Security

Assignment 13, Friday, December 16, 2016

Handing in your answers: For the full story, see

`http://www.sos.cs.ru.nl/applications/courses/security2016/exercises.html`

To summarize:

- Include your name and student number **in** the document (they will be printed!), as well as the name of your teaching assistant (Hans or Joost). When working together, include **both** your names and student numbers.
- Submit one single **pdf** file – when working together, only hand in **once**.
- Hand in via Blackboard, before the deadline.

Deadline: Monday, January 9, 09:00 sharp!

Goals: After completing these exercises successfully you should be able to

- perform computations of a Diffie–Hellman key exchange
- recognize the shortcomings of the Diffie–Hellman key exchange;
- perform computations for ElGamal encryption/decryption/signatures;
- understand the dangers involved in reusing randomness.

Marks: You can score a total of 100 points.

1. **(25 points)** The Diffie–Hellman (DH) key exchange is used to agree on a secret key between Alice and Bob. The prime $p = 1021$ determines the group $\mathbb{Z}_p^* = \{1, \ldots, p-1\}$ in which all operations are performed (i.e. all computations are performed modulo 1021).
The following messages are exchanged:

$$\begin{array}{lll} 1. \quad A \longrightarrow B & : & p = 1021, g = 10, g^a = 93 \\ 2. \quad B \longrightarrow A & : & g^b = 491 \end{array}$$

   (a) Given Alice's secret $a = 317$, compute the shared secret key. Show how you came to the solution.

   (b) Since the modulus is very small, one can compute the secret values. Derive Bob's secret from the exchanged messages. Feel free to use a calculator[1] or you can write a small progam. In any case, explain your steps.

   (c) Check that Bob has the same (shared) key as Alice using the private key from (b) (by doing the DH-computation for Bob's side).

   (d) We describe a modified communication when there is a middle-man. Assume that message 1. $A \rightarrow B$ is as showed above, but Eve captures the message and picks two random values: $r_A = 37, r_B = 404$. She uses these random values for the communication with Alice and Bob, respectively.

      i. Show the *four messages*: $A \rightarrow E(B), E(A) \rightarrow B, B \rightarrow E(A), E(B) \rightarrow A$. Use the protocol notation as used earlier in this course.

      ii. Compute the *established keys* K_{AE}, K_{BE} between Alice and Eve, and between Eve and Bob, respectively.

[1] e.g. `https://www.wolframalpha.com`

2. **(30 points)** Consider the ElGamal public-key encryption system. For $p = 31$, $G = \mathbb{Z}_p^*$ is a multiplicative cyclic group with generator $g = 3$. Suppose that the secret number in the system is $a = 17$. You will encrypt messages and decrypt ciphertexts in this group. Describe your computations.

   (a) Determine the corresponding value $A = g^a \in G$.

   (b) We are going to encrypt the message "remember" (in ECB mode) using ElGamal. To map letters to integers we use the mapping $a \mapsto 1$, $b \mapsto 2, \ldots, z \mapsto 26$. For the following steps, fill in each row in the table below, and explain the required computations:

      i. For each integer block, calculate a separate ephemeral public key A^r using the following values for r: 3, 6, 9, 12, 15, 18, 21 and 24.

      ii. For each integer block, calculate the first component $c_1 = R = g^r$ of the ciphertext using that same sequence for r.

      iii. Finally, for each integer block, calculate the second component $c_2 = m \cdot A^r$ of the ciphertext.

   (c) Let's now decrypt the ciphertext; complete the table below

      i. For each integer block calculate the inverse of the ephemeral public key $(A^r)^{-1} = {c_1}^{-a}$. (*Note*: ${c_1}^{-a}$ can be calculated as ${c_1}^{p-1-a}$, using Euler's Theorem and the fact that $\phi(p) = p - 1$).

      ii. For each integer block, use the inverse $(A^r)^{-1}$ to cancel out A^r in c_2 and thus retrieve $m = c_2 \cdot A^{-r}$.

	r	**e**	**m**	**e**	**m**	**b**	**e**	**r**
Encryption								
Mapping	18	5	.	.	.	.	.	.
r	3	6	9	12	15	18	21	24
A^r	.	.	.	.	.	.	.	.
$c_1 = g^r$	.	.	.	.	.	.	.	.
$c_2 = m \cdot A^r$	.	.	.	.	.	.	.	.
Decryption of ciphertext (c_1, c_2)								
$(A^r)^{-1} = {c_1}^{-a}$	.	.	.	.	.	.	.	.
$m = c_2 \cdot A^{-r}$	.	.	.	.	.	.	.	.

3. **(30 points)** The ElGamal signature scheme.

   Suppose $G = \mathbb{Z}_p^*$ for $p = 29$, with generator $g = 3$. For the order of G, we write $\#g = \phi(p)$. In this exercise we will use the (otherwise completely insecure) hash function $h(m) = m$. Let's assume that Alice's secret key is $a = 21$. Please make sure to use the correct modulus for each step.

   (a) Determine Alice's corresponding public key A.

   (b) Sign the message $m = 15$ using ElGamal signatures with random value $r = 5$.

      i. Verify that r and $\#g$ are relatively prime.

      ii. Compute $s_1 = R = g^r \bmod p$.

      iii. Compute $r^{-1} \bmod \#g$.

      iv. Compute $s_2 = (h(m) - a \cdot R) \cdot r^{-1} \bmod \#g$

   (c) Verify that the signature (s_1, s_2) is correct on message m using Alice's public key A.

      i. Check that $1 \leq s_1 < p$.

      ii. Compute $v := s_1^{s_2} \cdot A^{s_1} \bmod p$.

iii. Verify $g^{h(m)} \stackrel{?}{=} v$.

4. **(15 points)** Predictable randomness.

When using the ElGamal scheme, it is crucial that one uses a fresh random number r for each use. However, true random numbers are not that easy to obtain - in practice, they are typically generated *pseudo*-randomly, and sometimes this is done poorly. When this is done in an insecure fashion, an attacker could influence the randomness, cause a system to use the same 'random' value twice or even predict the randomness completely.

(a) Consider ElGamal encryption (let $G = \mathbb{Z}_p$ for some prime p). What can an attacker learn if the randomness r is known, and he intercepts an ElGamal ciphertext? Show how!

(b) Now consider ElGamal signatures. Show what an attacker can learn when the randomness r is known, and he obtains an ElGamal signature (s_1, s_2) (with the corresponding message m). Again, show how!

(c) Which of these scenarios has more devastating consequences? For example, consider the security of other ciphertexts and signatures for which the used randomness r' is still unknown.