Modern Cryptology (236506) – Exercise no. 5

Submission in singles until 10/06/2019.

The exercise must be typed and printed.

1. (a) Find the modular square roots of 3 modulo 23, using the algorithm we saw in class. describe your calculations.
(b) Find the modular square roots of 20 modulo 29, using the algorithm we saw in class. describe your calculations.

2. Bob uses the RSA encryption with the public key: \((n, e) = (187, 23)\). Alice encrypted a message for Bob, and sent him the encrypted message, which was 11. What was the original message Alice sent? Explain your calculations.

3. This question deals with weaknesses of the ElGamal signature scheme.
   (a) Show that given a legal signature \((R, S)\) on a message \(m\), an attacker can compute signatures for messages of the form \(m' \equiv (m + bS)a \pmod{p - 1}\), where \(b \in \mathbb{Z}_p\) can be chosen arbitrarily and \(a \equiv g^b \pmod{p}\).
   Hint: Look at the value of \(g^{ma+baS} \pmod{p}\).
   (b) Show that the ElGamal scheme is vulnerable to existential forgery attacks. I.e., show that an attacker can produce a combination of a message \(m\) and a legal signature on it \((R, S)\), but he cannot necessarily choose the value of \(m\).
   Hint: Choose \(R\) to be of the form \(R \equiv g^{\alpha x + \beta x} \pmod{p - 1}\) for some \(\alpha, \beta \in \mathbb{Z}_p\) such that \(\gcd(\beta, p - 1) = 1\) (in practice, \(\alpha\) and \(\beta\) are chosen randomly).

4. In the following cases, a signer signs messages using the Schnorr signature scheme with a fixed secret key \(s\), where the values of \(r\) are chosen in various ways. Show how to find the secret key, if:
   (a) The signer uses the same random number twice.
   (b) The signer chooses the random numbers randomly, but publishes the difference between the random numbers of two distinct signatures.
   (c) The signer signs 3 messages, using 3 random numbers which form an arithmetic progression, that is \((r, r + d, r + 2d)\).
   (d) The signer signs 4 messages, with random numbers chosen using an unknown polynomial of degree two \(P(x)\), where the \(i\)-th message is signed using the value \(P(i)\).
5. Random self reducibility of DLOG.

Let $p$ be a prime and let $g \in \mathbb{Z}_p^*$ be a generator. Suppose that there exists a polynomial-time algorithm $A$ that given $p, g, g^x \mod p$ finds $x$ for $\frac{1}{1600}$ of the possible $x$’s. Show how to use $A$ as a subroutine to construct a probabilistic polynomial time algorithm $B$ that solves the DLOG problem for all instances (i.e., for every $x \in \mathbb{Z}_{p-1}$) with probability $\frac{1}{\sqrt{3}}$. Analyze the running time of $B$ as a function of the running time of $A$ and the length of the binary representation of $p$. 
