Security

Assignment 12, Wednesday, December 9, 2015

Handing in your answers: the full story, see

`http://www.sos.cs.ru.nl/applications/courses/security2015/exercises.html`

Briefly,

- submission via Blackboard (`http://blackboard.ru.nl`);
- one single pdf file;
- make sure to write all names and student numbers and the name of your teaching assistant (Brinda or Joost).

Deadline: Thursday, December 17, 24:00 (midnight) sharp!

Goals: This assignment repeats and reinforces concepts and techniques from this course. After completing these exercises successfully, you should

- have seen the risks of shared RSA moduli;
- be able to perform large RSA computations using calculators.

Marks: You can score a total of 100 points.

1. **(20 points)** We assume that Alice and Bob use RSA public keys with the same modulus n but with different public exponents e_A and e_B. We further assume that Alice and Bob still have their p and q such that $p \cdot q = n$.

   (a) Show that Alice can decrypt messages sent to Bob. (*Hint:* How can Alice calculate d_B?)

   (b) Assume that $\gcd(e_A, e_B) = 1$. This implies that Eve can apply Extended Euclidean gcd algorithm to find integers x and y such that $x \cdot e_A + y \cdot e_B = \gcd(e_A, e_B) = 1$. Now if the message m was sent encrypted as c_A to Alice and as c_B to Bob, how can Eve obtain this message m from $c_A^x \pmod{n}$ and $c_B^y \pmod{n}$? Show the steps clearly.

2. **(30 points)** Consider the RSA cryptosystem[1]. Suppose $p = 17, q = 43$ and $e = 37$.

   (a) Determine the corresponding d (using the Extended Euclidean Algorithm).

   (b) We are going to encrypt the message "banana" with the public key in Electronic Code Book[2] (ECB) mode, that is block-by-block.

      i. Take blocks of length 1 letter, so we have 6 blocks: "b", "a", ..., "a".

      ii. Translate each block into a number: "a" $\mapsto$ 1, "b" $\mapsto$ 2, ..., "A" $\mapsto$ 27, "B" $\mapsto$ 28, ...

      iii. Complete Table 1 with these mappings from letter-blocks to integer-blocks.

   (c) Encrypt each integer-block with the public key. Give intermediate steps and fill out the second row of Table 1.

   (d) Decrypt it with the private key filling out the third row of Table 1. Again, give the intermediate steps as well.

[1] `http://en.wikipedia.org/wiki/RSA`

[2] `https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation`

	b	a	n	a	n	a
Mapping (Step b)	.	1	.	.	.	.
Ciphertext (Step c)	.	.	.	.	.	.
Recovered plaintext (Step d)	.	.	.	.	.	.

Table 1: Computation steps RSA

3. **(50 points)** Unlike in the previous problem, text is usually encoded in ASCII-encoding[3] instead. More importantly, asymmetric cryptography is not used to encrypt the actual text in the message directly; instead, it is used to encrypt a symmetric key which is in turn used to encrypt the actual text. Recall that we saw a protocol that did this in last week's assignment.

   In this assignment, we will look at a more realistic scenario. Imagine we have intercepted a session key that was encrypted with the following RSA public key (n, e):

$$\begin{aligned} \text{n} &= 90214098372175031699946524430948980 49733 \\ \text{e} &= 65537 \end{aligned}$$

   The RSA-encrypted session key is:

```
1A 5D E7 D5 AD 02 D6 7C E7 5E 65 60 3C E0 3F 4B 55
```

   The session key is a 128-bit AES key. AES[4] is a standard block cipher - for this course, that is all you need to know about it. AES was used in CBC[5] mode, to produce the following ciphertext:

```
B4 F5 55 60 68 CE 8D 5C E1 36 9C 6E 69 4F 20 20
ED 9E C2 18 A7 CC 04 CF 92 71 FC F5 FF F5 E7 5A
1D 75 74 C9 CB AB 2F B8 D1 80 4F EE FF 30 5D 9F
```

   The following IV was used:

```
55 BB 82 09 2A 18 AA A9 EF 68 0A 6C 2C 94 8F 00
```

   The plaintext is a string encoded using ASCII-encoding. Find the plaintext.

   To help you do this, we have set up the following web pages that enables you to do AES encryption/decryption, as well as XOR operations on large values:

   - http://www.sos.cs.ru.nl/applications/courses/security2015/aes/
   - http://www.sos.cs.ru.nl/applications/courses/security2015/xor/

   Furthermore, it will be helpful to use tools like Wolfram Alpha[6] to help you with the computations, or self-written code. Make sure that whatever you use is able to support large integers.

   First, write down your strategy in steps, using terms such as RSA, φ, d, CBC, IV, block, XOR, decrypt, encrypt, decode, *etc.* Be as specific as possible. Second, perform each step and make the computation explicit: what tool did you use, and how? What was your input, and what was the result? **Write down your intermediate steps and results!** This will also make it easier to stay organized while solving this exercise.

---

[3] https://en.wikipedia.org/wiki/ASCII#ASCII_printable_characters
[4] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[5] https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation
[6] http://www.wolframalpha.com/