Cryptography, exercise sheet 6 for 12 Oct 2021

We will work through these exercises on wonder.me; you can find the URL for the room in the Canvas announcement and in Zulip chat under the topic "Wonder session" in the "general" stream. These are exercises to challenge your understanding of the lectures you have watched already, in particular lectures I – VI about RSA.
These are not for homework.

You can use Sage or other computer-algebra systems for the computations in exercise 3 onwards.

You can call one of us over by choosing "invite to circle". Please note, though, that if we are in another circle busy talking we will not come right away and invitations expire quickly. So pick one of us who is in the TA corner, thus not occupied.

1. If you used GitKraken to generate an RSA key pair for GitHub pull the fixed software and make a new key. A problem with entropy in keys was announced yesterday.
   This is probably more urgent to do than the other exercises.

   For details on what happened see the security advisory. We will cover RSA with bad randomness next, and this is yet another great real-world example of what can go wrong with RSA.

2. This exercise is to be solved by hand, do not use a computer algebra system.
   Compute $15^{24} \bmod 72$ using the Chinese Remainder Theorem with calculations modulo 8 and modulo 9. Remember to reduce the exponents and the base in the CRT calculation and take a moment to think what moduli to use and to check the conditions.
   In case this is not obvious $72 = 2^3 \cdot 3^2$, so this is not an RSA number and you need to use (and understand) the Euler-phi function and when you can reduce exponents.
   See RSA-II for how CRT is used for RSA moduli.

3. Show how to retrieve the message m in RSA-OAEP from $M = (s, t)$. (See RSA I for the definition of RSA-OAEP.) This is just considering the encoding and decoding of the message and skips the RSA part. The functions G and H are cryptographic hash functions, so you cannot invert them.

4. Write the integer $a = 2342$ in binary and compute the right-to-left sliding window representation for it using window size $w = 3$, i.e., where windows have at most 3 bits. Write the representation with the most-significant bit on the left.

5. Perform one round of the Fermat test with base $a = 2$ to test whether 31 is prime.

6. Perform one round of the Miller-Rabin test with base $a = 2$ to test whether 157 is prime.
   What is the answer of the Miller-Rabin test?

7. Use the Pocklington test to prove that 157 is prime. You may use that 13 is prime.

8. Use Pollard's rho method for factorization to find a factor of 27887. Use starting point $\rho_0 = 17$, iteration function $\rho_{i+1} = \rho_i^2 + 1$ and Floyd's cycle finding method, i.e. compute $\gcd(\prod_{i=1}^{z}(\rho_{2i} - \rho_i), 27887)$ until a non-trivial gcd is found. Deviating from RSA-IV do the gcd computations after each i, so skip the product over z. Document the intermediate steps in a table, with one row for rho_i, one for ρ_{2i}, and one for their gcd.

9. Use the $p-1$ method to factor 27887 with basis $a = 2$ and exponent $s = \text{lcm}(1, 2, 3, 4, 5, \ldots, 11)$.
   Explain why the method worked.

10. Use Dixon's factorization method to factor the number $n = 403$ using $a_1 = 22$.
    Note: This lists all the a_i you need.