7. Exercise sheet
Hand in solutions until
Monday, 15 December 2008, 11:59 am (deadline!).

As usual: Any claim needs a proof or an argument.

Exercise 7.1 (ElGamal signatures). (7 points)

Compute an ElGamal signature for your student identification number represented in binary. Use $p = 467$ and $g = 3 \in \mathbb{Z}_p^*$ and work in $G = \langle g \rangle$. For simplicity, we take the function $\text{HASH}: \{0, 1\}^* \rightarrow \mathbb{Z}_{233}^*$, $x \mapsto (\sum_{0 \leq i < |x|} x_i 2^i) \mod 233$. (Eg. 18 translates to the string $10010$ which in turn translates into the number $18 \mod 233$.)

(i) Here $\#G = 233$ and thus $\exp_g: \mathbb{Z}_{233} \rightarrow G, \ a \mapsto g^a$ is an isomorphism.
[Note that $166^2 = 3$ and thus $g^{233} = 1$. Since $g \neq 1$...]

(ii) Setup: Compute Alice’ public key with $\alpha = 9$.

(iii) Sign: Sign the hash value of your student identification number.

(iv) Verify: Verify the signature.

Exercise 7.2 (Attacks on the ElGamal signature scheme). (4 points)

After prior failures princess Jasmin and Genie have been doing a lot of thinking and research. Genie has proposed to use the ElGamal signature scheme. They have chosen the prime number $p = 1334537$ and the generator $g = 16$.

The public key of the princess Jasmin is $a = 605828$.

(i) They have sent the message $(x, b, \gamma) = (7654, 642260, 4427)$. Unfortunately, Genie was not very careful. He wrote down the number $\beta$ somewhere and forgot to burn the piece of paper after calculating the signature. Now Jaffar knows the number $\beta = 377$. Compute the secret key $\alpha$. 

2
(ii) Princess Jasmin has changed her secret key. She now has the public key $a = 436,700$. This time Jaffar could not find the number $\beta$. Because of this he used an enchantment so that Jasmin’s random number generator has output the same value for $\beta$ twice in a row. This was the case for the messages $(2,008, 14,694, 21,273)$ and $(234, 14,694, 10,507)$. Now compute Jasmin’s secret key $\alpha$.

Exercise 7.3 (Hash crisis). (11+3 points)


(i) What is the purpose of X.509 certificates?
(ii) Where are they used?
(iii) How does such a certificate ensure a connection between a secret key and identification information (name, birth, and so on) of a person?
(iv) Who verifies this connection?
(v) How can I check that this verification was done (assuming the verification authority is honest)? In other words, how can I check the certificate?
(vi) What is the consequence of Lenstra’s observation?
(vii) Add further observations.

Exercise 7.4 (Security estimate). (8 points)

The ElGamal signature scheme works over some publicly known group of (often prime) order $\ell$, where $\ell$ has length $n$. In many cases this is a subgroup of some $\mathbb{Z}_p^\times$ with another (larger) prime $p$; then $\ell|\left(p - 1\right)$. However, it is necessary for its security that it is difficult to compute a discrete logarithm in the group and also, if applicable, in the surrounding group $\mathbb{Z}_p^\times$. The best known discrete logarithm algorithms achieve the following (heuristic, expected) running times:

<table>
<thead>
<tr>
<th>method</th>
<th>year</th>
<th>time for a group size of $n$-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>brute force (any group)</td>
<td>$-\infty$</td>
<td>$O^\sim\left(2^n\right)$</td>
</tr>
<tr>
<td>Baby-step Giant-step (any group)</td>
<td>1971</td>
<td>$O^\sim\left(2^{n/2}\right)$</td>
</tr>
<tr>
<td>Pollard’s $g$ method (any group)</td>
<td>1978</td>
<td>$O\left(n^22^{n/2}\right)$</td>
</tr>
<tr>
<td>Pohlig-Hellman (any group)</td>
<td>1978</td>
<td>$O^\sim\left(2^{n/2}\right)$</td>
</tr>
<tr>
<td>Index-Calculus for $\mathbb{Z}_p^\times$</td>
<td>1986</td>
<td>$2^{(\sqrt{2}+o(1))n^{1/2}\log_2^{1/2}n}$</td>
</tr>
<tr>
<td>Number-field sieve for $\mathbb{Z}_p^\times$</td>
<td>1990(?)</td>
<td>$2^{((64/9)^{1/3}+o(1))n^{1/3}\log_2^{2/3}n}$</td>
</tr>
</tbody>
</table>
It is not correct to think of \( o(1) \) as zero, but for the following rough estimates just do it. Estimate the time that would be needed to find a discrete logarithm in a group whose order has \( n \)-bits assuming the (strongest of the) above estimates are accurate with \( o(1) = 0 \) (which is wrong in practice!)

(i) for \( n = 1024 \) (standard size),
(ii) for \( n = 2048 \) (as required for Document Signer CA),
(iii) for \( n = 3072 \) (as required for Country Signing CA).

Repeat the estimate assuming that for the given group only Pollard’s \( \rho \) method is available, for example in case the group is a \( \ell \)-element subgroup of \( \mathbb{Z}_p^\times \) or an elliptic curve,

(iv) for \( n = 160 \),
(v) for \( n = 200 \),
(vi) for \( n = 240 \).

In April 2001 Reynald Lercier reported (http://perso.univ-rennes1.fr/reynald.lercier/file/nmbrJL01a.html) that they can solve a discrete logarithm problem modulo a 397-bit prime \( p \) within 10 weeks on a 525MHz computer.

(vii) Which bit size for the prime \( p \) is necessary to ensure that they cannot solve the DLP problem in \( \mathbb{Z}_p^\times \) given —say— 10’000 10GHz computers and 1 year (disregarding memory requirements).

[Note: The record for computing discrete logs in \( \mathbb{F}_p^\times \) lies at \( n = 613 \), see Antoine Joux http://perso.univ-rennes1.fr/reynald.lercier/file/nmbrJL05a.html.]