value of m without making use of the factorization of the moduli.

5. Consider RSA with values $p = 23$, $q = 31$, $n = 713$ and $d = 233$. Suppose the received ciphertext is $c = 266$.

   Examine the faster decryption method using the CRT, using these values:
   (a) Compute $m_p = c^{d \bmod p-1} \bmod p$.
   (b) Similarly compute m_q.
   (c) Combine these results using the CRT to decrypt the ciphertext.

6. Suppose that an attacker obtains an RSA private key $d = 233$ and also has the public key $e = 17$ and $n = 713$. Apply Miller's algorithm to factorize n.

7. In this question we show that $f(x) = x^2 \bmod n$ is a trapdoor one-way function, when $n = pq$ and $p \bmod 4 = q \bmod 4 = 3$ and p and q are different primes. We do this in three steps.

   (a) Explain why we know that $(p+1)/4$ is an integer, and explain why we know that if x has a square root in $\mathbb{Z}_p^*$ then show that $x^{(p+1)/4} \bmod p$ is a square root of x in $\mathbb{Z}_p^*$.

   (b) Use the part above to show that if p and q are known, then a square root modulo n can be efficiently computed (assume we have an efficient exponentiation function). Thus p and q are a trapdoor to invert f.

   (c) Now suppose that there exists an algorithm A that finds integers x, y, z such that $x \equiv y^2 \equiv z^2 \bmod n$ with $z \not\equiv y \bmod n$ and $z \not\equiv -y \bmod n$, then this can be used to factorize n. Hence deduce that inverting f is as hard as factorizing n, and that f is one-way.

8. It is common in the ElGamal encryption algorithm for users to share the modulus p and generator g. Why is it not possible for users to share the same modulus n in the RSA cryptosystem?

9. The Diffie–Hellman protocol can be defined, but is not necessarily secure, in any group.

   (a) Define Diffie–Hellman in the *additive* group modulo p for some prime p, instead of the multiplicative group where it is usually defined. Would this be secure for sufficiently large values of p?

   (b) Write down the equations for Diffie-Hellman on elliptic curves (ECDH) using the notation from the lecture slides. Show that for this to be secure the elliptic curve discrete log problem must not be an easy problem.

10. In the ElGamal cryptosystem Alice and Bob have public keys g^{x_A} and g^{x_B} respectively, with corresponding private keys x_A and x_B. When Alice wants to send a message confidentially to Bob she chooses an ephemeral private key a and constructs a new shared secret g^{ax_B}.

    Consider the following variant of the ElGamal cryptosystem. Instead of choosing a new random value a, Alice simply computes the static Diffie–Hellman value $X = (y_B)^{x_A} \bmod p$ and sends $c = mX \bmod p$ to Bob as the ciphertext.

    (a) How does Bob decrypt?

    (b) What could be the advantages of such a scheme as compared with normal ElGamal encryption?

    (c) Show that this scheme is, unfortunately, completely insecure against a known plaintext attack.