The Statistical Zero-knowledge Proof for Blum Integer Based on Discrete Logarithm*

Chunming Tang † Zhuojun Liu ‡ Jinwang Liu§

Abstract

Blum integers (BL), which has extensively been used in the domain of cryptography, are integers with form $p^{k_1}q^{k_2}$, where p and q are different primes both $\equiv 3 \ mod \ 4$ and k_1 and k_2 are odd integers. These integers can be divided two types: 1) $M = pq$, 2) $M = p^{k_1}q^{k_2}$, where at least one of k_1 and k_2 is greater than 1.

In [3], Bruce Schneier has already proposed an open problem: *it is unknown whether there exists a truly practical zero-knowledge proof for $M(= pq) \in BL$.* In this paper, we construct two statistical zero-knowledge proofs based on discrete logarithm, which satisfies the two following properties: 1) the prover can convince the verifier $M \in BL$; 2) the prover can convince the verifier $M = pq$ or $M = p^{k_1}q^{k_2}$, where at least one of k_1 and k_2 is more than 1.

In addition, we propose a statistical zero-knowledge proof in which the prover proves that a committed integer a is not equal to 0.

Keywords: cryptography, Blum integer, statistical zero-knowledge
MR: 94A60

1 Introduction

Informally, an integer M is a Blum integer, in symbols $M \in BL$, if and only if $M = p^{k_1}q^{k_2}$, where p and q are different primes both $\equiv 3 \ mod \ 4$ and

*This work was supported in part by 973 project G1998030600.

†ctang@mmrc.iss.ac.cn, Key Laboratory of Mathematics Mechanization, Institute of Systems Science, Chinese Academy of Sciences, P.R.China

‡zliu@zgc.gov.cn, Key Laboratory of Mathematics Mechanization, Institute of Systems Science, Chinese Academy of Sciences, P.R.China

§jwliu@xtnu.edu.cn, Department of Mathematics, Hunan University of Science and Technology. P.R.China

k_1 and k_2 are odd integers. These integers have some special properties[1, 2, 4], and were first used for cryptographic purposes by Blum in [5].

Usually, it is easy to construct a Blum integer(for example, we select randomly two different prime numbers p and q, which satisfy both $\equiv 3 \ mod \ 4$, then get a Blum integer M by $M = p^{k_1} q^{k_2}$, where k_1 and k_2 are odd.); however, it is very difficult to prove directly an integer $M \in BL$, because the composite integer M must be first factored in order to prove this case. It is well known that the problem of factoring composite integers is probabilistic polynomial time reducible to the problem of extracting square modulo composite integers, however, up to now, no efficient algorithm is known for deciding quadratic residuosity modulo composite numbers whose factorization is not given[4].

As a result, some interesting ways were proposed in order to avoid factoring composite integers, for example, A.D.Santis, G.D.Crescenzo, and G.Persiano[1] and J.V.D Graaf and R. Peralta[2] proved independently an integer $M \in BL$ from properties of Blum integers and not from their forms, that is , if an integer M satisfies some properties(seen in section 2), it is a Blum integer. In particular, they used the notion of zero-knowledge proof.

The notion of zero-knowledge proofs was introduced by Goldwasser, Micali and Rackoff[6]. A remarkable property of such proofs is that a prover can convince a polynomial bounded verifier of a fact while not releasing anything else. Zero-knowledge proof has received a lot of attention in the literature because of its several cryptographic applications and its relations to computational complexity and program checking questions.

Although [1] and [2] introduced these ways how to prove an integer $M \in BL$ by zero-knowledge proofs, they have a common faulty: all integers M with the form $p^{k_1} q^{k_2}$ were proven to be Blum integer, however, it was not clear $M = pq$ or $M = p^{k_1} q^{k_2}$, where at least one of k_1 and k_2 is greater than one. In [3], B.Schneier said that **it is unknown whether there exists a true and practical zero-knowledge proof for** $M(= pq) \in BL$. Because Blum integers M with form pq are very important and extensively applied in domain of cryptography, it is urgent to find a feasible way to prove $M(= pq) \in BL$.

In this paper, we propose a method to prove $M(= pq) \in BL$ in statistical zero-knowledge proofs, at the same time, we also present the method for $M(= p^{k_1} q^{k_2}) \in BL$ with statistical zero-knowledge proofs, where at least one of k_1 and k_2 is greater than one. In particular, both of them are based on discrete logarithms, which are different from the previous ways based quadratic residues modulo composite numbers whose factorization is not given.

In [8], a number being the product of two safe primes is proven, however, this type number only are some special Blum integers, i.e., it requires that both of $(p-1)/2$ and $(q-1)/2$ are primes. For arbitrary Blum integers, its proof is infeasible, because $(p-1)/2$ and $(q-1)/2$ may not be prime, moreover, if $M = p^{k_1}q^{k_2}$, its proof will not be completely infeasible. However, Our method is feasible for arbitrary Blum integers. Blum integers are vastly applied in cryptography, and it must not only be this type Blum integers, i.e., the product of two safe primes, for example, K-P-W group signature scheme uses a usual Blum integers[17], hence, we think that our works are significant.

The structure of this paper is following, in section 2 we introduce some definitions and facts used in this paper. We review the methods in [1] and in[2] in section 3. In section 4 and section 5 we introduce our main results and prove these results. Finally, concluding remarks will be given in section 6.

2 Definitions and Facts

2.1 Number theory

Quadratic Residues. For each integer $x > 0$, the set of integers less than x and relatively prime to x form a group under multiplication modulo x denoted Z_x^*. We say that $y \in Z_x^*$ is a *quadratic residue* modulo x iff there is a $w \in Z_x^*$ such that $w^2 \equiv y \ mod\ x$. If this is not the case we call y a *quadratic nonresidue* modulo x. For compactness, we define the *quadratic residue predicate* as follows

$$Q_x(y) = \begin{cases} 0 & if\ y\ is\ a\ quadratic\ residue\ modulo\ \mathrm{x}\ and \\ 1 & otherwise \end{cases}$$

Moreover, we let J_x^{+1} and J_x^{-1} denote, respectively, the sets of elements of Z_x^* with Jacobi symbol $+1$ and -1 and $QR_x = \{y \in J_x^{+1} | Q_x(y) = 0\}$, $NQR_x = \{y \in J_x^{+1} | Q_x(y) = 1\}$.

Blum integers. We denote by $\mathcal{N}$ the set of natural numbers and P_{rime} the set of prime numbers. For $n \in \mathcal{N}$, we define the set of Blum integers of size n, $BL(n)$, as follows: $M \in BL(n)$ if and only if $M = p^{k_1}q^{k_2}$, where p and q are different primes both $\equiv 3\ mod\ 4$ and k_1 and k_2 are odd integers.

Regular integers. A Blum integer enjoys an elegant structural property, namely, $|J_x^{+1}| = |J_x^{-1}|$. More generally, we define an integer to be *regular* if it enjoys the above property. We define $Regular(s)$ to be the set of regular integers with s distinct prime divisors.

We get two facts by the above these definitions:

Fact 1 *[1]An integer M is a Blum integer if and only if $M \in regular(2)$, $-1 \; mod \; M \in NQR_M$, and for each $w \in QR_M$ there exists an r such that $r^4 \equiv w \; mod \; M$.*

Fact 2 *[2]If An integer M is a Blum integer, x is a quadratic residue modulo M, and b is 1 or -1, then x has a square root modulo M with Jacobi symbol b.*

Lehmann's primality test. An odd integer $n > 1$ is *prime* if and only if

$$\forall a \in Z_n^* : \;\; a^{(n-1)/2} \equiv \pm 1 \; (mod \; n) \; and \; \exists a \in Z_n^* : \;\; a^{(n-1)/2} \equiv -1 \; (mod \; n).$$

2.2 Zero-knowledge

In this section we review the formal definitions for the two types of zero-knowledge protocols that will be of interest in this paper: interactive zero-knowledge proof and non-interactive zero-knowledge proof.

Definition 1 *Let P be a probabilistic Turing machine and V a probabilistic polynomial-time Turing machine that share the same input and can communicate with each other. Let L be a language. We say that a pair (P, V) is a perfect(statistical, computational) zero-knowledge proof system for L if*

1. *(Completeness) For all $x \in L$,*

$$Prob[t \leftarrow (P, V)(x); V(x, t) = ACCEPT] = 1.$$

2. *(Soundness) For all $X \notin L$, and any Turing machine P', it holds that*

$$Prob[t \leftarrow (P', V)(x); V(x, t) = ACCEPT] \leq 1/2.$$

3. *(Perfect(statistical, computational zero-knowledge)) For any probability polynomial time algorithm V', there exists a polynomial time algorithm S, called the simulator, such that for all $x \in L$ the following holds:*

   - *$S_{V'}(x) = \perp$ with probability at most 1/2;*
   - *Conditioned on $S_{V'}(x) \neq \perp$, the two distributions $S_{V'}(x)$ and $View_{V'}(x) = \{(r, t) | t \leftarrow (P, V(r))(x)\}$ are perfect(statistical, computational) indistinguishable.*

Definition 2 *We say that (P, V) is a non-interactive perfect(statistical, computational) zero-knowledge proof system for the language L if there exists a positive constant c such that:*

1. *(Completeness) $\forall x \in L$, $|x| = n$ and for all sufficiently large n,*

$$Pr(\sigma \leftarrow 0, 1^{n^c}; Proof \leftarrow P(\sigma, x) : V(\sigma, x, Proof) = 1) > 1 - 2^{-n}.$$

2. *(Soundness) For all probabilistic algorithms Adversary outputting pairs $(x, Proof)$, where $x \notin L$, $|x| = n$, and all sufficiently large n,*

$$Pr(\sigma \leftarrow 0, 1^{n^c}; (x, Proof) \leftarrow Adversary(\sigma) : V(\sigma, x, Proof) = 1) < 2^{-n}.$$

3. *(Perfect(Statistical, Computational) zero-knowledge) There exists an efficient simulator algorithm S such that $\forall x \in L$, the two probability spaces $S(x)$ and $View_V(x)$ are perfect(statistical, computational) indistinguishable, where by $View_V(x)$ we denote the probability space*

$$View_V(x) = \{\sigma \leftarrow 0, 1^{|x|^c}; Proof \leftarrow P(\sigma, x) : (\sigma, Proof)\}.$$

2.3 Commitment schemes

Pederson[7] proposed a computationally binding and unconditionally hiding scheme based on the discrete logarithm problem. Given a group G of prime order q and two random generators g and h such that $\log_g h$ is unknown and computing discrete logarithms is infeasible. A value $\alpha \in Z_q$ is committed to as $C_\alpha := g^\alpha h^r$, where r is randomly chosen from Z_q. We will use this commitment scheme for our construction and hence they will be statistical zero-knowledge proof of knowledge.

2.4 Zero-knowledge proofs of knowledge about some modular relations

In this section, we mainly review some results from in [8, 11, 12]. Other zero-knowledge proofs of knowledge based on discrete logarithm are referred in [9, 10, 13, 14, 15, 16, 18],

2.4.1 proving that a discrete logarithm lies in a given range

A statistical zero-knowledge protocol proving that a discrete logarithm lies in a given range in [11, 12] was proposed and is denoted by

$$PK\{(\alpha) : y = g^\alpha \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}}\}.$$

2.4.2 Proving in statistical zero-knowledge that $a+b \equiv d(mod\ n)$, $ab \equiv d(mod\ n)$ and $a^b \equiv d(mod\ n)$ hold

Let l be an integer such that $-2^l < a, b, d, n < 2^l$ holds and $\varepsilon > 1$ be security parameters. Furthermore, we assume that a group G of order $q > 2^{2\varepsilon l+5} (= 2^{2\ddot{l}+1})$ and two generators g and h are available such that $log_g h$ is not known. This group could for instance be chosen by the prover in which case she would have to prove that she has chosen it correctly. Finally, let the prover's commitments to a, b, d and n be $c_a := g^a h^{r_1}, c_b := g^b h^{r_2}, c_d := g^d h^{r_3}$, and $c_n := g^m h^{r_4}$, where r_1, r_2, r_3, and r_4 are randomly chosen elements of Z_q.

Camenisch and Michels([8]) assume that the verifier has already obtained the commitments c_a, c_b, c_d, and c_n. Then the prover can convince the verifier that $a + b \equiv d(mod\ n)$ holds by running the protocol denoted:

$$\begin{aligned} S_+ := PK\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \eta, \vartheta, \varrho, \lambda) :& \\ c_a = g^\alpha h^\beta \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge c_b = g^\gamma h^\delta \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge& \\ c_d = g^\varepsilon h^\zeta \wedge -2^{\ddot{l}} < \varepsilon < 2^{\ddot{l}} \wedge c_n = g^\eta h^\vartheta \wedge -2^{\ddot{l}} < \eta < 2^{\ddot{l}} \wedge& \\ \tfrac{c_d}{c_a c_b} = c_n^\varrho h^\lambda \wedge -2^{\ddot{l}} < \varrho < 2^{\ddot{l}}\}& \end{aligned}$$

Alternatively, she can convince the verifier that $ab \equiv d(mod\ n)$ holds by running the protocol:

$$\begin{aligned} S_* := PK\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \eta, \vartheta, \xi, \rho, \sigma) :& \\ c_a = g^\alpha h^\beta \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge c_b = g^\gamma h^\delta \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge& \\ c_d = g^\varepsilon h^\zeta \wedge -2^{\ddot{l}} < \varepsilon < 2^{\ddot{l}} \wedge c_n = g^\eta h^\vartheta \wedge -2^{\ddot{l}} < \eta < 2^{\ddot{l}} \wedge& \\ c_d = c_b^\alpha c_n^\rho h^\sigma \wedge -2^{\ddot{l}} < \rho < 2^{\ddot{l}}\}.& \end{aligned}$$

At the same time, they presented a protocol in which the prover can convince the verifier that $a^b \equiv d(mod\ m)$ holds for the committed integers without revealing any further information. The protocol is denoted by S_{exp} and is referred in Appendix A. In the following, when denoting a protocol, we will abbreviate the protocol S_{exp} by a clause like to the statement that is proven and assume that the prover send the verifier all necessary commitments; e.g.,

$$PK\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \theta, \kappa) : c_a = g^\alpha h^\beta \wedge c_b = g^\gamma h^\delta \wedge c_d = g^\varepsilon h^\zeta \wedge$$

$$c_n = g^\theta h^\kappa \wedge (\alpha^\gamma \equiv \varepsilon(mod\ \theta))\}.$$

Theorem 1 *Let a, b, d, and m be integers that are committed to by the prover as described above, Then All three Potocols S_+, S_*, and S_{exp} are statistical zero-knowledge proofs that $a + b \equiv d(mod\ n)$, $ab \equiv d(mod\ n)$ and $a^b \equiv d(mod\ n)$ hold, respectively.*

2.4.3 proving the pseudo-primality of a committed number

In [8], J.Camenish and M.Michels show how the prover and the verifier can do Lehmann's primality test for a number committed by prover such that the verifier is convinced that the test was correctly done but does not learn any other information. The general idea is that the prover commits to s random bases a_i and then prove that for these bases $a_i^{(m-1)/2} \equiv \pm 1 (mod\, m)$ holds. Furthermore, the prover must commit to a base, say ã, such that $\tilde{a}^{(m-1)/2} \equiv -1(mod\, m)$ holds to satisfy the second condition in Lehmann's primality test. We call this protocol S_{prime} which is described in Appendix B. In the following section, $PK\{(\alpha, \beta) : c_a = g^\alpha h^\beta \wedge \alpha \in \{prime\})$ denotes proving that an integer a is a prime by S_{prime}.

Theorem 2 *Given a commitment c_m to an integer, the protocol S_{prime} is a statistical zero-knowledge proof that the committed integer is a prime with error-probability at most 2^{-s} for the primality-test.*

All described protocols can be combined in natural ways. First of all, one can use multiple bases instead of a single one in any of the above proofs. Then, executing any number of instances of these protocols in parallel and choosing the same challenges for all of them in each round corresponding to the $\wedge$-composition of the statements the single protocols prove.

3 Known Protocols Proving $M \in BL$

In this section, we review mainly the results in [1] and [2] and make some remarks. Protocol 1 and Protocol 2 comes from [2] and [1], respectively.

Protocol 1

1. P and V use the mutually trusted source of randomness to obtain 100 random numbers $\{x_i : i = 1, ..., 100\}$ in J_M^{+1} and 100 random signs $\{b_i : i = 1, ..., 100\}$ with $b_i \in \{-1, 1\}$.
2. for $i = 1$ to 100, P displays a square root r_i of x_i or of x_{-i} modulo M with Jacobi symbol equal to b_i.

Remark 1: According to Fact 2, P can convince V that $M \in BL$ holds, if P knows factorization of M. But, at the end V can not know that M has the form pq or the form $p^{k_1}q^{k_2}$, and he only know $M \in BL$. Protocol 1 is interactive computational zero-knowledge proof.

Protocol 2

1. It is sufficient for the prover to first prove that M is a $Regular(2)$ integer and that -1 is a quadratic non-residue modulo M using the proof system given in [4].

2. all it is left to prove is that every quadratic residue has a fourth root modulo x. This is done by giving, for each element $y \in J_M^{+1}$ taken from the random string, a fourth root modulo M of y or $-y$, depending on the quadratic residuosity of y.

Remarks 2: According to Fact 1, P can also convince V that $M \in BL$ holds; However, V does not know that M has the form pq or the form $p^{k_1}q^{k_2}$ in this protocol too. Protocol 2 is non-interactive perfect zero-knowledge proof.

In Protocol 1 and Protocol 2, V believes $M \in BL$ according Fact 2 and Fact 1, respectively. But there exists a common problem: V can not know form of M, because all Blum integers, not only $M = pq$ but also $M = p^{k_1}q^{k_2}$, have same properties which satisfy Fact 1 and Fact 2. **Hence it is proposed as an open problem whether there exists a truly practical zero-knowledge proof for $M(= pq) \in BL$[3].** We will deal with it in the following sections.

4 The statistical zero-knowledge proof for $a+b = d$, $ab = d$, and $d = a^b$

In [8], Camenisch and Michels obtained the statistical zero-knowledge proofs for $a + b \equiv d(mod\ n)$, $ab \equiv d(mod\ n)$, and $a^b \equiv d(mod\ n)$, however, the verifier gets only commitments to some integers without obtaining any further information in these protocols. Now, we will generalize their results and construct the statistical zero-knowledge proof for $a + b = d$, $ab = d$, and $d = a^b$, furthermore, the verifier also obtains nothing information except commitments to some integers.

Assume l, q and commitment scheme be uniform in 2.4.2, and the verifier gets commitments c_a, c_b, c_d to a, b, d, respectively. Then, in the following two protocols S'_+ and S'_* the prover can convince the verifier that $a + b = d$ and $ab = d$ hold.

$$
\begin{aligned}
S'_+ := PK\{&(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \lambda) : \\
&c_a = g^\alpha h^\beta \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \\
&c_b = g^\gamma h^\delta \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \\
&c_d = g^\varepsilon h^\zeta \wedge -2^{\ddot{l}} < \varepsilon < 2^{\ddot{l}} \wedge \\
&\tfrac{c_d}{c_a c_b} = h^\lambda\}
\end{aligned}
$$

$S'_* := PK\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \sigma) :$

$$
\begin{aligned}
&c_a = g^{\alpha} h^{\beta} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \\
&c_b = g^{\gamma} h^{\delta} \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \\
&c_d = g^{\varepsilon} h^{\zeta} \wedge -2^{\ddot{l}} < \varepsilon < 2^{\ddot{l}} \wedge \\
&c_d = c_b^{\alpha} h^{\sigma} \}
\end{aligned}
$$

The following protocol S'_{exp} will guarantee that the prover convinces the verifier that $a^b = d$ holds.

$$
\begin{gathered}
S'_{exp} := PK\{(\alpha, \beta, \xi, \chi, \gamma, \delta, \eta, (\lambda_i, \mu_i, \xi_i, \sigma_i, \tau_i, \vartheta_i, \psi_i)_{i=1}^{l_b-1}, (\omega_i, \rho_i)_{i=1}^{l_b-2},) : \\
c_a = g^{\alpha} h^{\beta} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \\
c_d = g^{\gamma} h^{\delta} \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \\
(\textstyle\prod_{i=0}^{l_b-1} c_{b_i}^{2^i})/c_b = h^{\eta} \wedge \\
c_{v_1} = g^{\lambda_1} h^{\mu_1} \wedge \ldots \wedge c_{v_{l_b-1}} = g^{\lambda_{l_b-1}} h^{\mu_{l_b-1}} \wedge \\
c_{v_1} = c_a^{\alpha} h^{\xi_1} \wedge c_{v_2} = c_{v_1}^{\lambda_1} h^{\xi_2} \wedge \ldots \wedge c_{v_{l_b-1}} = c_{\nu_{l_b-2}}^{\lambda_{l_b-2}} h^{\xi_{l_b-1}} \wedge \\
-2^{\ddot{l}} < \lambda_1 < 2^{\ddot{l}} \wedge \ldots \wedge -2^{\ddot{l}} < \lambda_{l_b-1} < 2^{\ddot{l}} \wedge \\
c_{\mu_1} = g^{\omega_1} h^{\rho_1} \wedge \ldots \wedge c_{\mu_{l_b-2}} = g^{\omega_{l_b-2}} h^{\rho_{l_b-2}} \wedge \\
-2^{\ddot{l}} < \omega_1 < 2^{\ddot{l}} \wedge \ldots \wedge -2^{\ddot{l}} < \omega_{l_b-2} < 2^{\ddot{l}} \wedge \\
((c_{b_0} = h^{\sigma_0} \wedge c_{\mu_0}/g = h^{\tau_0}) \vee (c_{b_0}/g = h^{\vartheta_0} \wedge c_{\mu_0}/c_a = h^{\psi_0})) \wedge \\
((c_{b_1} = h^{\sigma_1} \wedge c_{\mu_1}/c_{\mu_0} = h^{\tau_1}) \vee \\
(c_{b_1}/g = h^{\vartheta_1} \wedge c_{\mu_1} = c_{\mu_0}^{\lambda_1} h^{\psi_1})) \wedge \ldots \wedge \\
((c_{b_{l_b-2}} = h^{\sigma_{l_b-2}} \wedge c_{\mu_{l_b-2}}/c_{\mu_{l_b-3}} = h^{\tau_{l_b-2}}) \vee \\
(c_{b_{l_b-2}}/g = h^{\vartheta_{l_b-2}} \wedge c_{\mu_{l_b-2}} = c_{\mu_{l_b-3}}^{\lambda_{l_b-2}} h^{\psi_{l_b-2}})) \wedge \\
((c_{b_{l_b-1}} = h^{\sigma_{l_b-1}} \wedge c_d/c_{\mu_{l_b-2}} = h^{\tau_{l_b-1}}) \vee \\
(c_{b_{l_b-1}}/g = h^{\vartheta_{l_b-1}} \wedge c_d = c_{\mu_{l_b-2}}^{\lambda_{l_b-1}} h^{\psi_{l_b-1}}))\}
\end{gathered}
$$

Theorem 3 *Let a, b, and d be integers that are committed to by the prover as described above, Then All three Protocols S'_+, S'_*, and S'_{exp} are statistical zero-knowledge proofs that $a+b=d$, $ab=d=$ and $a^b=d$ hold, respectively.*

Proof: We explain mainly this reason that $a+b=d$ holds, however, the proofs of $ab=d$ and $a^b=d$ are omitted.

The statistical zero-knowledge claims follows from the statistical zero-knowledgeness of the building blocks.

Running the prover with this protocol and using standard techniques, the knowledge extractor can compute integers $\hat{a}, \hat{b}, \hat{d}, \hat{r_1}, \hat{r_2}, \hat{r_3}$ such that $c_a = g^{\hat{a}} h^{\hat{r_1}}$, $c_b = g^{\hat{b}} h^{\hat{r_2}}$, and $c_d = g^{\hat{d}} h^{\hat{r_3}}$ hold. Moreover, $-2^{\ddot{l}} < \hat{a} < 2^{\ddot{l}}$, $-2^{\ddot{l}} < \hat{b} < 2^{\ddot{l}}$, and $-2^{\ddot{l}} < \hat{d} < 2^{\ddot{l}}$, hold for these integers.

When running the prover with S'_+, the knowledge extractor can further compute integers $\hat{r_4} \in Z_q$ such that $c_d/(c_a c_b) = h^{\hat{r_4}}$ holds.

Therefore we have $g^{\hat{d}-\hat{a}-\hat{b}}h^{\hat{r_3}-\hat{r_1}-\hat{r_2}} = h^{\hat{r_5}}$ and hence, provided that the discrete log of h to the base g is not known, we must have

$$\hat{d} \equiv \hat{a} + \hat{b}(mod\ q).$$

Thus we have $\hat{d} = \hat{a} + \hat{b} + \bar{w}q$ for some integer $\bar{w}$. Since $2^{2\ddot{l}+1} < q$ and due to the constraints on $\hat{a}, \hat{b}, \hat{d}$ we can conclude that the integer $\bar{w}$ must be 0 and hence

$$\hat{d} = \hat{a} + \hat{b}$$

must hold. $\blacksquare$

In the following, when denoting a protocol, we will abbreviate the protocol S'_{exp} by a clause like to the statement that is proven and assume that the prover send the verifier all necessary commitments; e.g.,

$$PK\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta) : c_a = g^\alpha h^\beta \wedge c_b = g^\gamma h^\delta \wedge c_d = g^\varepsilon h^\zeta \wedge (\alpha^\gamma = \varepsilon)\}$$

Remarks: By using protocol S'_+, S'_*, we can construct a statistical zero-knowledge proof proving that a committed integer a is either odd or even.

5 A protocol proving that a committed integer $a \neq 0$ holds with statistical zero-knowledge

In this section, we will construct a protocol by which the prover can convince the verifier that an integer a is not 0, furthermore, it is statistical zero-knowledge.

For an arbitrary integer a, it can be written $\prod_{i=1}^{i=r} p_i^{k_i}$, where $p_1, ..., p_r$ are primes and $k_1, ..., k_r$ are integers. Now, if the prover can prove that a has form $\prod_{i=1}^{r} p_i^{k_i}$ and all $p_1, ..., p_r$ are primes, then $a \neq 0$ holds.

Assume l, q and commitment scheme be uniform in 2.4.2, and let prover's commitments to $a, s_1 = p_1^{k_1}, ..., s_r = p_r^{k_r}, p_1, ..., p_r, k_1, ..., k_r$, and suppose the verifier has already obtained all commitments before the protocol begins. The following protocol will prove that the integer a is not 0.

$S_{a\neq 0} := PK\{(\alpha, \beta, \rho, (\delta_i, \varepsilon_i, \zeta_i, \eta_i, \theta_i, \mu_i)_{i=1}^{i=r}) :$

$$c_a = g^{\alpha} h^{\beta} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \tag{1}$$
$$c_{s_1} = g^{\delta_1} h^{\varepsilon_1} \wedge ... \wedge c_{s_r} = g^{\delta_r} h^{\varepsilon_r} \wedge \tag{2}$$
$$(-2^{\ddot{l}} < \delta_1 < 2^{\ddot{l}}) \wedge ... \wedge (-2^{\ddot{l}} < \delta_r < 2^{\ddot{l}}) \wedge \tag{3}$$
$$c_a / c_{s_1} ... c_{s_r} = h^{\rho} \wedge \tag{4}$$
$$c_{p_1} = g^{\zeta_1} h^{\eta_1} \wedge ... \wedge c_{p_r} = g^{\zeta_r} h^{\eta_r} \wedge \tag{5}$$
$$(-2^{\ddot{l}} < \zeta_1 < 2^{\ddot{l}}) \wedge ... \wedge (-2^{\ddot{l}} < \zeta_r < 2^{\ddot{l}}) \wedge \tag{6}$$
$$c_{k_1} = g^{\theta_1} h^{\mu_1} \wedge ... \wedge c_{p_r} = g^{\theta_r} h^{\mu_r} \wedge \tag{7}$$
$$(-2^{\ddot{l}} < \theta_1 < 2^{\ddot{l}}) \wedge ... \wedge (-2^{\ddot{l}} < \theta_r < 2^{\ddot{l}}) \wedge \tag{8}$$
$$(\delta_1 = \zeta_1^{\theta_1}) \wedge ... \wedge (\delta_r = \zeta_r^{\theta_r}) \wedge \tag{9}$$
$$\zeta_1 \in \{prime\} \wedge ... \wedge \zeta_r \in \{prime\}\} \tag{10}$$

Theorem 4 *Let a be an integer that is committed by c_a. Then $S_{a \neq 0}$ is a statistical zero-knowledge proof that $a \neq 0$ holds.*

Proof: *Completeness:* If $a \neq 0$, the prover can prove that $a = \prod_{i=1}^{r} p_i^{k_i}$ holds in (1)-(9); in (10), the prover proves that all of $p_1, ..., p_r$ are prime numbers. As a result, the verifier believes that $a \neq 0$ holds .

Soundness: If $a = 0$, the prover may prove that a is a composite integer in (1)-(9); however, she can not prove that each of $p_1, ..., p_r$ is prime; so, the verifier rejects.

Zero-knowledgeness: $S_{a \neq 0}$ is statistical zero-knowledge from Theorem 1, 2, and 3. $\blacksquare$

Remark: From the above protocol, we can prove that an integer b is not equal to another integer d. We first commit to b, d, and $a = d - b$; then prove that $d = a + b$ holds from S'_+; finally, we prove that $S_{d \neq 0}$ from the above protocol.

6 The Protocol Proving $M \in BL$

According to $S_{a \neq 0}$ and this protocols in 2.4.2, we propose our protocol which can prove either that Blum integer M has the form pq or that Blum integer M has the form $p^{k_1} q^{k_2}$ in this section, where at least one of k_1 and k_2 is greater than one.

6.1 Initialization

Let l, M be two integers such that $-2^l < M < 2^l$, $\varepsilon > 1$ be security parameters. Furthermore, we assume that a group G of prime order $s > 2^{2\varepsilon l+5}(= 2^{2\ddot{l}+1})$ and two generates g and h are available such that $log_g h$ is not known. Finally, we select the following commitment: in which integer a is committed $c_a := g^a h^r$, where r is randomly chosen elements of Z_s. According to this commitment, let commitments to M be c_M.

6.2 The protocol proving $M \in BL$ and $M = pq$

If $M \in BL$ and $M = pq$, prover commits to $M, p, q, a = p - q, p_1 = (p-3)/4, q_1 = (q-3)/4$ and sends all commitments $c_M, c_p, c_q, c_a, c_{p_1}, c_{q_1}$ to verifier. In the following protocol S_{M1}, the prover can convince the verifier $M \in BL$ and $M = pq$.

$S_{M1} := PK\{(\alpha, \bar{\alpha}, \beta, \bar{\beta}, \gamma, \bar{\gamma}, \lambda, \pi, \bar{\pi}, \upsilon, \zeta, \eta, \mu, \bar{\mu}, \nu, \bar{\nu}):$

$$c_M = g^{\alpha} h^{\bar{\alpha}} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \tag{11}$$

$$c_p = g^{\beta} h^{\bar{\beta}} \wedge -2^{\ddot{l}} < \beta < 2^{\ddot{l}} \wedge \tag{12}$$

$$c_q = g^{\gamma} h^{\bar{\gamma}} \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \tag{13}$$

$$c_M/(c_p^{\gamma}) = h^{\lambda} \wedge \tag{14}$$

$$\beta \in \{prime\} \wedge \gamma \in \{prime\} \wedge \tag{15}$$

$$c_a = g^{\pi} h^{\bar{\pi}} \wedge -2^{\ddot{l}} < \pi < 2^{\ddot{l}} \wedge \tag{16}$$

$$c_p/c_a c_q = h^{\upsilon} \wedge \tag{17}$$

$$S_{\pi \neq 0} \wedge \tag{18}$$

$$c_{p_1} = g^{\mu} h^{\bar{\mu}} \wedge -2^{\ddot{l}} < \mu < 2^{\ddot{l}} \wedge \tag{19}$$

$$c_{q_1} = g^{\nu} h^{\bar{\nu}} \wedge -2^{\ddot{l}} < \nu < 2^{\ddot{l}} \wedge \tag{20}$$

$$c_p/c_{p_1}^4 g^3 = h^{\zeta} \wedge c_q/c_{q_1}^4 g^3 = h^{\eta}\} \tag{21}$$

Remark: in this protocol, (11)-(14) proves $M = pq$; (15) tests that p and q are primes; (16)-(18) proves $p \neq q$, (19)-(21) proves $p \equiv 3(mod\, 4)$ and $q \equiv 3(mod\, 4)$.

6.3 The protocol proving $M \in BL$ and $M = p^{k_1} q^{k_2}$

If $M \in BL$ and $M = p^{k_1} q^{k_2}$, where at least one of k_1 and k_2 is not equal to one, prover commits to $M, P = p^{k_1}, Q = q^{k_2}, p, q, a = p - q, k_1, k_2, p_1 =$

$(p-3)/4, q_1 = (q-3)/4, m_1 = (k_1-1)/2, m_2 = (k_2-1)/2$, and sends all commitments $c_M, c_P, c_Q, c_p, c_q, c_a, c_{k_1}, c_{k_2}, c_{p_1}, c_{p_2}, c_{m_1}, c_{m_2}$ to verifier. In the following protocol S_{M2}, the prover can convince the verifier $M \in BL$ and $M = p^{k_1}q^{k_2}$.

$$S_{M2} := PK\{(\alpha, \bar{\alpha}, \alpha_1, \bar{\alpha_1}, \beta, \bar{\beta}, \beta_1, \bar{\beta_1}, \gamma, \bar{\gamma}, \lambda, \psi, \bar{\psi}, \omega, \bar{\omega}, \varsigma, \bar{\varsigma}, \tau, \bar{\tau}, \delta, \varepsilon,$$
$$\pi, \bar{\pi}, \upsilon, \mu, \bar{\mu}, \nu, \bar{\nu}, \rho, \varrho) :$$

$$c_M = g^{\alpha}h^{\bar{\alpha}} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \tag{22}$$
$$c_P = g^{\beta}h^{\bar{\beta}} \wedge -2^{\ddot{l}} < \beta < 2^{\ddot{l}} \wedge \tag{23}$$
$$c_Q = g^{\gamma}h^{\bar{\gamma}} \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \tag{24}$$
$$c_M/(c_P^{\gamma}) = h^{\lambda} \wedge \tag{25}$$
$$c_p = g^{\psi}h^{\bar{\psi}} \wedge -2^{\ddot{l}} < \psi < 2^{\ddot{l}} \wedge \tag{26}$$
$$c_q = g^{\omega}h^{\bar{\omega}} \wedge -2^{\ddot{l}} < \omega < 2^{\ddot{l}} \wedge \tag{27}$$
$$c_{k_1} = g^{\varsigma}h^{\bar{\varsigma}} \wedge -2^{\ddot{l}} < \varsigma < 2^{\ddot{l}} \wedge \tag{28}$$
$$c_{k_2} = g^{\tau}h^{\bar{\tau}} \wedge -2^{\ddot{l}} < \tau < 2^{\ddot{l}} \wedge \tag{29}$$
$$\beta = \psi^{\varsigma} \wedge \gamma = \omega^{\tau} \wedge \tag{30}$$
$$\psi \in \{prime\} \wedge \omega \in \{prime\} \wedge \tag{31}$$
$$c_a = g^{\pi}h^{\bar{\pi}} \wedge -2^{\ddot{l}} < \pi < 2^{\ddot{l}} \wedge \tag{32}$$
$$c_p/c_a c_q = h^{\upsilon} \wedge \tag{33}$$
$$S_{\pi \neq 0} \wedge \tag{34}$$
$$c_{p_1} = g^{\mu}h^{\bar{\mu}} \wedge -2^{\ddot{l}} < \mu < 2^{\ddot{l}} \wedge \tag{35}$$
$$c_{q_1} = g^{\nu}h^{\bar{\nu}} \wedge -2^{\ddot{l}} < \nu < 2^{\ddot{l}} \wedge \tag{36}$$
$$c_p/(c_{p_1}^4 g^3) = h^{\rho} \wedge c_q/(c_{q_1}^4 g^3) = h^{\varrho} \wedge \tag{37}$$
$$c_{m_1} = g^{\alpha_1}h^{\bar{\alpha_1}} \wedge -2^{\ddot{l}} < \alpha_1 < 2^{\ddot{l}} \wedge \tag{38}$$
$$c_{m_2} = g^{\beta_1}h^{\bar{\beta_1}} \wedge -2^{\ddot{l}} < \beta_1 < 2^{\ddot{l}} \wedge \tag{39}$$
$$c_{k_1}/(c_{m_1}^2 g) = h^{\delta} \wedge c_{k_2}/(c_{m_2}^2 g) = h^{\varepsilon} \wedge \tag{40}$$
$$(S_{\alpha_1 \neq 0} \vee S_{\beta_1 \neq 0})\} \tag{41}$$

Remark: in this protocol, (22)-(25) proves $M = PQ$; (26)-(30) proves $P = p^{k_1}$ and $Q = q^{k_2}$; (31) tests that p and q are primes; (32)-(34) proves $p \neq q$; (35)-(37) proves $p \equiv 3(mod\, 4)$ and $q \equiv 3(mod\, 4)$; (38)-(41) proves that k_1 and k_2 are odd numbers and at least one of them is greater than 1.

6.4 Two theorems

Theorem 5 *If M is a Blum integer with form pq, and M, p, q are committed to by the prover as described in S_{M1}. Then the protocol S_{M1} is a statistical zero-knowledge proof that $M \in BL$ and $M = pq$ hold.*

Proof:

1) *completeness:* If $M \in BL$ and $M = pq$, then, the verifier can believe that M is product of two integers p and q in (11)-(14); in (15), the protocol can convince the verifier that both of two integers are prime; the prover proves that primes p is not equal to q (16)-(18)in; and it is proven that q and p have form $4r + 3$, where r is an integer, in (19)-(21). Hence, if M is a Blum integer and it has form pq, the verifier can believe that $M \in BL$ and $M = pq$ hold.

2) *soundness:* If $M \in BL$, but $M = p^{k_1}q^{k_2}$, where p and q are different primes both $\equiv 3 \; mod \; 4$, k_1 and k_2 are odd integers, and at least one of k_1 and k_2 is more than 1. Obviously, in (11)-(14), the verifier can believe that M is product of two integers p' and q', however, (15) must not hold because at least one of p' and q' is not prime.

If $M \notin BL$:

1. if M is a prime, it is rejected in (11)-(14).
2. if M is a composite integer, and consists of two same prime factors, it is rejected in (16)-(18).
3. if M is a composite integer, and consists of two different prime factors p and q. If at least one of $p \equiv 3 \; mod \; 4$ and $q \equiv 3 \; mod \; 4$ does not hold, it is rejected in (19)-(20).
4. if M is a composite integer, and consists of two factors p and q, however, at least one of p and q is not prime, it is rejected in (5).

Hence, if $M \notin BL$ or $M \in BL$ with form $p^{k_1}q^{k_2}$, the prover can not convince the verifier that $M \in BL$ and $M = pq$ hold.

3) *zero-knowledge:* because $S_{a \in Prime}$ is a statistical zero-knowledge with error-probability at most 2^{-k}, $S_{M=pq}$ is a statistical zero-knowledge with error-probability at most 2^{-k} from theorem 2, and 3, where parameter k is the number of random number selected by prover in primality tests. ∎

Theorem 6 *If M is a Blum integer with form $p^{k_1}q^{k_2}$, and commitment scheme is used by the prover as described in S_{M2}. Then the protocol S_{M2} is* *a statistical zero-knowledge proof that $M \in BL$ and $M = p^{k_1}q^{k_2}$ hold. where p and q are different primes both $\equiv 3$ mod 4, k_1 and k_2 are odd integers, and at least one of them is greater than 1.*

Proof:

1) *completeness:* If $M \in BL$ and $M = p^{k_1}q^{k_2}$, then, the verifier believes that M is product of two integers P and Q in (22)-(25); in (26)-(30), the prover proves that $P = p^{k_1}$ and $Q = q^{k_2}$ hold; the prover convinces the verifier that p and q are prime numbers in (31), and p and q are different primes both $\equiv 3$ *mod* 4 in (32)-(37); in (38)-(41), the prover proves that k_1 and k_2 are odd numbers and at least one of them is greater than 1. As a result, the verifier accepts that $M \in BL$ and $M = p^{k_1}q^{k_2}$.

2) *soundness:* If $M \in BL$, but $M = pq$, where p and q are different primes both $\equiv 3$ *mod* 4. Obviously, in (38)-(41), the verifier can reject that M has form $M = p^{k_1}q^{k_2}$ because both of k_1 and k_2 are equal to 1.

If $M \notin BL$:

1. if M is a prime, it is rejected in (22)-(25).
2. if M is a power k of a prime p, i.e., $M = p^k$, it is rejected in (32)-(34).
3. if M is a composite integer, and has the form $p_1^k q_2^k$, where p and q are two different primes, but, at least one of $p \equiv 3$ *mod* 4 and $q \equiv 3$ *mod* 4 does not hold, it is rejected in (35)-(37).
4. if M is a composite integer, and consists of two prime factors p and q, where p and q are different primes both $\equiv 3$ *mod* 4, however, at least one of k_1 and k_2 is even, it is rejected in (38)-(40).
5. if M is a composite integer, and consists of at least three prime factors, then, $M \in BL$ will be rejected in (22)-(31).

Hence, if $M \notin BL$ or $M \in BL$ with form pq, the prover can not convince the verifier that $M \in BL$ and $M = p^{k_1}q^{k_2}$ hold.

3) *zero-knowledge:* Our protocol only uses basic protocols S_+, S_*, S_{exp}, $S_{p \in prime}$ and $S_{a \neq 0}$, however, they all are statistical zero-knowledge proofs, hence, our protocol is a statistical zero-knowledge proof. ∎

7 Conclusion

In this paper, we mainly propose three statistical zero-knowledge protocols, the first is to prove that an integer a does not equal to 0, the second is to prove $M \in BL$ with form pq, and the third is to prove $M \in BL$ with form $p^{k_1}q^{k_2}$, where at least one of k_1 and k_2 is greater than one.

References

[1] A.D.Santis, G.G.Crescenzo, and G.Persiano, Secret Sharing and Perfect Zero-knowledge, *Advances in Cryptology-CRYPTO'93*, pp.73-84, Berlin: Springer, 1994.

[2] J.V.D.Graaf, and R.Peralta, A simple and secure way to show the validity of your public key. *Advances in Cryptology- CRYPTO'87*, pp.128-134, Berlin: Springer, 1987.

[3] B.Schneier. Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C. *John Wiley* & *Son, Inc.*, New York et al, 1996.

[4] M.Blum, A.D.Santis, S.Micali, and G.Persiano, Non-interactive zero-knowledge, *SIAM J. Comput*, 20(6), pp.1084-1118, 1991.

[5] M.Blum, Coin Flipping by Telephone, *IEEE COMPCON*, pp.133-137, 1982.

[6] S.Goldwasser, S.Micali, and C.Rackoff, The Knowledge Complexity of Interactive Proof Systems. *SIAM J. Comput*, 18, pp.186-208, 1989.

[7] T.P Pedersen, Non-interactive and information-theoretic secure verifiable secret sharing. *Advances in Cryptology-CRYPTO'91*, pp 129-140, Berlin: Springer, 1991.

[8] J.Camenisch, M.Michels, Proving in Zero-knowledge that a Number is the Product of Two Safe Primes. *Advances in Cryptology-EUROCRYPT'99*, pp 106-121, Berlin: Springer, 1999.

[9] D Chaum, J.H Evertse, and J van de Graaf, and Peralta R, Demonstrating possession of a discrete logarithm without revealing it. *Advances in Cryptology-CRYPTO'86*, pp 200-212, Berlin: Springer, 1987.

[10] C.P Schnorr, Efficient signature generation for smart cards. *J of Cryptology*, 4(3):239-252, Berlin: Springer, 1991.

[11] A Chan, Y Frankel, and Y Tsiounis, Easy come-easy go divisible cash. *Advances in Cryptology-EUROCRYPT'98*, pp 561-575, Berlin: Springer, 1998.

[12] E Fujisaki, and T Okamoto, Statistical zero-knowledge protocols to prove modular polynomial relations. *Advances in Cryptology-CRYPTO'97*, pp 16-30, Berlin: Springer, 1997.

[13] S Brands, Electronic cash systems based on the representation problem in groups of prime order, *Advances in Cryptology-CRYPTO'93*, pp 1-15, Berlin: Springer, 1993.

[14] D Chaum, J.E Evertse, and J van de Graaf, An improved protocol for demonstrating possession of discrete logarithms and some generalizations, *Advances in Cryptology-EUROCRYPT'87*, pp 127-141, Berlin: Springer, 1988.

[15] D Chaum, and T.P Pedersen, Wallet databases with observers, *Advances in Cryptology-CRYPTO'92*, pp 89-105, Berlin: Springer, 1993.

[16] R Cramer, I Damgard, and B Schoenmakers, Proofs of partial knowledge and simplified design of witness hiding protocols, *Advances in Cryptology-CRYPTO'94*, pp 174-187, Berlin: Springer, 1994.

[17] S.J Kim, S.J Park , and D.H Won, Convertible group signatures. *Advances in cryptology-ASIACRYPT'96*, pp 311-321, Berlin: Springer, 1996.

[18] J Camenisch, and M Stadler, Efficient group signature schemes for large groups. *Advances in Cryptology-CRYPTO'97*, pp 410-424, Berlin: Springer, 1997.

Appendix A

This Protocol will prove that $a^b \equiv d(mod\ n)$ holds.

$S_{exp} := PK\{(\alpha, \beta, \xi, \chi, \gamma, \delta, \varepsilon, \zeta, \eta, (\lambda_i, \mu_i, \nu_i, \xi_i, \sigma_i, \tau_i, \vartheta_i, \varphi_i, \psi_i)_{i=1}^{l_b-1}, (\omega_i, \rho_i)_{i=1}^{l_b-2},) :$

$$
\begin{gathered}
c_a = g^{\alpha} h^{\beta} \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \\
c_d = g^{\gamma} h^{\delta} \wedge -2^{\ddot{l}} < \gamma < 2^{\ddot{l}} \wedge \\
c_n = g^{\varepsilon} h^{\zeta} \wedge -2^{\ddot{l}} < \varepsilon < 2^{\ddot{l}} \wedge \\
(\textstyle\prod_{i=0}^{l_b-1} c_{b_i}^{2^i})/c_b = h^{\eta} \wedge \\
c_{v_1} = g^{\lambda_1} h^{\mu_1} \wedge \ldots \wedge c_{v_{l_b-1}} = g^{\lambda_{l_b-1}} h^{\mu_{l_b-1}} \wedge \\
c_{v_1} = c_a^{\alpha} c_n^{\nu_1} h^{\xi_1} \wedge c_{v_2} = c_{v_1}^{\lambda_1} c_n^{\nu_2} h^{\xi_2} \wedge \ldots \wedge c_{v_{l_b-1}} = c_{v_{l_b-2}}^{\lambda_{l_b-2}} c_n^{\nu_{l_b-1}} h^{\xi_{l_b-1}} \wedge \\
-2^{\ddot{l}} < \lambda_1 < 2^{\ddot{l}} \wedge \ldots \wedge -2^{\ddot{l}} < \lambda_{l_b-1} < 2^{\ddot{l}} \wedge \\
-2^{\ddot{l}} < \nu_1 < 2^{\ddot{l}} \wedge \ldots \wedge -2^{\ddot{l}} < \nu_{l_b-1} < 2^{\ddot{l}} \wedge \\
c_{\mu_1} = g^{\omega_1} h^{\rho_1} \wedge \ldots \wedge c_{\mu_{l_b-2}} = g^{\omega_{l_b-2}} h^{\rho_{l_b-2}} \wedge \\
-2^{\ddot{l}} < \omega_1 < 2^{\ddot{l}} \wedge \ldots \wedge -2^{\ddot{l}} < \omega_{l_b-2} < 2^{\ddot{l}} \wedge \\
((c_{b_0} = h^{\sigma_0} \wedge c_{\mu_0}/g = h^{\tau_0}) \vee (c_{b_0}/g = h^{\vartheta_0} \wedge c_{\mu_0}/c_a = h^{\psi_0})) \wedge \\
((c_{b_1} = h^{\sigma_1} \wedge c_{\mu_1}/c_{\mu_0} = h^{\tau_1}) \vee \\
(c_{b_1}/g = h^{\vartheta_1} \wedge c_{\mu_1} = c_{\mu_0}^{\lambda_1} c_n^{\varphi_1} h^{\psi_1} \wedge -2^{\ddot{l}} < \varphi_1 < 2^{\ddot{l}})) \wedge \ldots \wedge \\
((c_{b_{l_b-2}} = h^{\sigma_{l_b-2}} \wedge c_{\mu_{l_b-2}}/c_{\mu_{l_b-3}} = h^{\tau_{l_b-2}}) \vee \\
(c_{b_{l_b-2}}/g = h^{\vartheta_{l_b-2}} \wedge c_{\mu_{l_b-2}} = c_{\mu_{l_b-3}}^{\lambda_{l_b-2}} c_n^{\varphi_{l_b-2}} h^{\psi_{l_b-2}} \wedge -2^{\ddot{l}} < \varphi_{l_b-2} < 2^{\ddot{l}})) \wedge \\
((c_{b_{l_b-1}} = h^{\sigma_{l_b-1}} \wedge c_d/c_{\mu_{l_b-2}} = h^{\tau_{l_b-1}}) \vee \\
(c_{b_{l_b-1}}/g = h^{\vartheta_{l_b-1}} \wedge c_d = c_{\mu_{l_b-2}}^{\lambda_{l_b-1}} c_n^{\varphi_{l_b-1}} h^{\psi_{l_b-1}} \wedge -2^{\ddot{l}} < \varphi_{l_b-1} < 2^{\ddot{l}}))\}
\end{gathered}
$$

Appendix B

The following protocol will prove that n is a prime.

1. The prover picks random $\hat{a}_i \in_R Z_n$ for $i = 1, ..., t$ and commits to them as $c_{\hat{a}_i} = g^{\hat{a}_i} h^{r_{\hat{a}}}$ with $r_{\hat{a}} \in_R Z_Q$ for $i = 1, ..., t$. She sends $c_{\hat{a}_1}, ..., c_{\hat{a}_t}$ to the verifier.

2. The verifier picks random integers $-2^l < \breve{a}_i < 2^l$ for $i = 1, ..., t$ and sends them to the prover.

3. The prover computes $a_i := \hat{a}_i + \breve{a}_i (mod\, n)$, $c_{a_i} := g^{a_i} h^{r_{a_i}}$ with $r_{a_i} \in_R Z_Q$, $d_i := a_i^{(n-1)/2} (mod\, n)$, and $c_{d_i} := g^{d_i} h^{r_{d_i}}$ with $r_{d_i} \in_R Z_Q$ for all $i = 1, ..., t$. Moreover, the prover commits to $(n-1)/2$ by $c_b := g^{(n-1)/2} h^{r_b}$ with $r_b \in_R Z_Q$. Then the prover searches a base $\tilde{a}$ such that $\tilde{a}^{(n-1)/2} \equiv -1 (mod\, n)$ holds and commits to $\tilde{a}$ by $c_{\tilde{a}} := g^{\tilde{a}} h^{r_{\tilde{a}}}$ with $r_{\tilde{a}} \in_R Z_Q$.

4. The prover sends $c_b, c_{\tilde{a}}, c_{a_1}, ..., c_{a_t}, c_{d_1}, ..., c_{d_t}$ to the verifier and then they carry out the following protocol.

$$
\begin{gathered}
S_{prime} := PK\{(\alpha, \beta, \gamma, \nu, \xi, \rho, \kappa, (\delta_i, \varepsilon_i, \zeta_i, \eta_i, \vartheta_i, \omega_i, \rho_i, \kappa_i, \mu_i, \psi_i)_{i=1}^t : \\
c_b = g^\alpha h^\beta \wedge -2^{\ddot{l}} < \alpha < 2^{\ddot{l}} \wedge \\
c_n = g^\nu h^\xi \wedge -2^{\ddot{l}} < \nu < 2^{\ddot{l}} \wedge \\
c_b^2 g / c_n = h^\gamma \wedge \\
c_{\tilde{a}} = g^\rho h^\kappa \wedge (\rho^\alpha \equiv -1 (mod\, \nu)) \wedge \\
c_{\hat{a}_1} = g^{\delta_1} h^{\varepsilon_1} \wedge ... \wedge c_{\hat{a}_t} = g^{\delta_t} h^{\varepsilon_t} \wedge \\
c_{a_1} / g^{\breve{a}_1} = g^{\delta_1} c_n^{\zeta_1} h^{\eta_1} \wedge ... \wedge c_{a_t} / g^{\breve{a}_t} = g^{\delta_t} c_n^{\zeta_t} h^{\eta_t} \wedge \\
-2^{\ddot{l}} < \delta_1 < 2^{\ddot{l}} \wedge ... \wedge -2^{\ddot{l}} < \delta_t < 2^{\ddot{l}} \wedge \\
-2^{\ddot{l}} < \zeta_1 < 2^{\ddot{l}} \wedge ... \wedge -2^{\ddot{l}} < \zeta_t < 2^{\ddot{l}} \wedge \\
c_{a_1} = g^{\rho_1} h^{\kappa_1} \wedge ... \wedge c_{a_t} = g^{\rho_t} h^{\kappa_t} \wedge \\
(c_{d_1} / g = h^{\vartheta_1} \vee c_{d_1} g = h^{\vartheta_1}) \wedge ... \wedge (c_{d_t} / g = h^{\vartheta_t} \vee c_{d_t} g = h^{\vartheta_t}) \wedge \\
c_{d_1} = g^{\mu_1} h^{\psi_1} \wedge ... \wedge c_{d_t} = g^{\mu_t} h^{\psi_t} \wedge \\
(\rho_1^\alpha \equiv \mu_1 \ (mod\, \nu)) \wedge ... \wedge (\rho_t^\alpha \equiv \mu_t \ (mod\, \nu))\}
\end{gathered}
$$