TD - Quadratic residue - Zero-knowledge protocol

$a \neq 0$ is a *square* (or *quadratic residue*) modulo b iff it exists x such that $x^2 \equiv a \mod b$. We say that x is a *square root* of a modulo b.
In the sequel, p and q are two odd distinct prime numbers and $n = p.q$.

1. Number of squares in $\mathbb{Z}/n\mathbb{Z}^\star$

a. Verify that if $x^2 \equiv a \mod b$, then $(b-x)^2 \equiv a \mod b$.

b. Prove that if a is a square modulo n, then a is a square mod p and mod q too.

c. Proove that any square $a \neq 0$ modulo p has exactly 2 roots : x and $y = p - x$.

d. Deduce that any square a in $\mathbb{Z}/n\mathbb{Z}$ relatively prime to p and q has exactly four distinct square roots: x_1, $n - x_1$, x_2 and $n - x_2$. **Hint:** use Chinese remainder theorem.

e. By using the property that $(\mathbb{Z}/p\mathbb{Z}^\star, \times)$ is a cyclic group, prove that there are $\frac{p-1}{2}$ non zero squares modulo p.

f. Deduce the number of squares in $\mathbb{Z}/n\mathbb{Z}^\star$.

a. $(b-x)^2 = b^2 - 2bx + x^2 = x^2 = a \mod b$.

b. $a = x^2 + kpq$; thus $a \equiv x^2 \mod p$ is a square mod p (similarly for q).

c. Let $x \neq y$ such that $a = x^2 = y^2 \mod p$. Then $x^2 - y^2 = (x-y)(x+y) = 0 \mod p$. But $\mathbb{Z}/p\mathbb{Z}$ is a field (since p is prime): there are no zero divisor. Thus, since $x - y \neq 0 \mod p$, necessarily $x + y = 0 \mod p$; therefore $y = p - x$.

d. From b., any square $a \mod n$ is a square both $\mod p$ and $\mod q$. Since $a \neq 0 \mod p$, a has exactly two distinct roots $u_1 = u$ and $u_2 = p - u$ modulo p (resp. $v_1 = v$ and $v_2 = q - v$ modulo q). From Chinese remainder theorem, this defines exactly 4 distinct roots for a mod n: $u_i.q.q^{-1[p]} + v_j.p.p^{-1[q]} \mod n$ with $1 \leq i, j \leq 2$.
From a., those roots can be expressed as x_1, $n - x_1$, x_2 et $n - x_2$.

e. Since p is prime, $(\mathbb{Z}/p\mathbb{Z}^\star, .)$ is cyclic; let g a primitive root (generator). Assume there exists x such that $g = x^2$. From Fermat theorem, $g^{p-1} = 1 \mod p$; since g is a primitive root and p odd, we have $g^{\frac{p-1}{2}} \mod p = -1 \mod p = p - 1 \mod p$. Then $x^{p-1} = -1 \neq 1$ since $p \neq 2$ and therefore, yet from Fermat theorem, $x \notin \mathbb{Z}/p\mathbb{Z}$. Thus g is not a square modulo p.
We deduce that the only non zero squares are the $\frac{p-1}{2}$ elements of the form g^{2i} for $1 \leq i \leq \frac{p-1}{2}$.
NB: g^{2i} has exactly two distinct square roots mod p: $x = g^i$ and $g^{i+\frac{p-1}{2}} = -x = p - x$.

f. Let g a primitive root in $(\mathbb{Z}/p\mathbb{Z}^\star, .)$ which is cyclic. From Chinese remainder theorem, each couple of squares $(u, v) \in \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/q\mathbb{Z}$ corresponds to exactly a unique square in $\mathbb{Z}/n\mathbb{Z}$. Including 0, there are exactly $\frac{p+1}{2}$ squares modulo p. Thus, we have $\frac{(p+1)(q+1)}{4} = \frac{n+p+q+1}{4}$ squares modulo n; thus the number of non zero squares modulo n is $\frac{n+p+q+1}{4} - 1 = \frac{n+p+q-3}{4}$.

2. Intractability of computing square roots. Let $a < n$; the goal of this question is to prove that computing square roots x of $a \neq 0$ modulo n is (polynomially) more expensive than factorization of n. The proof is performed by reduction (contradiction proof).
In all this question, it is assumed that we know the four distinct roots x_1, x_2, $(n-x_1)$ et $(n-x_2)$ of a modulo n; we prove that then that the factors p and q of n can be quicly computed.

a. Let $u = x_1 - x_2 \mod n$ and $v = x_1 + x_2 \mod n$. Prove that $u.v \equiv 0 \mod n$.

b. Justify that $1 \leq u, v < n$; then explicit how to compute p and q from u and v.

c. Give an upper bound on the number of operations performed (Big O notation) with respect to the number of bits of n.

d. Argue that the function *Square* of $\mathbb{Z}/n\mathbb{Z}$ defined by $Square(x) = x^2 \mod n$ may be considered as a one-way function.

a. $u.v = x_1^2 - x_2^2 = a^2 - a^2 = 0 \mod n$.

b. We can suppose $1 \leq x_1, x_2 < n$. Since $x_1 \neq x_2$, $u = x_1 - x_2 \neq 0$. Since $x_1 \neq n - x_2$, $v = x_1 + x_2 = x_1 - (n - x_2) \neq 0$. Therefore $1 \leq u, v < n$.
Now we have $u.v = k.n$; also $n = p.q$ divides $u.v$. Since p and q are primes and $u < p.q$, then p divides u or q divides u but $p.q$ does not divide u. Then $\gcd(n, u)$ returns one of the two factors of n; the other factor is $n/pgcd$(n,u).

c. Let $t = \log_2 n$ the size -number of bits- of n. The previous computation consists in two additions, a gcd and a division. The cost is dominated by the one of the gcd, which, by Euclid's algorithm is $O(t^2)$ (or $O(t \log^2 t \log\log t) = \tilde{O}(t)$ by Schonhagge's algorithm).

d. Computing $x^2 \mod n$ is performed efficiently in $O(t^2)$ ($\tilde{O}(t)$ using a fast integer multiplication algorithm). However, computing x from x^2 is polynomially more difficult then factorization: indeed, if we can compute the square roots of $a \mod n$, we can compute p and q in $O(t^2)$ as stated above. Then, under the conjecture (commonly considered at this time) that integer factorization is a computationally impossible problem, *Square* is a one-way function.

3. Quadratic authentication protocol. Let $n = pq$ an integer of 1024 bits with p and q large primes; p and q are known by a trusted third part TTP, but, a priori, not by Alice not Bob.
To authenticate to Bob, Alice chooses the integer $x_A < n$ as unique private key. Let $a = x_A^2 \mod n$; TTP delivers to Alice a passport one which are written the public integers n and a.

a. We assume that only Alice (and may be TTP) knows x_A and that nobody, except TTP, can compute square roots modulo n. Is this reasonable ?

b. To authenticate Alice, Bob reads a and n from her passport and uses the following protocol (which is repeated 2 or 3 times):

   1. Alice chooses an integer $r < n$ at random; she keeps it secret.
   2. Alice computes $y = r^2 \mod n$ and $z = x_A.r \mod n$;
   3. Alice sends y and z to Bob;
   4. Bob tests Alice's identity by verifying $a.y - z^2 = 0 \mod n$.

   Prove that if Eve, a spy who cannot compute square roots mod n, has succeeded to compute r, then Eve knows Alice's private key x_A. What to deduce?

c. However, with previous protocol, Eve can impersonate Alice; instead of steps 1 and 2, Eve chooses at random an integer z and computes $y = z^2/a \mod n$.
   To avoid this, the following *zero-knowledge* protocol is used (which is repeated k times);

   1. Alice chooses r at random, computes $y = r^2 \mod n$ and sends y to Bob;
   2. Bob chooses at random $b \in \{0, 1\}$; Bob sends b to Alice;
   3. If Alice receives 0, then she sends $z = r$ to Bob (i.e. a square root of y modulo n); else, if she receives 1, she sends to Bob $z = x_A.r \mod n$ (i.e. a square root of $y.a \mod n$).
   4. Bob tests Alice's identity by verifying that $y.a^b - z^2 = 0 \mod n$.

   Give an upper bound on the probability that Eve, who wants to impersonate Alice, can correctly answer to Bob after k executions of the protocol.

a. Currently, no algorithm is known to factorize n (1024 bits) in a time lesser than the duration of a passport (let say 5 years). Then, we can assume that nobody knows p and q except TTP.
   Moreover, we assume that nobody knows the private key x_A of Alice, except Alice and may be TTP.
   The only solution to compute x_A is then to compute the square root of $a \mod n$; from 2., this computation is more difficult that factorizing n. Thus, we may assume that nobody knows x_A (except Alice or TTP). The assumption is reasonable.
   Complement (not asked): yet, we may assume that TTP, who knows p and q, does not know x_A. Indeed, computing x_A from a requires to know how to compute square roots mod p. However, we can then prove that TTP would know how to compute discrete logarithm, which is conjectured computationally impossible. Let g be a primitive root of $\mathbb{Z}/p\mathbb{Z}^\star$. Let $y < p$; by computing a square root of $y \mod p$ (or of y/g if y does not have square root), TTP can compute y_1 such that $y_1^2 = y$. If $y_1 = g$, he then recovers the discrete logarithm of y: 2 (ou 1). Else by repeating this square root computation from y_1 until finding $y_k = g$, he can computes the discrete logarithm i of y_1; and then the discrete logarithm $2.i$ (or $2.i + 1$) of y. TTP then would know how to compute the discrete logarithm modulo p.

b. If Eve cannot compute square roots, r being chosen at random, knowing y is of no help. The only solution to compute r is then to use z; but computing r from z is equivalent to compute x_A. The only solution to compute r is then to compute x_A.
We deduce that only Alice can systematically answers correctly to Bob; then Bob can authenticate Alice.

c. If Eve doesn't know Alice's private key nor computing square roots, the only solution for her is to cheat. She has to bet on what Bob will send (0 or 1) to sends him a value y corresponding to a z she knows. But her probability to correctly succeed her bet is $1/2$. Then her probability to impersonate to Bob after k iterations is 2^{-k} which is rather small (if $k = 40$, it is $< 10^{-12}$).