How to Prove That a Committed Number Is Prime

Tri Van Le[1], Khanh Quoc Nguyen[2], and Vijay Varadharajan[2]

[1] College of Engineering and Applied Science
University of Wisconsin, Milwaukee
EMS Building 3200 N Cramer Street, Milwaukee, WI 53201
lvtri@cs.uwm.edu

[2] School of Computing and Information Technology
University of Western Sydney, Nepean
P.O BOX 10 Kingswood, NSW 2747 Australia
{qnguyen,vijay}@cit.nepean.uws.edu.au

Abstract. The problem of proving a number is of a given arithmetic format with some prime elements, is raised in RSA undeniable signature, group signature and many other cryptographic protocols. So far, there have been several studies in literature on this topic. However, except the scheme of Camenisch and Michels, other works are only limited to some special forms of arithmetic format with prime elements. In Camenisch and Michels's scheme, the main building block is a protocol to prove a committed number to be prime based on algebraic primality testing algorithms. In this paper, we propose a new protocol to prove a committed number to be prime. Our protocol is $O(t)$ times more efficient than Camenisch and Michels's protocol, where t is the security parameter. This results in $O(t)$ time improvement for the overall scheme.

1 Introduction

In many applications, it is essential to prove that a number is of an arithmetic format of which some elements are prime. This problem is raised in many recently proposed cryptographic protocols [4,11,13,14]. The protocols proposed in [13,14] are sound only if there exists a proof that a given number n is a product of two safe prime numbers. In [11], the divisible electronic cash scheme requires a zero-knowledge proof that a committed number is a product of two primes. Furthermore, though not necessary, it is recommended in [15] to show that a number n is a product of two prime numbers p, q such that $(p+1)/2$ and $(q+1)/2$ are also primes.

Previously, there have been several studies in the literature related to this subject. de Graaf and Peralta [12] provided an efficient proof that a given number n is of the form $n = p^r q^s$, where r and s are odd, p and q are primes. Another protocol is that of Boyar et al.[1] which proves a given number n is square-free, i.e., all the factors of n are singular. Gennaro at el.[16] extended these two results to show that a number n is a product of quasi-safe primes p, q, i.e., each of $(p-1)/2$ and $(q-1)/2$ has only one prime factor.

K. Y. Lam, E. Okamoto and C. Xing (Eds.): ASIACRYPT'99, LNCS 1716, pp. 208–218, 1999.

More recently, Camenisch and Michels [3] proposed a general solution for this problem. They used the general paradigm of proving that a number is of a specific arithmetic format [6,8,10,17]. In this paradigm, the prover builds an arithmetic circuit corresponding to the arithmetic relation. She then commits all inputs of the circuit in some commitments. The proof is then a set of protocols showing that the prover knows the secret elements concealed in the commitments and the final output of the circuit is the desired number and the relations between committed elements correspond to the arithmetic circuit. As all elements are concealed in the commitments, in order to demonstrate that some elements are prime, the prover must be able to show that committed numbers are prime. In [3], a proof that a committed number is prime is at least $O(t^2)$ fold more expensive than a proof of an arithmetic relation.

Our main contribution of this paper is an efficient protocol to prove in (statistical) zero-knowledge that a committed number is prime. Our technique results in an efficient proof that a number is is of an arithmetic format where some involved elements are prime. The protocol is $O(t)$ times more efficient than the protocol in [3], where $1/2^t$ is the error probability of the proof. This consequently leads to $O(t)$ fold improvement of the general protocol.

2 Preliminary

In this section, we review a commitment scheme and statistical zero-knowledge proofs that demonstrate basic arithmetic relations amongst some commitments. The commitment scheme is unconditional hiding and conditionally binding and other protocols are statistical zero-knowledge. They are all well-known in literature. The reader is referred to [2,3,6,10,11] for detailed discussions of these protocols and other variations.

In the following, we assume that $G = \langle g \rangle$ is a group of large known order Q over the finite field $\mathcal{Z}_P$ for some known prime P and h is a second generator of the group such that $\log_g h$ is not known to the prover.

A commitment scheme: To commit an element x, the prover chooses $r \in_R \mathcal{Z}_Q$ and sends $y = g^x h^r$ to the verifier. Given y, it is infeasible for the verifier to obtain any information about x and it is infeasible for the prover to find two different pairs (x, r) and (x', r') such that $y = g^x h^r = g^{x'} h^{r'}$ unless she can compute $\log_g(h)$.

Proving the knowledge of a representation: of the element y to the bases $g_1, .., g_k$, involves proving the knowledge of $x_1, \ldots, x_k$ such that $y = \prod_{i=1}^k g_i^{x_i}$. The protocol works as follows. The prover chooses $r_1, \ldots, r_k \in_R \mathcal{Z}_Q$, computes $w := \prod_{i=1}^k g_i^{r_i}$, and sends w to the verifier. The verifier picks a random challenge $c \in_R \{0,1\}^t$ and sends it to the prover. The prover computes $s_i := r_i - cx_i \bmod Q$ for $i = 1, \ldots, t$. The verifier accepts , iff $w = y^c \prod_{i=1}^k g_i^{s_i}$. Following the notations of [3,4], we denote this protocol as $\mathrm{PK}\{(\alpha_1, \ldots, \alpha_k) : y = \prod_{i=1}^k g_i^{\alpha_i}\}$.

Proving the equality of discrete logarithm: to the bases g_1 and h_1 in the representation of elements y_1, y_2 to the bases $(g_1, \ldots, g_k)$ and $(h_1, \ldots, h_k)$ respectively, involves proving the knowledge of $x_1, \ldots, x_k$, $z_1, z_2, \ldots, z_k$ such that $x_1 = z_1$, $y_1 = \prod_{i=1}^{k} g_i^{x_i}$ and $y_2 = \prod_{i=1}^{k} h_i^{z_i}$. The protocol works as follows. The prover chooses $r_1, \ldots, r_k \in_R \mathcal{Z}_Q$ and $u_2, \ldots, u_k \in_R \mathcal{Z}_Q$, computes $w_1 := \prod_{i=1}^{k} g_i^{r_i}$ and $w_2 := h_1^{r_1} \prod_{i=2}^{k} h_i^{u_i}$, and sends w_1, w_2 to the verifier. The verifier picks a random challenge $c \in_R \{0,1\}^t$ and sends it to the prover. The prover computes $s_i := r_i - cx_i \bmod Q$ for $i = 1, \ldots, k$ and $v_i := u_i - cz_i \bmod Q$ for $i = 2, \ldots, k$. The verifier accepts iff $w_1 = y_1^c g_1^{s_1} \prod_{i=2}^{k} g_i^{s_i}$ and $w_2 = y_2^c h_1^{s_1} \prod_{i=2}^{k} h_i^{u_i}$. We denote this protocol as $\mathrm{PK}\{(\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_k) : \alpha_1 = \beta_1 \wedge y_1 = \prod_{i=1}^{k} g_i^{\alpha_i} \wedge y_2 = \prod_{i=1}^{k} h_i^{\beta_i}\}$.

Proving that a discrete logarithm is in a given range: This protocol proves that the discrete logarithm x of $y = g^x h^r$ satisfies $x \in [a, b]$ for given parameters $a, b < Q/2$. Several such schemes exist in literature. We review the scheme of [11] here. The protocol works as follows ($e = \lfloor (b-a)/3 \rfloor - 1$):

- The prover chooses $x_1, r_1, r_2 \in_R [0, e]$, sets $x_2 = x_1 - e$ and $w_1 := g^{x_1} h^{r_1}$ and $w_2 := g^{x_2} h^{r_2}$. She then sends the un-order pair (w_1, w_2) to the verifier.
- The verifier chooses $c \in_R [0, 1]$ and sends c to the prover.
- If $c = 0$, the prover sends x_1, x_2, r_1, r_2 to the verifier. Otherwise, the prover sends $(x + x_j, r + r_j)$ ($j = 1$ or 2) such that $x + x_j \in [a + e, b - e]$.
- The prover accepts iff $w_1 = g^{x_1} h^{r_1}$, $w_2 = g^{x_2} h^{r_2}$ when $c = 0$ and $yw_j = g^{x+x_j} h^{r+r_j}$ when $c = 1$.

This is repeated t times to achieve the error probability of $1/2^t$. We denote this protocol as $\mathrm{PK}\{(\alpha_1, \ldots, \alpha_k) : y = \prod_{i=1}^{k} g_i^{\alpha_i} \wedge \alpha_1 \in [a, b]\}$. The scheme is not very efficient. Constructions of [5,10] are much more efficient but use a composite modulo m and require a proof that m is a product of two primes.

Building on these protocols, we next present zero-knowledge protocols to prove secret modular quadratic residue and secret modular exponentiation. Both protocols use a protocol that demonstrates secret modular multiplicative relation. The protocols to prove secret modular multiplicative and exponentiation relations, were introduced in [3]. We present them here for the sake of completeness.

Secret modular multiplicative relation: Assume that a prover has committed to x, y, z, n in the commitments c_x, c_y, c_z and c_n such that $0 < x, y, z, n < 2^l$ where $l = |Q|/2 - 1$. The prover can convince the verifier that $xy \equiv z \bmod n$ using the following proofs:

(1) $\mathrm{PK}\{(x, r_x) : c_x = g^x h^{r_x} \wedge x \in [1, 2^l]\}$.
(2) $\mathrm{PK}\{(y, r_y) : c_y = g^y h^{r_y} \wedge y \in [1, 2^l]\}$.
(3) $\mathrm{PK}\{(z, r_z) : c_z = g^z h^{r_z} \wedge z \in [1, 2^l]\}$.
(4) $\mathrm{PK}\{(n, r_n) : c_n = g^n h^{r_n} \wedge n \in [1, 2^l]\}$.
(5) $\mathrm{PK}\{(u, r_u) : c_u = g^u h^{r_u} \wedge u \in [1, 2^l]\}$.

(6) $\mathrm{PK}\{(y, u, r_y, r_u, \rho) : c_z = c_x^y c_n^u h^\rho \wedge c_y = g^y h^{r_y} \wedge c_u = g^u h^{r_u}\}$

Here clause (6) is a combination of two proofs of equality of discrete logarithms [4]. It is straightforward to prove that clause (6) is a zero-knowledge proof of knowledge. We denote this protocol as $\mathrm{PK}\{(x, y, z, n) : xy \equiv z \bmod n\}$.

Lemma 1. *Let x, y, z, n be the values committed in c_x, c_y, c_z and c_n respectively. $\mathrm{PK}\{(x, y, z, n) : xy \equiv z \bmod n\}$ is a statistical zero-knowledge proof of $xy \equiv z \bmod n$.*

Proof. The statistical zero-knowledge claim follows from the statistical zero-knowledge property of the protocol components. We now show why the multiplicative relation holds.

We let the knowledge extractor to run the protocol with the prover. From (1)(2)(3) (4) and (5), the knowledge extractor can obtain $\tilde{x}, \tilde{y}, \tilde{z}, \tilde{n}, \tilde{u}\tilde{r_x}, \tilde{r_y}, \tilde{r_z}, \tilde{r_n}$ and $\tilde{r_u}$ such that $c_x = g^{\tilde{x}} h^{\tilde{r_x}}$, $c_y = g^{\tilde{y}} h^{\tilde{r_y}}$, $c_z = g^{\tilde{z}} h^{\tilde{r_z}}$, $c_n = g^{\tilde{n}} h^{\tilde{r_n}}$ and $c_u = g^{\tilde{u}} h^{\tilde{r_u}}$. Moreover $0 < \tilde{x}, \tilde{y}, \tilde{z}, \tilde{n}, \tilde{u} < 2^l$.

Furthermore from (6), the extractor can extract $\tilde{\rho}, \tilde{y}, \tilde{u}, \tilde{r_y}$ and $\tilde{r_u}$ such that $c_z = c_x^{\tilde{y}} c_n^{\tilde{u}} h^{\tilde{\rho}}$, $c_y = g^{\tilde{y}} h^{\tilde{r_y}}$ and $c_u = g^{\tilde{u}} h^{\tilde{r_u}}$. Assuming that $\log_g(h)$ is not known, this shows $\tilde{z} = \tilde{x}\tilde{y} + \tilde{u}\tilde{n} \bmod Q$. But $0 < \tilde{x}, \tilde{y}, \tilde{z}, \tilde{n}, \tilde{u} < 2^l$ and $l < |Q|/2$. Hence $\tilde{z} = \tilde{x}\tilde{y} + \tilde{u}\tilde{n} \bmod Q$ holds only if $\tilde{z} = \tilde{x}\tilde{y} + \tilde{u}\tilde{n}$ holds, i.e., $\tilde{z} = \tilde{x}\tilde{y} \bmod \tilde{n}$ holds for the committed values $\tilde{x}, \tilde{y}$ and $\tilde{n}$.

Secret modular quadratic residue: Using the proof of secret modular multiplicative relation, the prover can prove that x is a quadratic modulo n for x and n committed in c_x and c_n respectively using $\mathrm{PK}\{(y, y, x, n) : y^2 \equiv x \bmod n\}$. This is because if there exists y such that $y^2 \equiv x \bmod n$, then x is a quadratic residue modulo n. Let us denote this protocol as $\mathrm{PK}\{(x, n) : x \in QR_n\}$.

Secret modular exponentiation relation: Given the commitments c_x, c_y, c_z and c_n, to prove that $x^y \equiv z \bmod n$, the prover proceeds as follows:

- Let $y = \sum_{i=0}^{l-1} y_i 2^i$, $(y_i \in [0, 1])$ and $x_0 = x$, $x_i = x_{i-1}^2 \bmod n$ $(i = 1, \ldots, l-1)$. Also let $u_i = x_i^{y_i}$ and $w_i = w_{i-1}u_i \bmod n$ $(i = 0, \ldots, l-1$ and $w_0 = 1)$.
- The prover commits to all x_i, y_i, u_i, w_i $(i = 1, \ldots, l-1)$ in the commitments:

$$\begin{aligned} c_{y_i} &= g^{y_i} h^{\hat{r}_i} \\ c_{x_i} &= g^{x_i} h^{\tilde{r}_i} && (c_{x_0} = c_x) \\ c_{u_i} &= g^{u_i} h^{\check{r}_i} \\ c_{w_i} &= g^{w_i} h^{\bar{r}_i} && (c_{w_{l-1}} = c_z). \end{aligned}$$

  She then sends all her commitments to the verifier.
- The prover and the verifier now engage in the following protocols $(i = 0, \ldots, l-2)$.
  (1) $\mathrm{PK}\{(x_i, x_i, x_{i+1}, n) : x_i^2 \equiv x_{i+1} \bmod n\}$
  (2) $\mathrm{PK}\{(w_i, u_{i+1}, w_{i+1}, n) : w_i u_{i+1} \equiv w_{i+1} \bmod n\}$

(3) $\mathrm{PK}\{(\omega) : (\prod_{i=0}^{l-1} c_{y_i}^{2^i})/c_y \equiv h^\omega \bmod Q\}$
(4) $\mathrm{PK}\{(x_i, y_i, u_i) : y_1 \in [0,1] \wedge u_i \equiv x_i^{y_i}\}$ using the sub-protocol described below.
(5) $\mathrm{PK}\{(w_0) : w_0 \equiv 1\}$

Let us denote this protocol as $\mathrm{PK}\{(x, y, z, n) : x^y \equiv z \bmod n\}$. The intuition is that clause (1) shows that $x_{i+1} = x_i^2 \bmod n$, $\forall i = 0, \dots, l-2$. Because of $c_{x_0} = c_x$, we have $x_0 = x$ and thus $x_i = x^{2^i}$. Next clause (4) shows $u_i \equiv x_i^{y_i}$. Hence clauses (1) and (4) show $u_i = x^{y_i 2^i} \bmod n$. Furthermore, clauses (2) and (5) show that $w_{i+1} = w_i u_{i+1} \bmod n$ $(i = 0, \dots, l-2)$ and $w_0 = 1$, this implies that $w_i = \prod_{j=0}^{i} u_i = \prod_{j=0}^{i}(x^{y_j 2^j}) \bmod n$. This further implies that $w_{l-1} = \prod_{j=0}^{l-1} x^{y_j 2^j} = x^{\sum_{j=0}^{l-1} y_j 2^j} \bmod n$. However clause (3) shows that the discrete logarithms of c_y and $(\prod_{i=0}^{l-1} c_{y_i}^{2^i})$ to base g are equivalent, i.e., $y = \sum_{j=0}^{l-1} y_j 2^j$. Thus it is clear that $w_{l-1} = x^{\sum_{j=0}^{l-1} y_j 2^j} \bmod n = x^y = \bmod n$. Finally, as $c_{w_{l-1}} = c_z$ and commitments are conditionally binding, we have $z = w_{l-1} = x^y \bmod n$.

Now it remains to show the existence of the sub-protocol.

Sub-protocol Given three commitments c_{x_i}, c_{y_i} and c_{u_i}, the sub-protocol proves that $y_i \in [0,1]$, $u_i = x_i^{y_i}$. Because $y_i = 0$ or 1, we only have to consider two cases:

1. Case 1: $y_i = 0$. We have $c_{y_i} = h^{\hat{r}_i}$ for some $\hat{r}_i$. Also $u_i = x_i^{y_i}$ iff $c_{u_i} = gh^{\check{r}_i}$. This is equivalent to showing that $c_{u_i}/g = h^{r_i}$ for some r_i.
2. Case 2: $y_i = 1$. This means $c_{y_i} = gh^{\hat{r}_i}$ or $c_{y_i}/g = h^{\hat{r}_i}$. Also now we have $c_{u_i} = g^{x_i} h^{\check{r}_i}$ or $c_{u_i}/c_{x_i} = h^{r_i}$ for some r_i.

Thus to show that $u_i = x_i^{y_i}$, one has to show the knowledge of:

$$(c_{y_i} = h^{\hat{r}_i} \wedge c_{u_i}/g = h^{r_i}) \vee (c_{y_i}/g = h^{\hat{r}_i} \wedge c_{u_i}/c_{x_i} = h^{r_i}).$$

For clarity, we present the proof for:

$$(\alpha = h^{r_\alpha} \wedge \beta = h^{r_\beta}) \vee (\eta = h_\eta^r \wedge \kappa = h^{r_\kappa}).$$

Without loss of generality, we assume that the prover knows the $\log_h \alpha$ and $\log_h \beta$. The protocol works as follows:

– The prover chooses $\rho_1, \sigma_2, \mu_2, \lambda_2$, computes $\psi_1 := h^{\rho_1}$, $\psi_2 := h^{\sigma_2} \eta^{\mu_2} \kappa^{\lambda_2}$ and sends ψ_1, ψ_2 to the verifier.
– The verifier chooses random $\lambda, \mu \in \mathcal{Z}_Q$. He sends (λ, μ) to the prover.
– The prover computes $\mu_1 := \mu \oplus \mu_2$, $\lambda_1 := \lambda \oplus \lambda_2$ and $\sigma_1 := \rho_1 - \mu_1 r_\alpha - \lambda_1 r_\beta$. She then sends $(\sigma_1, \mu_1, \lambda_1, \sigma_2, \mu_2, \lambda_2)$ to the verifier.
– The verifier accepts iff $\mu = \mu_1 \oplus \mu_2$, $\lambda = \lambda_1 \oplus \lambda_2$, $\psi_1 = h^{\sigma_1} \alpha^{\mu_1} \beta^{\lambda_1}$ and $\psi_2 = h^{\sigma_2} \eta^{\mu_2} \kappa^{\lambda_2}$.

This is an example of zero-knowledge proof of arbitrary monotonic statements built with $\wedge$'s and $\vee$'s. In this protocol, $\oplus$ denotes the XOR operation. Such proofs are discussed in [2,7]. For this reason, its security proof is omitted here. The reader is referred to [2,7] for further discussions.

3 Main Result

Our main result is an efficient zero-knowledge proof of a committed number n to be prime. The proof consists of two steps. First we show that n has only one prime factor. Next we show that n is square free. Clearly then n must be prime. We assume that that $2\tau \geq |n| \geq 2t$ for some known τ and t, which is also the security parameter. This can be proven using the protocol that proves a discrete logarithm is in a given range described earlier.

3.1 Proving That n Has Only One Prime Factor

Given an odd *prime number* n committed in a commitment $\texttt{commit}(n)$, this subsection presents a statistical zero-knowledge protocol which convinces the prover that n has only one prime factor. First we need to show that n is odd. This is done as follows:

- PK$\{k, 2, n-1, 2^{2\tau+1} : 2k \equiv n-1 \bmod 2^{2\tau+1}\}$, where $\texttt{commit}(n-1)$ is computed as $\texttt{commit}(n)/g$.
- $2^{2t-1} \leq k \leq 2^{2\tau}$ with the proof of a discrete logarithm in a given range.

As $n-1 < 2^{2\tau+1}$ and $\texttt{commit}(n-1)/\texttt{commit}(n) = g$, the proof demonstrates that $2k \equiv n-1$. Next we show that n has only one prime factor. There are two different methods of proving that. One works for the case $n \equiv 3 \bmod 4$. The other works for any odd n. The former is more efficient. So far $n \equiv 3 \bmod 4$ shows no apparent security weakness. In fact it is recommended in many applications(e.g. in Blum numbers) to choose prime numbers of this form. We present both methods here.

Specific case $n \equiv 3$ mod 4. The proof that n has only a prime factor, is the following protocol:

- COMMON INPUT: a commitment $\texttt{commit}(n)$ of a prime number n satisfying $n \equiv 3 \bmod 4$.
- Repeat t times:
  - RANDOM INPUT: $0 < x < 2^{\tau}$
  - Prover: outputs a quadratic residue z modulo n out of $\pm x$, i.e, $z = x$ or $-x$, a commitment $\texttt{commit}(y)$ of the square root $y = z^{1/2} \bmod n$) and proves that z is quadratic residue modulo n using PK$\{(z, n) : z \in QR_n\}$.
- Verifier: accepts iff he accepts all t proofs.

The zero-knowledge property of the protocol comes from the statistical zero-knowledge of involved proofs. We do not further evaluate them. Here we prove the soundness and completeness of the protocol. Before progressing further, let us review some basic number theory facts [9]:

1. For any odd prime number n, (-1) is a quadratic non-residue modulo n if and only if $n \equiv 3 \bmod 4$.

2. Let n be an odd prime number. For any values u and v, uv is a quadratic residue modulo n if and only if either both or none of u and v are quadratic residue.
3. From (2), we can derive that for an odd prime n and a quadratic non-residue u, only one of v or uv is quadratic residue modulo n for any given v.
4. If n has more than one prime factor, a random number x is quadratic residue with no better than 1/4 probability.

Next to the proof of completeness and soundness.

- **Completeness:** Because n is an odd prime and $n \equiv 3 \bmod 4$, (-1) is a quadratic non-residue modulo n. This means that out of $\pm x$, there is one and only one quadratic residue modulo n. The protocol completeness follows.
- **Soundness:** Observe that if an odd number n has more than one prime factor, then for a random non-zero number, the probability that it is quadratic modulo n is at most 1/4. If (x) is a quadratic non-residue, from [16], we have that $(-x)$ is a quadratic residue is with the probability of 1/2. Thus the error probability of each round is 1/2. After t rounds, the error probability is $1/2^t$.

General case. The proof that n has only one prime factor, is based on the following protocol:

- COMMON INPUT: n an odd prime.
- Repeat $24t$ times:
  - RANDOM INPUT: $0 < x < 2^\tau$
  - Prover: either says that x is quadratic non-residue or runs $\mathrm{PK}\{(x, n) : x \in QR_n\}$ to prove that x is quadratic.
- Verifier: accepts iff he accepts at least $9t$ proofs.

The zero-knowledge property is straightforward. The completeness and soundness intuition is as follows. To convince the verifier that n is prime, clearly the prover must try to show as many random input x's as possible are in QR_n. Observe that the probability of a random $x \in QR_n$ is 1/2 if n is prime and at most 1/4 if n has 2 or more prime factors. Thus ideally, out of $24t$ random x's, there should be $12t$ quadratic residues if n is prime and at most $6t$ quadratic residues if n has more than one prime factor. Using elementary probability theory, we have the following lemmas (see appendix for the proof of the lemmas):

Lemma 2. *For a number $t \geq 40$, the probability that there exists $9t$ quadratic residues modulo n out of $24t$ random numbers is at least $1 - 1/2^t$ if a random number is quadratic residue with the probability of $1/2$.*

Lemma 3. *For a number $t \geq 40$, the probability that there exists $9t$ quadratic residues modulo n out of $24t$ random numbers is at most $1/2^t$ if a random number is quadratic residue with the probability of $1/4$.*

Subsequently, we choose the threshold $9t$ so that the error probability of the protocol (i.e. the probability of failure for the honest prover, and the probability of success for the dishonest prover) is at most $(1/2^t)$ where the value of t is assumed to be at least 40.

3.2 Proving That n Is Square-Free

In this step, we can safely assume that n has only an odd prime factor, i.e., $n = p^{\alpha}$ for a prime p and $\alpha \geq 1$. In order to prove that n is square-free, the prover and the verifier runs the following protocol:

- RANDOM INPUT: $0 < x < 2^{\tau}$
- Prover: runs the proof $\mathrm{PK}\{(x, n, x, n) : x^n \equiv x \bmod n\}$ to show $x^n \equiv x \bmod n$.
- Verifier: accepts that n is square-free if he accepts the proof $\mathrm{PK}\{(x, n, x, n) : x^n \equiv x \bmod n\}$

Again, the zero-knowledge property of the protocol comes from the statistical zero-knowledge of the associated proofs. We do not evaluate them further. The completeness is straightforward. It remains to show the soundness of the protocol.

Theorem 4. *Assume that n has only a prime factor, then the protocol proves that n is square-free and so prime, with overwhelming probability.*

Proof. Let $n = p^{\alpha}$, where p is the prime factor of n. To prove the theorem, we show that if $\alpha > 1$, $x^n \equiv x \bmod n$ happens with negligible probability for a randomly chosen x.

First, consider the case $gcd(x, n) \neq 1$. As $n = p^{\alpha}$, p divides $gcd(x, n)$. This implies that p^{α} divides x^{α}. But since x^{α} divides x^n, we have n divides x^n. Thus $x^n \equiv x \bmod n$ is equivalent to $x \equiv 0 \bmod n$ which happens with negligible probability for $0 < x < 2^{\tau}$.

Now we can conclude that $x^n \equiv x \bmod n$ occurs with non-negligible probability only if $gcd(x, n) = 1$. This means that $x \in Z_n^*$, where Z_n^* denotes the set of all numbers in Z_n relatively prime to n. The order of Z_n^* is $\phi(n) = (p-1)p^{\alpha-1}$. So for $x \in Z_n^*$, $x^{n-1} \equiv 1 \bmod n$ holds only if $g^{gcd(n-1,\phi(n))} = g^{p-1} = 1$. As the order of Z_n^* is $(p-1)p^{\alpha-1}$, there are only $(p-1)$ such x's in Z_n^*. Hence for a random $x \in Z_n^*$. the probability that $x^n \equiv x \bmod n$ is $(p-1)/(p-1)p^{\alpha-1} = 1/p^{\alpha-1}$ which is negligible if $\alpha > 1$

3.3 Efficiency Comparison with Previous Works

We consider a basic proof of knowledge of secret modular multiplicative relation as the basic proof. Each secret modular exponentiation relation proof is estimated to cost about $3t$ basic proofs.

The only other general protocol is that of Camenisch and Michels [3] requires t modular exponentiation relation proof. This is equivalent to about $3t^2$ basic proofs.

In our more efficient but less general version, the first step which proves n to have one prime factor uses t secret modular quadratic residue proofs. The second step is in fact a proof of secret modular exponentiation relation, which uses $3t$ basic proofs. Thus our total computation and communication costs is $4t$ basic proofs. This means an improvement of the order of $0.75(t)$.

In the more general but less efficient version, the first step requires $24t$ independent procedures. In each procedure, the prover either says a random number is a quadratic non-residue or proves that the number is a quadratic residue. The cost of saying that a number is a quadratic non-residue, is negligible. The cost of proving a quadratic residue is equivalent to a basic proof. Of course the verifier can always accept the proof once $9t$ proofs of quadratic residuosity are achieved. Thus in practice, the first step costs $9t$ basic proofs. The second step which is the same for both of our protocols, requires $3t$ basic proofs. Hence on average, the protocol costs $12t$ proofs. This means that the gained efficiency over the protocol of [3] is of the order of $0.33(t)$.

In practice, if $t = 40$, our two protocols are about an order of 30 and 12 times respectively more efficient than the protocol of [3]. For the case $t = 80$, the figures are about 60 and 25 times, respectively.

3.4 Generating a Random Number x

In both steps, the protocol makes use of some random numbers x's. In case such random numbers do not exist, a random number x can be generated as follows:

- The prover chooses a random x_1, commits it in the commitment c_{x_1} and sends it to the verifier.
- The verifier chooses a random x_2, commits it in the commitment c_{x_2} and sends it to the prover.
- The prover opens the commitment c_{x_1} and sends x_1 to the verifier.
- The verifier opens the commitment c_{x_2} and sends x_2 to the prover.
- If x_1 is consistent with c_{x_1} and x_2 is consistent with c_{x_2}, then the random number x is computed as $x = x_1 + x_2 \bmod 2^{|N|}$.

This technique is known to be secure. The reader is referred to [3,16] for further details.

References

1. J. Boyar, K. Friedl and C. Lund, Practical zero-knowledge proofs: giving hints and using deficiencies, Journal of Cryptology, 4(3):185-206, 1991. 208
2. S. Brands, “Rapid Demonstration of Linear Relations Connected by Boolean Operators”, Proceedings of Eurocrypt’97, LNCS 1223, pp. 318-333. 209, 212, 212
3. J. Camenisch and M. Michels, “Proving in Zero-Knowledge that a Number is the Product of Two Safe Primes”, Proceedings of Eurocrypt’99, LNCS 1592, pp. 106–121. Also appeared as BRICS Technical Report RS-98-29. 209, 209, 209, 209, 209, 210, 215, 216, 216, 216
4. J. Camenisch and M. Stadler, “Efficient group signature schemes for large groups”, Proceedings of CRYPTO ’97, LNCS 1294, pages 410-424. 208, 209, 211
5. A. Chan, Y. Frankel and T. Tsiounis, “Easy come-easy go divisible cash”, Proceedings of Eurocrypt’98, LNCS , pp. 561-575. 210

6. R.Cramer and I. Damgard, “Zero-knowledge for Finite Field Arithmetic or: Can Zero-knowledge be for Free?”, In Proceedings of CRYPTO ’98, LNCS 1462, pp. 424–441, 1998. 209, 209
7. R.Cramer, I.Damgard and B.Schoenmakers, Crypto’94, “Proofs of partial knowledge and simplified design of witness hiding protocols”, Proceedings of CRYPTO ’94, LNCS 839, pp.174-187.
8. A. De Santis, G. Di Crescenzo and G. Persiano, “Secret sharing and perfect zero-knowledge”, Proceedings of CRYPTO’ 93, LNCS 773, pp.73-84. 212, 212 209
9. A. Menezes, P. van Oorschot and S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996. 213
10. E. Fujisaki and T. Okamoto, “Statistical zero-knowledge protocols to prove modular polynomial relation”, Proceedings of CRYPTO ’97, LNCS 1294, pp. 16-30. 209, 209, 210
11. T. Okamoto An efficient divisible electronic cash scheme, Proceedings of CRYPTO ’95, LNCS, pp. 439-451. 208, 208, 209, 210
12. J. van de Graaf and R. Peralta, “A simple and secure way to show the validity of your public key”, Proceedings of CRYPTO ’87, LNCS 293, pp. 128-134. 208
13. R. Gennaro, H. Krawczyk and T. Rabin, “RSA-based undeniable signatures”, Proceedings of CRYPTO ’97, LNCS 1294, pp. 132-149. 208, 208
14. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, “Robust and efficient sharing of RSA functions”, Proceedings of CRYPTO ’96, LNCS 1109, pp. 157-172. 208, 208
15. K.Koyama, U. Maurer, T. Okamoto and S. Vanstone, “New public-key schemes based on elliptic curves over the ring Z_n”, Proceedings of CRYPTO ’91, pp.252-266 208
16. R. Gennaro, D. Micciancio and T. Rabin, “An efficient non-interactive statistical zero-knowledge proof system for quasi-safe prime products”, in Proceedings of 5rd ACM conference on Computer and Communication Security, 1998. 208, 214, 216
17. M. Stadler, “Cryptographic Protocols for Revocable Privacy”, Ph.D Thesis, Swiss Federal Institute of Technology, Zurich 1996. 209

A Proof of Lemma 2

Given a random number that is quadratic residue with the probability of 1/2, the probability that there are exact i quadratic residues in $24t$ random numbers is $\frac{1}{2^{24t}}\binom{24t}{i}$. Thus the probability that there are at least $9t$ quadratic residues in $24t$ random numbers is $\mathcal{P} = \sum_{i=9t}^{24t} \frac{1}{2^{24t}}\binom{24t}{i}$. We then have $\mathcal{P} = \sum_{i=9t}^{24t} \frac{1}{2^{24t}}\binom{24t}{i} > \sum_{i=12t}^{21t} \frac{1}{2^{24t}}\binom{24t}{i} = \sum_{i=0}^{9t} \frac{1}{2^{24t}}\binom{24t}{i+3t}$.

Furthermore, since $1 = \sum_{i=0}^{24t} \frac{1}{2^{24t}}\binom{24t}{i}$, we have $1 - \mathcal{P} = \sum_{i=0}^{9t-1} \frac{1}{2^{24t}}\binom{24t}{i}$.

Furthermore for $0 \le i < 9t$, we have

$$\frac{\binom{24t}{i+3t}}{\binom{24t}{i}} = \frac{\frac{24t!}{(3t+i)!(24t-3t-t)!}}{\frac{24t!}{i!(24t-i)!}} = \frac{(24t-i)\dots(24t-3t-i)}{(3t+i)\dots i}.$$

As $0 \leq i < 9t$,

$$\frac{\binom{24t}{i+3t}}{\binom{24t}{i}} \geq \frac{(24t-9t)\dots(24t-3t-9t)}{(3t+9t)\dots(9t)} = \frac{(15t)\dots(12t)}{(12t)\dots(9t)} > \left(\frac{15}{12}\frac{12}{9}\right)^{3t/2} > 2^t.$$

So we now have

$$\mathcal{P} > \sum_{i=0}^{9t} \frac{1}{2^{24t}} \binom{24t}{i+3t} > 2^t \sum_{i=0}^{9t} \frac{1}{2^{24t}} \binom{24t}{i} = 2^t(1-\mathcal{P})$$

This means $\mathcal{P} > 1 - 1/2^t$ which completes the proof of lemma 2.

B Proof of Lemma 3

Given a random number that is quadratic residue with the probability of $1/4$, the probability that there are exact i quadratic residues in $24t$ random numbers is $\frac{3^{24t-i}}{4^{24t}}\binom{24t}{i}$. Thus the probability that there are at least $9t$ quadratic residues in $24t$ random numbers is $\mathcal{P} = \sum_{i=9t}^{24t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i}$. We then have $\mathcal{P} = \sum_{i=9t}^{24t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i} < 5\sum_{i=9t}^{12t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i} = 5\sum_{i=6t}^{9t} \frac{3^{21t-i}}{4^{24t}}\binom{24t}{i+3t}$.

Further, since $1 = \sum_{i=0}^{24t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i}$, we have

$$1 - \mathcal{P} = \sum_{i=0}^{9t-1} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i} > \sum_{i=6t}^{9t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i}.$$

As $6t \leq i < 9t$ and $t \geq 40$,

$$\frac{\frac{3^{24t-i}}{4^{24t}}\binom{24t}{i}}{\frac{3^{21t-i}}{4^{24t}}\binom{24t}{i+3t}} = 3^{3t}\frac{\binom{24t}{i}}{\binom{24t}{i+3t}} = 3^{3t}\frac{(3t+i)\dots(i)}{(24t-i)\dots(21t-i)} \geq 3^{3t}\frac{(9t)\dots(6t)}{(18t)\dots(15t)} > 5(2^t).$$

So we now have

$$(1-\mathcal{P}) > \sum_{i=6t}^{9t} \frac{3^{24t-i}}{4^{24t}}\binom{24t}{i} > 5(2^t)\sum_{i=6t}^{9t} \frac{3^{21t-i}}{4^{24t}}\binom{24t}{i+3t} > (2^t)\mathcal{P}.$$

This shows $\mathcal{P} < 1/2^t$ which completes the proof of lemma 3.