VE475
Introduction to Cryptography

Homework 5

Manuel — UM-JI (Summer 2019)

Non-programming exercises:

- Write in a neat and legible handwriting, or use LaTeX
- Clearly explain the reasoning process
- Write in a complete style (subject, verb and object)

Progamming exercises:

- Write a README file for each program
- Upload an archive with all the programs onto Canvas

Ex. 1 — *RSA setup*

Most RSA setups require the message m to be coprime to the RSA modulus n. In the exercise we will prove that m can still be decrypted if $gcd(m, n)$ is not 1.

1. Why is it likely for n to be coprime with m?
2. Let k be a multiple of $\varphi(n)$.
   a) Show that if $\gcd(m, n) = 1$, then $m^k \equiv 1 \bmod p$ and mod q.
   b) Prove that for any arbitrary m, $m^{k+1} \equiv m \bmod p$ and mod q.
3. Let e and d the RSA encryption and decryption exponent, respectively.
   a) Show that $m^{ed} \equiv m \bmod n$ for all m.
   b) Conclude on the necessity of having $\gcd(m, n) = 1$.

Ex. 2 — *RSA decryption*

The ciphertext 5859 was obtained using RSA encryption with $n = 11413$ and $e = 7467$. Recover the plaintext.

Ex. 3 — *Breaking RSA*

Wiener's attack allows to recover the decryption exponent under the condition that it is small enough.

1. Why would one select short encryption or decryption keys?
2. Search and explain how Wiener's attack is working.
3. How to ensure not generate a weak decryption key?
4. Given $n = 317940011$ and $e = 77537081$, apply Wiener's attack in order to factor n. Either provide the source code of your program or clearly detail all the steps.

Ex. 4 — *Programming*

Implement the three functions `generate`, `encrypt` and `decrypt`, which generate the RSA parameters, encrypt, and decrypt, respectively.

The function `generate` takes as input a security level and generate p and q such that n is long enough to match the required security level. No special requirement is requested on `encrypt` and `decrypt`.

Common security levels:

Security level (bits)	80	112	128	192	256
RSA modulus (bits)	1024	2048	3072	7680	15360

Ex. 5 — *Simple questions*

Let n, e, d, p, q be the usual RSA parameters.

1. A message m is encrypted into the ciphertext c. Explain how to run a CCA attack on "texbook RSA".
2. Instead of using a single exponent one wants to encrypt twice using a single n but two different exponents. Is this double encryption adding any security? Explain your answer.
3. Let $n = 642401$. Knowing that $516107^2 \equiv 7 \bmod n$ and $187722^2 \equiv 4 \cdot 7 \bmod n$ factorize n.
4. Describe how an RSA scheme would work if instead of the two primes p and q, three primes p, q, and r were used. Explain the drawback of such a setup.
5. Determine the smallest generator of $\mathrm{U}(\mathbb{Z}/97\mathbb{Z})$.
6. Consider the multiplicative group $G = \mathrm{U}(\mathbb{Z}/101\mathbb{Z})$.
   a) Prove that 2 is a generator of G.
   b) In G, determine $\log_2 24$, knowing that $\log_2 3 = 69$.
   c) In G, determine $\log_2 24$, knowing that $\log_2 5 = 24$.
7. Knowing that $3^6 \equiv 44 \bmod 137$, and $3^{10} \equiv 2 \bmod 137$, find $0 \leq x \leq 135$ such that $3^x \equiv 11 \bmod 137$.
8. Let $G = \mathrm{U}(\mathbb{Z}/11\mathbb{Z})$
   a) Compute 6^5 in G.
   b) Prove that 2 is a generator of G
   c) Let x be such that $2^x \equiv 6 \bmod 11$. Without calculating it, decide whether x is even or odd.

Ex. 6 — *DLP*

In this exercise we want to determine x such that $3^x \equiv 2 \bmod 65537$.

1. Prove that 2048 divides x, while 4096 does not.
2. How many possible choices need to be considered for x? Determine x.
3. Can the Pohlig-Hellman algorithm be applied to this example? If so show the details.
4. Explain why such primes are not fitting a cryptographic context.

Note: in homework 4 exercise 2 it was proved that 3 is a generator of $\mathrm{U}(\mathbb{Z}/65537\mathbb{Z})$.