# RSA IV

## Factorization overview and Pollard rho

Tanja Lange
(some slides joint work with Daniel J. Bernstein)

Eindhoven University of Technology

2MMC10 – Cryptology

# How to factor RSA numbers?

Trial division.

# How to factor RSA numbers?

Trial division.

The prime-number theorem says that there are about

$$n/\ln(n)$$

primes up to $n$.

# How to factor RSA numbers?

Trial division.

The prime-number theorem says that there are about

$$n/\ln(n)$$

primes up to $n$.

That means roughly

$$(2^{2048}/\ln(2^{2048})) - (2^{2047}/\ln(2^{2047})) = 1.1377 \cdot 10^{613}$$

primes with 2048 bits.

No chance to find $p$ or $q$ by trial factorization.

# How to factor RSA numbers?

Trial division.

The prime-number theorem says that there are about

$$n/\ln(n)$$

primes up to $n$.

That means roughly

$$(2^{2048}/\ln(2^{2048})) - (2^{2047}/\ln(2^{2047})) = 1.1377 \cdot 10^{613}$$

primes with 2048 bits.

No chance to find $p$ or $q$ by trial factorization.

But: trial factorization is a useful step when factoring normal numbers.

# Short summary of factorization methods

- For small factors: trial factorization.

# Short summary of factorization methods

- For small factors: trial factorization.
- For medium factors:
  - Pollard's rho method.
  - $p - 1$ method, $p + 1$ method, ECM (elliptic curve method).

# Short summary of factorization methods

- For small factors: trial factorization.
- For medium factors:
  - Pollard's rho method.
  - $p-1$ method, $p+1$ method, ECM (elliptic curve method).
- For RSA numbers: Number field sieve
  - Works by turning hard factorization of one number into many easier factorizations.
  - Uses sieving (think of Eratosthenes) to find small factors.
  - Uses the above to find medium size factors.
  - Also needs a stage of linear algebra at the end.
- The number field sieve has subexponential complexity, so we need to more than double the bit length to make the attack twice as hard.

Will use $n$ for RSA numbers (hard to factor) and $m$ for normal numbers. Typically, $m$ is odd without very small prime divisors.

# Pollard's rho method for factorization

Define $\rho_0 = 0$, $\rho_{k+1} = \rho_k^2 + 11$.

Every prime $\leq 2^{20}$ divides
$S = (\rho_1 - \rho_2)(\rho_2 - \rho_4)(\rho_3 - \rho_6) \cdots (\rho_{3575} - \rho_{7150})$.
Also many larger primes do.

If such $p$ divides $m$, it divides $\gcd(S, m)$.
Computing $S$ takes $\approx 2^{14}$ multiplications mod $m$, very little memory.

Compare to $\approx 2^{16}$ divisions for trial division up to $2^{20}$.

Using Pollard rho to factor $m$ means computing $\rho_{k+1} = \rho_k^2 + 11 \bmod m$.

# Pollard's rho method for factorization

Define $\rho_0 = 0$, $\rho_{k+1} = \rho_k^2 + 11$.

Every prime $\leq 2^{20}$ divides
$S = (\rho_1 - \rho_2)(\rho_2 - \rho_4)(\rho_3 - \rho_6) \cdots (\rho_{3575} - \rho_{7150})$.
Also many larger primes do.

If such $p$ divides $m$, it divides $\gcd(S, m)$.
Computing $S$ takes $\approx 2^{14}$ multiplications mod $m$, very little memory.

Compare to $\approx 2^{16}$ divisions for trial division up to $2^{20}$.

Using Pollard rho to factor $m$ means computing $\rho_{k+1} = \rho_k^2 + 11 \bmod m$.

More generally: Choose $z$.
Compute $\gcd(S, m)$ where $S = (\rho_1 - \rho_2)(\rho_2 - \rho_4) \cdots (\rho_z - \rho_{2z})$.

# Analysis of Pollard's rho method for factorization

More generally: Choose $z$.
Compute $\gcd(S, m)$ where $S = (\rho_1 - \rho_2)(\rho_2 - \rho_4) \cdots (\rho_z - \rho_{2z})$.

How big does $z$ have to be for all primes $\leq y$ to divide $S$?

# Analysis of Pollard's rho method for factorization

More generally: Choose $z$.
Compute $\gcd(S, m)$ where $S = (\rho_1 - \rho_2)(\rho_2 - \rho_4) \cdots (\rho_z - \rho_{2z})$.

How big does $z$ have to be for all primes $\leq y$ to divide $S$?

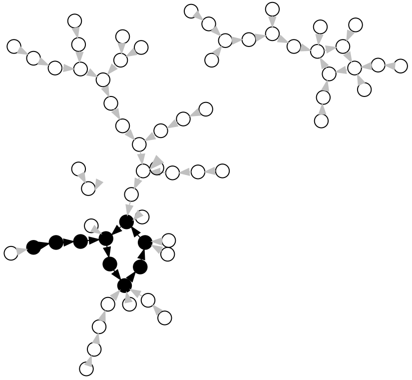
Consider walk $\rho_i$ modulo $p$.
There are $p$ elements modulo $p$, so expect collision $\rho_i \equiv \rho_j \bmod p$ after $\sqrt{\pi p/2}$ steps.

# Analysis of Pollard's rho method for factorization

More generally: Choose $z$.
Compute $\gcd(S, m)$ where $S = (\rho_1 - \rho_2)(\rho_2 - \rho_4) \cdots (\rho_z - \rho_{2z})$.

How big does $z$ have to be for all primes $\leq y$ to divide $S$?

Consider walk $\rho_i$ modulo $p$.
There are $p$ elements modulo $p$, so expect collision $\rho_i \equiv \rho_j \bmod p$ after $\sqrt{\pi p/2}$ steps.



Problem: We don't see collision as we work modulo $n$, not $p$.
But $p$ divides $\gcd(\rho_i - \rho_j, m)$.

$S$ implicitly uses Floyd,
product reduces number of gcd steps:
$\rho_i \equiv \rho_j \bmod p \Rightarrow \rho_k \equiv \rho_{2k} \bmod p$
for $k \in (j - i)\mathbf{Z} \cap [i, \infty] \cap [j, \infty]$.

# Analysis of Pollard's rho method for factorization

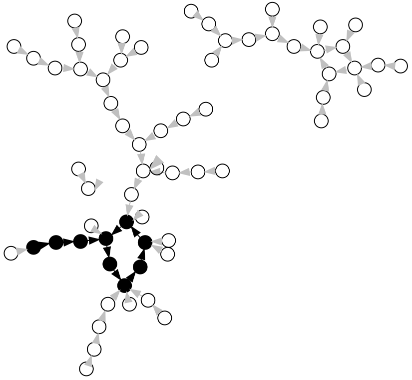
More generally: Choose $z$.
Compute $\gcd(S, m)$ where $S = (\rho_1 - \rho_2)(\rho_2 - \rho_4) \cdots (\rho_z - \rho_{2z})$.

How big does $z$ have to be for all primes $\leq y$ to divide $S$?

Consider walk $\rho_i$ modulo $p$.
There are $p$ elements modulo $p$, so expect collision $\rho_i \equiv \rho_j \bmod p$ after $\sqrt{\pi p/2}$ steps.

Problem: We don't see collision as we work modulo $n$, not $p$.
But $p$ divides $\gcd(\rho_i - \rho_j, m)$.

$S$ implicitly uses Floyd,
product reduces number of gcd steps:
$\rho_i \equiv \rho_j \bmod p \Rightarrow \rho_k \equiv \rho_{2k} \bmod p$
for $k \in (j-i)\mathbf{Z} \cap [i, \infty] \cap [j, \infty]$.



Plausible conjecture: $y^{1/2+o(1)}$; so $y^{1/2+o(1)}$ mults mod $m$.