Wrocław University of Technology, WPPT na 12.11.13

CRYPTOGRAPHY, 2013 Assignments, list # 5

1. Is it possible to show that an ElGamal ciphertext of a random key has been prepared with the public key of Alice?
2. Prove that it is possible to *re-encrypt* a ciphertext created with the public key of Alice. That is, one can transform a ciphertext c into another ciphertext of the same plaintext M without using the private key of Alice. Find a method such that the probability distribution of the resulting ciphertext is uniformly discributed in the of all ciphertexts of M.

   More difficult part: change the Elgamal encryption so that re-encryption is possible even without the public key originally used for encryption.
3. Let n be an RSA number. Is the set of elements coprime with n^2 with multiplication $\mod n$ a group? Take an $a < n^2$ that is coprime with n. Determine possible values for the order of n. Use CRT and the fact that $\mathbb{Z}_{p^i} \setminus \{0\}$ with multiplication $\mod p^i$ is a cyclic group.

   Prove that the order of $n+1$ modulo n^2 is n.

   For the Paillier encryption scheme, are the ciphertexts uniformly distributed in the set $\{1, 2, ..., n^2 - 1\}$? If not, what is the the distribution?
4. One of the ideas to bypass security of ElGamal signatures (and similar ones) is to compute random parameters in so called "kleptographic way". The solution is to

   - store $U = g^u$ inside a infected device (but not u, element u must be kept secret by the attacker)
   - instead of choosing parameter k at random during signature creation, execute the following procedure
     (a) restore k' from the previous signature generated by the device,
     (b) $k := H(U^{k'})$

   Show how the attacker may derive k and consequently signing key x using the previous signature (r', s'). Hint: compute $(r')^u$... Is DSA secure against such kind of attacks?
5. Consider a signing device D such that after receiving the message m to be signed, D performs the following steps:

   (a) choose k and $r := g^k$
   (b) compute $H(M)$
   (c) compute signature component s according to ElGamal scheme (or DSA, Schnorr, ...).

   We assume that we can "rewind" D to exactly the same state as occurs after step (a) and replace the module for computing H by another one. How to derive the signing key in this scenario? Use such D to compute discrete logarithms of public keys. Formulate the attack in the language of *random oracle model*.
6. Show that it is necessary to use hash function for ElGamal signatures. For this purpose create a signature of an arbitrary message without the secret key.

   Hint: choose (u, v) such that v and $p-1$ are coprime. Take $r = \alpha^u y^v \mod p$ and $s = rv^{-1} \mod (p-1)$. Consider the message $m = su \mod (p-1)$.
7. Specification of the signature algorithm also determines the proper range of all elements. In order to see that it is essential consider the ElGamal signatures. Assume that we do not check that $r < p$.

   Show that in this case we could forge a signature for any m' given a signature (r, s) of m. Let $u = Hash(m) \cdot (Hash(m))^{-1} \mod (p-1)$. Put $s' = su \mod (p-1)$ and compute r' using CRT such that $r' = ru \mod p-1$ and and $r' = r \mod p$.

/-/ Mirosław Kutyłowski and Maciej Gebala