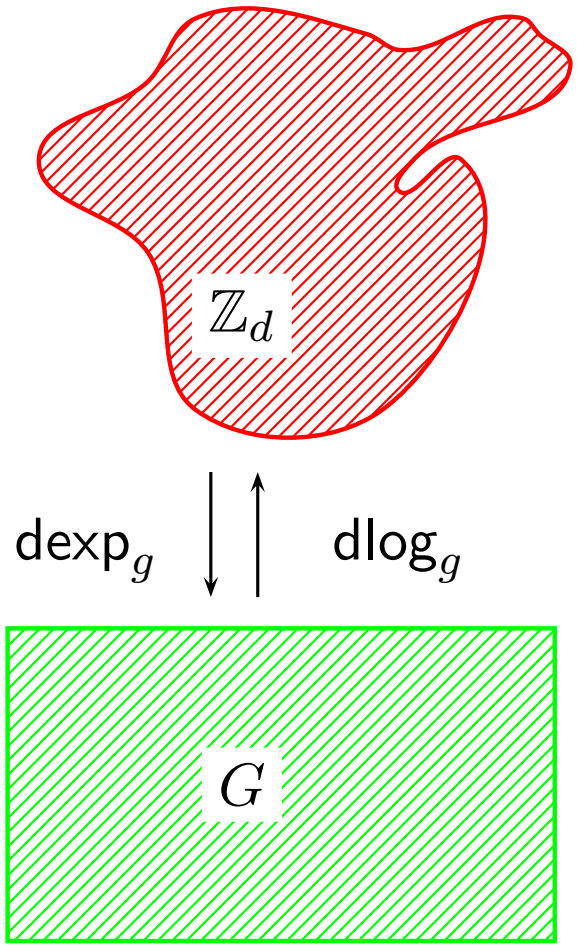
The art of cryptography: Heads and tails – Cryptographic random generation summer 2015

The Blum-Blum-Shub generator

Prof. Dr. Joachim von zur Gathen

Corollary. *Let G be a group of order d and $a \in \mathbb{Z}$ coprime to d. Then the exponentiation map $\pi_a : x \mapsto x^a$ is a permutation of G.*


$\mathbb{Z}_d$
dexp_g
dlog_g
G

Chinese Remainder Theorem (CRT). *Suppose that the integer N factors as $N = q_1 \cdots q_r$ with pairwise coprime $q_1, \ldots, q_r$. Then the group homomorphism*

$$\begin{array}{rcl} \mathbb{Z}_N^\times & \to & \mathbb{Z}_{q_1}^\times \times \cdots \times \mathbb{Z}_{q_r}^\times \\ x \bmod N & \mapsto & (x \bmod q_1, \ldots, x \bmod q_r) \end{array}$$

is an isomorphism.

THEOREM. *Let p be an odd prime, $e \geq 1$ and $\square_{p^e} = \{b^2 : b \in \mathbb{Z}_{p^e}^\times\}$ in $\mathbb{Z}_{p^e}^\times$. Then $\#\mathbb{Z}_{p^e}^\times = \phi(p^e) = p^{e-1}(p-1)$, and*

i. $\#\square_{p^e} = \phi(p^e)/2$.

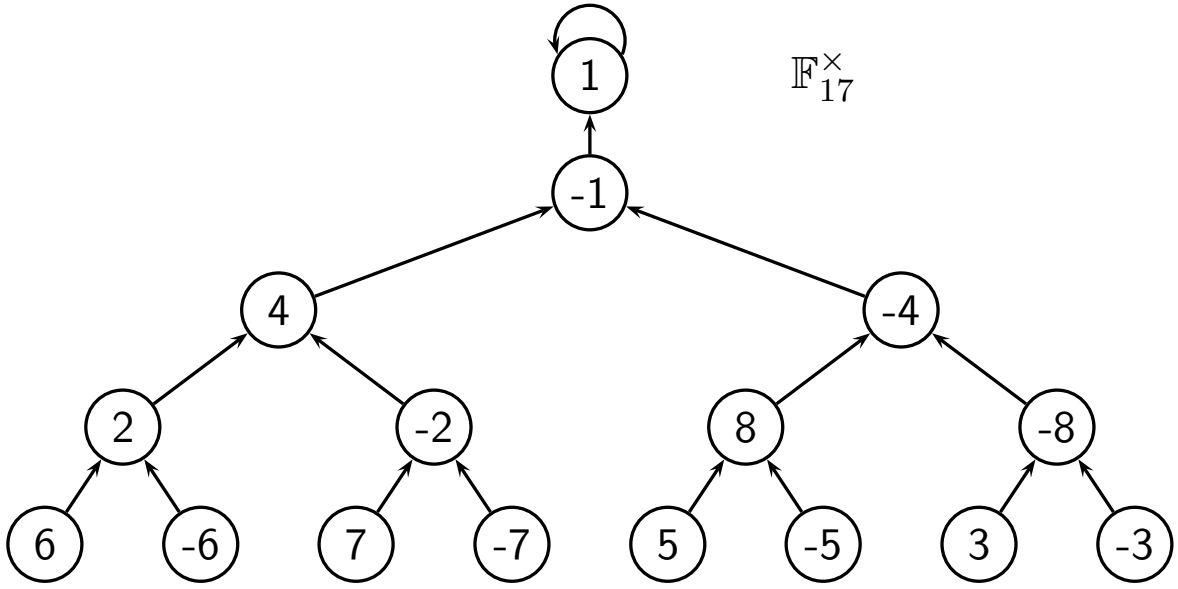
ii. *For any $a \in \mathbb{Z}_{p^e}^\times$, $a \in \square_{p^e} \iff a^{\phi(p^e)/2} = 1$.*

iii. *Any $a \in \square_{p^e}$ has exactly two square roots b_1 and b_2, and $b_1 + b_2 = 0$.*

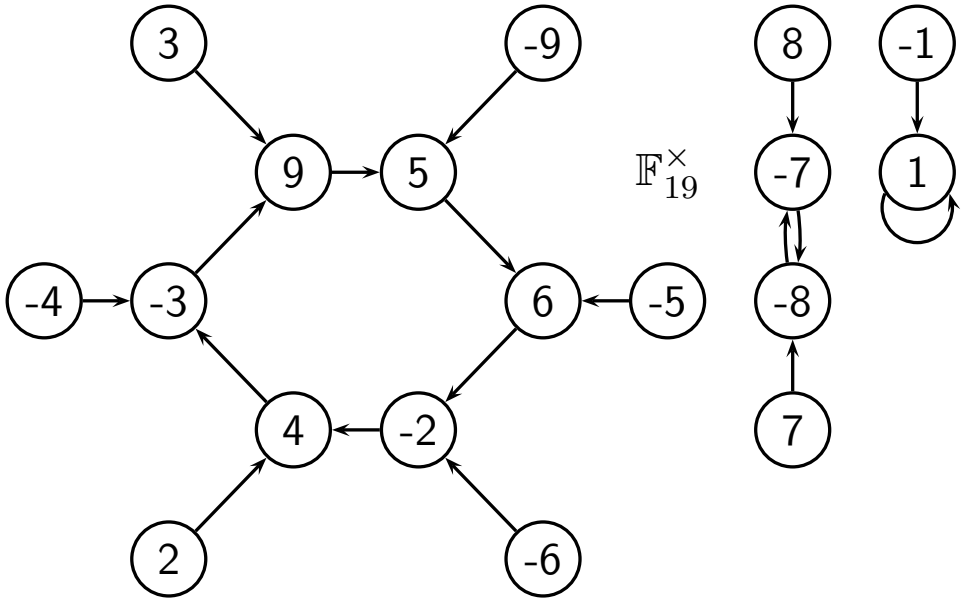
iv. *There is a probabilistic polynomial-time algorithm which, on input p^e and $a \in \mathbb{Z}_{p^e}^\times$, determines whether $a \in \square_{p^e}$, and if so, computes a square root $b \in \mathbb{Z}_{p^e}^\times$ with $a = b^2$.*

There is a concise way of associating to each number a the value of an indicator yes/no telling whether a is a square or not. Taking ± 1 for yes/no, we have the *Legendre symbol* for $a \in \mathbb{Z}$ and a prime p:

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } p \mid a, \\ 1 & \text{if } p \nmid a \text{ and } a \text{ is a square in } \mathbb{Z}_p^\times, \\ -1 & \text{otherwise.} \end{cases}$$



$\mathbb{F}_{17}^{\times}$



$\mathbb{F}_{19}^{\times}$

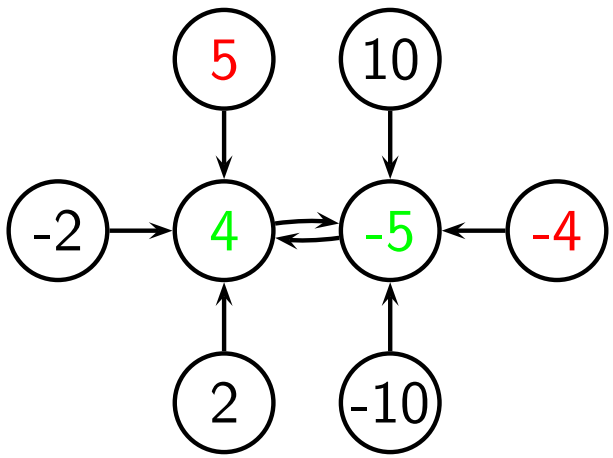
When $N = p \cdot q$ is the product of two distinct odd primes, then the situation is much more interesting. On the one hand, we can again consider the set

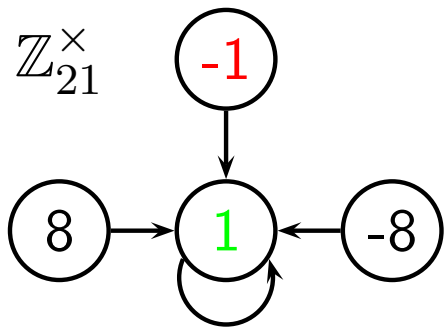
$$\square_N = \{b^2 : b \in \mathbb{Z}_N^\times\}$$

of squares modulo N. The CRT decomposes $\mathbb{Z}_N^\times$ into two constituents:

$$\mathbb{Z}_N^\times \cong \mathbb{Z}_p^\times \times \mathbb{Z}_q^\times.$$

Now some $a \in \mathbb{Z}_N^\times$ is a square if and only if it is a square both in $\mathbb{Z}_p^\times$ and in $\mathbb{Z}_q^\times$. These conditions are independent, and therefore only a quarter of the $\phi(N) = (p-1)(q-1)$ elements of $\mathbb{Z}_N^\times$ are squares.


5
10
-2
4
-5
-4
2
-10


$\mathbb{Z}_{21}^{\times}$
-1
8
1
-8

The *quadratic residuosity problem* in $\mathbb{Z}_N$ is to decide on input $a \in \square_N \cup \boxtimes_N$ whether $a \in \square_N$. Of course, given the factors p and q, this becomes easy since we can compute $\left(\frac{a}{p}\right)$ and $\left(\frac{a}{q}\right)$. But no polynomial-time algorithm is known if these factors are not provided.

Under the isomorphism

$$\chi\colon \mathbb{Z}_N^\times \longrightarrow \mathbb{Z}_p^\times \times \mathbb{Z}_q^\times$$

of the CRT, we have

$$\begin{aligned} \chi(\square_N) &= \square_p \times \square_q, \\ \chi(\boxtimes_N) &= \boxtimes_p \times \boxtimes_q\,. \end{aligned}$$

We consider the squaring map $\sigma_p\colon \mathbb{Z}_p^\times \longrightarrow \square_p \subseteq \mathbb{Z}_p^\times$ with $\sigma_p(a) = a^2$. If p is 3 modulo 4, then -1 is not a square modulo p, and exactly one of the two square roots a and $-a$ of a^2 is a square.

We now assume that p and q are both 3 modulo 4. Then $N = pq$ is called a Blum integer. If $\chi(a) = (u, v)$, then $\chi(a^2)$ has the four square roots

$$(u, v), (-u, v), (u, -v), (-u, -v).$$

Exactly one of them is a square, and χ^{-1} of this square is called the *principal (square) root* of a^2. If, say, $u \in \square_p$ and $v \in \boxtimes_p$, then $(u, -v)$ is the square among the four.

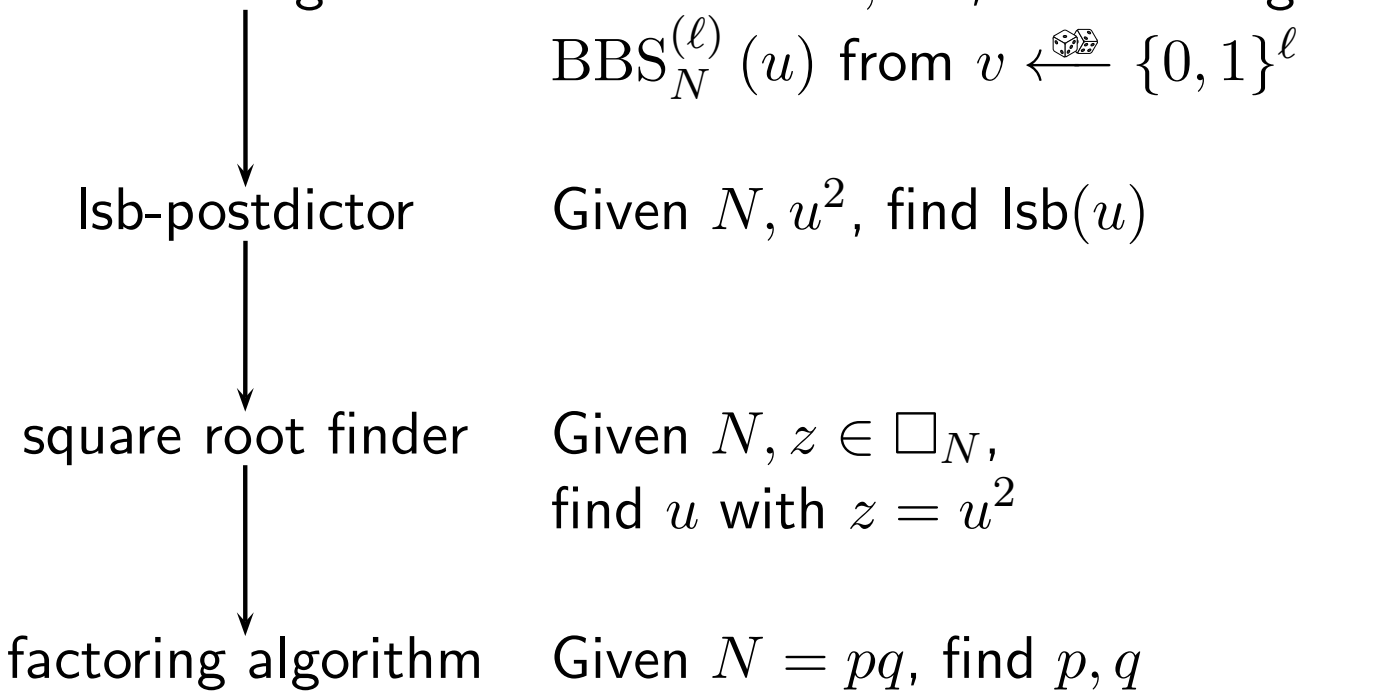
	$\left(\frac{a}{q}\right) = 1$	$\left(\frac{a}{q}\right) = -1$
$\left(\frac{a}{p}\right) = 1$	$\left(\frac{a}{N}\right) = 1$ $\square_N$ $\square_p \times \square_q$ $(u, -v)$	$\left(\frac{a}{N}\right) = -1$ $\square_p \times \boxtimes_q$ (u, v)
$\left(\frac{a}{p}\right) = -1$	$\left(\frac{a}{N}\right) = -1$ $\boxtimes_p \times \square_q$ $(-u, -v)$	$\left(\frac{a}{N}\right) = 1$ $\boxtimes_N$ $\boxtimes_p \times \boxtimes_q$ $(-u, v)$

EXAMPLE. We let $p = 3$ and $q = 7$, so that $N = 21$ and $\mathbb{Z}_{21}^{\times} = \{-10, -8, -5, -4, -2, -1, 1, 2, 4, 5, 8, 10\}$ in the symmetric system. Then $\square_3 = \{1\}$, $\boxtimes_3 = \{-1\}$, $\square_7 = \{-3, 1, 2, \}$, and $\boxtimes_7 = \{-2, -1, 3\}$. Not surprisingly, 1 is the principal root of 1, and 4 that of $-5 = 16$. But also -5 (and not 2) is the principal root of 4. In other words, $-5 = 16$ is the principal root of $25 = 4$ in $\mathbb{Z}_{21}$.

	$-3, 1, 2$	$-2, -1, 3$
1	$4 \leftrightarrow (1, -3)$ $1 \leftrightarrow (1, 1)$ $-5 \leftrightarrow (1, 2)$	$-2 \leftrightarrow (1, -2)$ $-8 \leftrightarrow (1, -1)$ $10 \leftrightarrow (1, 3)$
-1	$-10 \leftrightarrow (-1, -3)$ $8 \leftrightarrow (-1, 1)$ $2 \leftrightarrow (-1, 2)$	$5 \leftrightarrow (-1, -2)$ $-1 \leftrightarrow (-1, -1)$ $-4 \leftrightarrow (-1, 3)$

For an n-bit Blum integer N, an integer $\ell > 0$ as the bit-length of the desired output, and a seed $u = u_0^2 \in \square_N$ for $u_0 \xleftarrow{\text{🎲}} \mathbb{Z}_N^\times$, we define its output as

$$\mathrm{BBS}_N^{(\ell)}(u) = (\mathsf{lsb}(u), \mathsf{lsb}(u^2), \mathsf{lsb}(u^4), \ldots, \mathsf{lsb}(u^{2^{\ell-1}})).$$


BBS-distinguisher
Given N, u^{2^ℓ}, distinguish $\mathrm{BBS}_N^{(\ell)}(u)$ from $v \xleftarrow{\$} \{0,1\}^\ell$
lsb-postdictor
Given N, u^2, find $\mathrm{lsb}(u)$
square root finder
Given $N, z \in \square_N$, find u with $z = u^2$
factoring algorithm
Given $N = pq$, find p, q

REDUCTION. Lsb-postdictor $\mathcal{L}$ from $\mathrm{BBS}^{(\ell)}$-distinguisher $\mathcal{B}$.

Input: $N \in \mathbb{Z}, z = u^2 \in \square_N$.
Output: A bit in $\{0,1\}$.

1. $k \xleftarrow{\$} \{1, \ldots, \ell\}$.
2. $(v_1, \ldots, v_{k-1}, b) \xleftarrow{\$} \mathbb{B}^k$.
3. $v \leftarrow (v_1, \ldots, v_{k-1}, b, \mathsf{lsb}(z), \mathsf{lsb}(z^2), \ldots, \mathsf{lsb}(z^{2^{\ell-k-1}})) \in \{0,1\}^\ell$.
4. $b^* \leftarrow \mathcal{B}(N, z^{2^{\ell-k}}, v)$.
5. Return $b \oplus b^* \oplus 1$.

LEMMA. *Let $\mathcal{B}$ be a $\mathrm{BBS}^{(\ell)}$-distinguisher with advantage ϵ. Then $\mathcal{L}$ as in Reduction is an* lsb*-postdictor with advantage $\epsilon/2\ell$.*

REDUCTION. Square root finder $\mathcal{S}$ from lsb-postdictor $\mathcal{L}$.

Input: An n-bit Blum integer N and $y \in \square_N$.
Output: $x \in \mathbb{Z}_N^\times$ with $x^2 = y$ or "failure".

1. $a_0, b \xleftarrow{\text{🎲}} \mathbb{Z}_N$.
2. $u_0 \xleftarrow{\text{🎲}} \frac{\epsilon^3}{8} \cdot [0 \,..\, \frac{8}{\epsilon^3})$, $v \xleftarrow{\text{🎲}} \frac{\epsilon}{8} \cdot [0 \,..\, \frac{8}{\epsilon})$.
3. $\alpha_0, \beta \xleftarrow{\text{🎲}} \mathbb{B}$.
4. For t from 1 to n do steps 5–10
5. $a_t \leftarrow [a_{t-1}/2]_N$, $u_t \leftarrow (u_{t-1} + \alpha_{t-1})/2$.
6. $A_t \leftarrow \{i \in \mathbb{Z} : |2i+1| \leq 2\lceil n\epsilon^{-2} \rceil\}$.
7. For $i \in A_t$ do steps 8–9.
8. $c_{t,i} \leftarrow [(2i+1)a_t + b]_N$, $w_{t,i} \leftarrow \lfloor (2i+1)u_t + v \rfloor$.
9. $\alpha_{t,i} \leftarrow \mathcal{L}(N, c_{t,i}^2 y) + \beta + w_{t,i} \mod 2$.
10. If $\sum_{i \in A_t} \alpha_{t,i} < \#A_t/2$ then $\alpha_t \leftarrow 0$ else $\alpha_t \leftarrow 1$.
11. $x \leftarrow [a_n^{-1} \left\lfloor u_n N + \frac{1}{2} \right\rfloor]_N$.
12. If $x^2 = y$ in $\mathbb{Z}_N$ then return x else return "failure".

LEMMA. *Let $N = pq$ be a Blum integer, $y \in \square_N$, and $\mathcal{L}$ a polynomial-time* lsb*-postdictor as above with advantage at least ϵ. Then the square root finder $\mathcal{S}$ from Reduction on input N and y returns x satisfying $x^2 = y$ in $\mathbb{Z}_N$ with success probability at least $2^{-9}\epsilon^4$, and uses time polynomial in n and ϵ^{-1}.*

REDUCTION. Factoring algorithm $\mathcal{F}$ from square root finder $\mathcal{S}$.

Input: Positive odd integer N.
Output: A proper factor p of N or "failure".

1. $v \xleftarrow{\text{🎲}} \mathbb{Z}_N^\times$.
2. If $\mathcal{S}(N, v^2)$ returns "failure" then return "failure" else $v^* \leftarrow \mathcal{S}(N, v^2)$.
3. If $v \in \pm v^*$ in $\mathbb{Z}_N$ then return "failure".
4. $p \leftarrow \gcd(v - v^*, N)$.
5. Return p.

LEMMA. *Let $N = pq$ be a Blum integer and $\mathcal{S}$ a square-root finder as above with success probability at least σ. Then $\mathcal{F}$ as in Reduction returns a proper factor of N with probability $\sigma/2$.*

THEOREM. *Under the factoring assumption, the Blum-Blum-Shub generator is a pseudorandom generator.*