Modern Cryptology (236506) – Public Key Algorithms

1 Probabilistic algorithms

Definition 1 We say that $A$ is a probabilistic polynomial time algorithm for computing a function $f$ if $A$ is polynomial and:

$$\forall x \Pr[A(x) \neq f(x)] \leq \frac{1}{3}$$

The probability is upon $r$, the randomness of $A$.

Notes:

- $\frac{1}{3}$ - a constant smaller than $\frac{1}{2}$. By repeating $A$ a polynomial number of times, we can decrease the error probability exponentially.
- $\forall x$ the error probability is small.

2 Computing square roots modulo a prime $p$

Reminder: $a$ is QR iff $a^{\frac{p-1}{2}} \equiv 1 \pmod{p}$ (otherwise, $a^{\frac{p-1}{2}} \equiv -1 \pmod{p}$).

So it is easy to check if a number is QR. We will see that it is also easy to find the square roots.

if $b^2 \equiv a \pmod{p}$, then $b, -b$ are the square roots of $a$ (it is enough to find one of them). Assume that $a$ is a QR.

Case 1: $p = 4k + 3$.

$$1 \equiv a^{\frac{p-1}{2}} \equiv a^{\frac{4k+3-1}{2}} \equiv a^{\frac{4k+2}{2}} \equiv a^{2k+1} \pmod{p}$$
Multiply by $a$:

$$a \equiv a^{2k+2} \equiv (a^{k+1})^2 \pmod{p}$$

So $a^{k+1}$ is a square root of $a$ and we are done.

**Case 2:** $p = 4k + 1$.

$$1 \equiv a^{p-1} \equiv a^{4k} \equiv a^2 \pmod{p}$$

So $a^2 \equiv 1 \pmod{p}$ or $a^{2k+1} \equiv a \pmod{p}$. We would like to have $a^i \equiv 1 \pmod{p}$ with an odd $i$, i.e., $i = 2m + 1$, and then $a \equiv a^{2m+2} \equiv (a^m + 1)^2 \pmod{p}$.

**The Tonelli-Shanks Algorithm**

$a^{2k} \equiv 1 \pmod{p}$. We can write $2k = 2^l(2j + 1)$, so:

$$a^{2^l(2j+1)} \equiv 1 \pmod{p} \Rightarrow a^{2^{l-1}(2j+1)} \in \{\pm 1\}$$

Assume $a^{2^{l-1}(2j+1)} \equiv 1 \pmod{p}$. We can continue in the same way, if we always get 1, eventually we get $a^{2j+1} \equiv 1 \pmod{p}$ and we are done.

Assume we get $-1$:

$$l' < l, \ a^{2^{l'}(2j+1)} \equiv -1 \pmod{p}$$

We would like to multiply by $-1$:

$$-1 \cdot a^{2^{l'}(2j+1)} \equiv 1 \pmod{p}$$

How to continue? (we do not know a square root of $-1$)

**The trick:** Assume that $b$ is QNR, then:

$$-1 \equiv b^{\frac{p-1}{2}} \equiv b^{2k} \equiv b^{2^l(2j+1)} \pmod{p}$$

Note that $l' < l$, so after we multiply both sides by $-1$ (or by $b^{2^l(2j+1)}$) we get:

$$b^{2^l(2j+1)} \cdot a^{2^{l'}(2j+1)} \equiv -1 \cdot -1 \equiv 1 \pmod{p}$$

Now, we can continue:

$$b^{2^{l-1}(2j+1)} \cdot a^{2^{l'-1}(2j+1)} \in \{\pm 1\}$$

We continue in the same way: if we get 1 we continue as before, and if we get $-1$ we multiply both sides by the equation $b^{2^{l}(2j+1)} \equiv -1 \pmod{p}$. We continue until we get $l' = 0$, i.e., we get:

$$b^{2^1(2j+1)} \cdot b^{2^2(2j+1)} \cdots b^{2^n(2j+1)} \cdot a^{2j+1} \equiv 1 \pmod{p}$$

Note that $0 < l_1, l_2, \ldots, l_n$, because $l' < l_i$ for all $i$. So we get:

$$b^{2B} \cdot a^{2j+1} \equiv 1 \pmod{p}$$

Or:

$$b^{2B} \cdot a^{2j+2} \equiv a \pmod{p} \Rightarrow (b^B \cdot a^{j+1})^2 \equiv a \pmod{p}$$

So we found a root.
What is probabilistic in the algorithm? Finding QNR. In order to use a QNR $b$ we choose $b$ at random and check if it is a QNR, i.e., we check if $b^{\frac{p-1}{2}} \equiv -1 \pmod{p}$. If we choose $b \in \mathbb{Z}_p^*$ uniformly at random, we succeed with probability $\frac{1}{2}$. If we try 3 times, we succeed with probability $\frac{7}{8}$. One $b$ suffices.

3 Primality Testing

We are given a number $N$, and we want to check if $N$ is a prime. The idea is to find a property that distinguishes primes and composites.

If $N$ is prime, then for all $a \in \mathbb{Z}_N^*$ it holds that $a^{N-1} \equiv 1 \pmod{N}$.

First approach: We choose at random $1 \leq a \leq N-1$. If $a^{N-1} \not\equiv 1 \pmod{N}$ we return “composite”. Otherwise, we repeat on this process with another $a$. If we receive $a^{N-1} \equiv 1 \pmod{N}$ for a lot of values $a$, we return “prime”.

Note that if $N$ is prime, we will always return “prime”. What happens if $N$ is composite? We might hope that if $N$ is composite, there are a lot of values $a$ such that $a^{N-1} \not\equiv 1 \pmod{N}$. However, this is not the case, there are $\infty$ carmichael numbers: composite numbers $N$ such that for all $a \in \mathbb{Z}_N^*$, it holds that $a^{N-1} \equiv 1 \pmod{N}$. For example: 561, 1105, 1729, ...

The Miller-Rabin Algorithm

(for input $N > 2$, and a parameter $t$)

1. If $N$ is even, or $N = a^b$ for $1 < a, b$, return: “composite”.
2. Calculate $1 \leq r$ and odd $u$ such that $N - 1 = 2^r u$.
3. For $i = 1, ..., t$:
   a) Choose at random $1 \leq a \leq N - 1$.
   b) If $\gcd(a, N) \neq 1$ or $a^{N-1} \not\equiv 1 \pmod{N}$, return “composite”.
   c) Calculate the values: $a^u, a^{2u}, a^{3u}, ..., a^{2^{r-1}u} \equiv 1 \pmod{N}$. If one of them is a non-trivial square root of 1, return “composite”.
4. Return “prime”.

Notes:

- If $N$ is prime, we will always return “prime” (for all $a$, it holds that $a^{N-1} \equiv 1 \pmod{N}$, and there are no non-trivial square roots of 1 mod $N$).
- In order to check if $N = a^b$ for $1 < a, b$, note that $1 < b \leq \log_2 N$. We can check for every $1 < b \leq \log_2 N$ if there exists $a$ such that $N = a^b$ by numerical methods.
• If $N$ is composite, and we passed 1. \( \Rightarrow \) $N$ is odd, and has at least two different primes in its prime factorization (if $N = p^e$ we discover it in 1.). Assume we choose $a$ such that $\gcd(a, N) = 1$ and $a^{N-1} \equiv 1 \pmod{N}$ (otherwise, we are done). Let’s look at the values: $a^u, a^{2u}, a^{2^2u}, \ldots, a^{2^r u}$. Note that $a^{2^r u} \equiv a^{N-1} \equiv 1 \pmod{N}$. So there exists a minimal $i$ such that $a^{2^i u} \equiv 1 \pmod{N}$. If $i > 0$ we get $(a^{2^{i-1} u})^2 \equiv 1 \pmod{N}$. I.e., $a^{2^{i-1} u}$ is a square root of 1 different than 1. $N$ has at least two different primes in its prime factorization, so 1 has at least 4 different square roots mod $N$. It can be shown that in this way we find a non-trivial square root of 1 with high probability (greater than $\frac{1}{2}$).

• **Conclusion:** the success probability of the Miller-Rabin algorithm is at least $\frac{1}{2}$ in each iteration. After $t$ iterations we succeed with probability at least $1 - \frac{1}{2^t}$. 