

On the soundness of the Schnorr Scheme

D.R. Stinson

January 30, 2007

Soundness of an identification scheme means that anyone who can impersonate Alice with non-negligible probability in polynomial time can compute Alice's private key in polynomial time. The soundness of the Schnorr Scheme is discussed briefly on page 375 of the textbook *Cryptography Theory and Practice, Third Edition*. There it is shown that if an adversary (Oscar or Olga) knows a value γ , two challenges r_1 and r_2 , and corresponding valid responses y_1 and y_2 , then the adversary can easily compute Alice's private key. The question is how the adversary can compute relevant values of γ, r_1, r_2, y_1 and y_2 . In this note we discuss some plausible assumptions under which the adversary can compute this information.

We assume the existence of two algorithms: Algorithm FINDGAMMA takes as input all the public parameters associated with the Schnorr Scheme, and in polynomial time outputs a value γ . The Algorithm FINDRESPONSE takes as input the public parameters, a value γ found by FINDGAMMA, and a challenge r . FINDRESPONSE either outputs a response y or "fail".

The idea is to run the algorithm FINDRESPONSE with many different random challenges, until correct responses are found to two different challenges. That is, we perform the following steps:

1. Call FINDGAMMA(p, q, α, t, v) and obtain a value γ (to be used in step 2).
2. Call FINDRESPONSE($p, q, \alpha, t, v, \gamma, r$) with random values r until two valid responses are obtained.

Now assume that the probability that FINDRESPONSE outputs a correct response is $1/t^c$, where c is some constant. That is, FINDRESPONSE will output a correct response for $2^t/t^c$ different challenges r (recall that there are 2^t possible challenges). Observe that we are assuming that FINDRESPONSE has a non-negligible success probability.

Now suppose we call the procedure FINDRESPONSE with t^c different random challenges (this is a polynomial number of challenges, as a function of the security parameter t). The probability of obtaining no correct responses is

$$p_0 = \left(1 - \frac{1}{t^c}\right)^{t^c}$$

and the probability of obtaining exactly one correct response is

$$p_1 = t^c \times \frac{1}{t^c} \times \left(1 - \frac{1}{t^c}\right)^{t^c-1} = \left(1 - \frac{1}{t^c}\right)^{t^c-1}.$$

It is easy to see that $p_0 \approx p_1 \approx e^{-1} \approx 0.37$. Now, the probability of obtaining at least two correct responses is

$$1 - p_0 - p_1 \approx 0.26.$$

We end up achieving a constant success probability in polynomial time, which means that the adversary has an efficient way to compute Alice's private key under the stated assumptions.

Finally, note that similar techniques should be used to establish soundness of other identification schemes such as the Okamoto and Guillou-Quisquater Schemes.

Acknowledgements

This note came about through some helpful discussions with Claude Crépeau and Andreea Panait, who pointed out that the discussion in my textbook was incomplete. They suggested modelling the attack as two separate algorithms, and I followed that suggestion in writing up this note.