Diffie-Hellman

key exchange

Government

Alice

Bob

Government

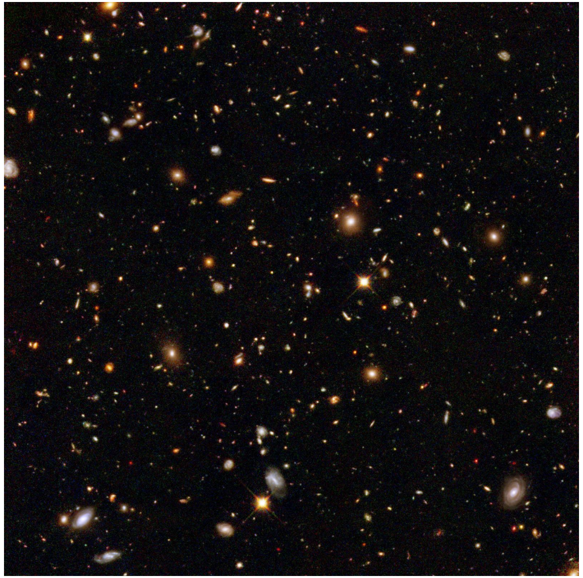
- 735 !!!

Alice

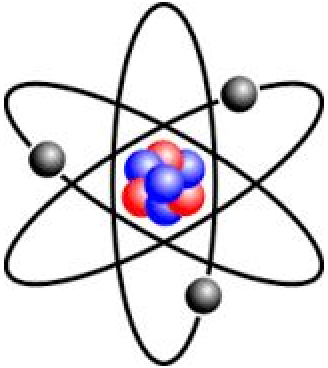
- 861276836409873 459862134031984 750987 ???

Bob

What else on stage?



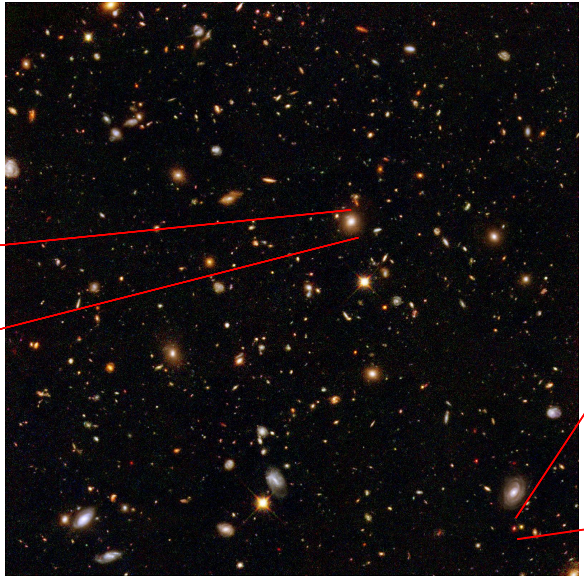
Group G



generator g

Last but not least:

Bob's secret



Group G

Alice's secret

1) Alice
2) Bob
3) Government
4) Alice's secret
5) Bob's secret
6) Group G (most important)
7) group generator g (second most important)

Group choice

- multiplicative group mod p (*)
- Schnorr's group
- Elliptic curves
- quadratic residues mod p
- additive group mod n (!)
- we're NOT working up to isomorphism

Multiplication mod p

$G = \{1, 2, ..., p\text{-}1\}$,

if $p = 7$,

$G = \{1, 2, 3, 4, 5, 6\}$

"overflow" multiplication rules:

$3 \times 3 \equiv 2 \bmod (7)$

$6 \times 6 \equiv ?? \bmod (7)$

generator choice g - must generate

2 IS NOT a generator of
$G = \{1, 2, 3, 4, 5, 6\}$

$2 \times 2 \times 2 \times \ldots$ does not give us all group elements.

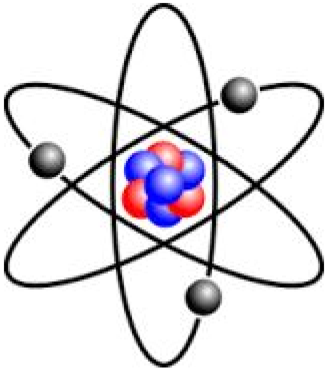
$$2 \times 2 \equiv 4 \pmod 7$$
$$2 \times 4 \equiv 1 \pmod 7$$
$$2 \times 1 \equiv 2 \pmod 7$$
$$2 \times 2 \equiv 4 \pmod 7$$
$$2 \times 4 \equiv 1 \pmod 7$$

3 IS a generator of
$G = \{1, 2, 3, 4, 5, 6\}$

$3 \times 3 \times 3 \times \ldots$ does give us all group elements.

$$3 \times 3 \equiv 2 \pmod 7$$
$$3 \times 2 \equiv 6 \pmod 7$$
$$3 \times 6 \equiv 4 \pmod 7$$
$$3 \times 4 \equiv 5 \pmod 7$$
$$3 \times 5 \equiv 1 \pmod 7$$
$$3 \times 1 \equiv 3 \pmod 7$$
$$3 \times 3 \equiv 2 \pmod 7$$

generator g MUST generate group G



generator g

Tasks

1. Using python or similar check if 13 is a generator of G = {1, ..., 1300582} (remember to work mod 1300583!)

2. Find a prime number, which is NOT a generator of G = {1, ..., 1300582} (remember to work mod 1300583!)

3. How many times do you need to multiply 13 by itself to get 12? (remember to work mod 1300583!)

Discrete log

previous task: how many times do you need to multiply 13 by itself to get 12?

$13^{1174920} \equiv 12 \pmod{1300583}$

In general:
$g^x \equiv b \pmod{p}$

Claim 1. It's easier to solve for b, then to solve for x. The difference in hardness is exponential.

Tasks

4. Compute a for
$13^{1174920} \equiv a \pmod{1300583}$
in less than 200 basic operations.

(basic operation is multiplication, taking a remainder, addition, division, subtraction)

Remark

4. Compute a for
$13^{1174920} \equiv a \pmod{1300583}$
in less than 200 basic operations.

(basic operation is multiplication, taking a remainder, addition, division, subtraction)

If you're careful about your choices, there are no significantly better ways to solve:
$13^x \equiv 12$
than to brute force

Actual complexities

Repeated squaring takes O (log(n)) basic operations.

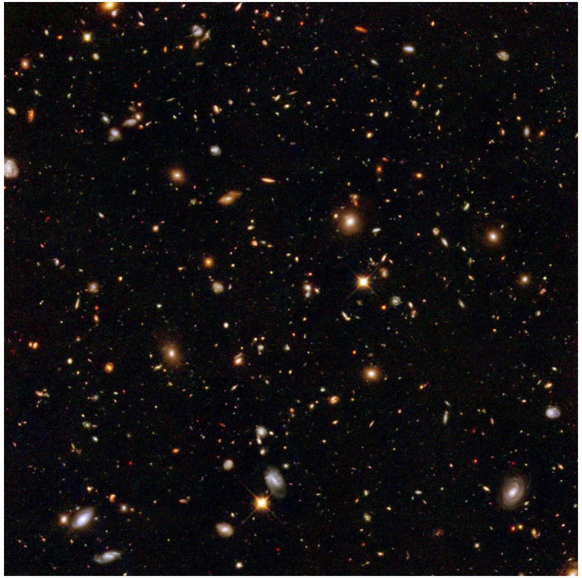
In our example log(1174920) = 13

State of the art baby-step giant step algorithm for discrete log takes O(sqrt(n))

In our example sqrt (1300583) = 1140

Group G > Universe

2^{1024} >



Actual complexities

Repeated squaring requires $O(\log(2^{1024})) \sim 1024$ basic operations

baby-step giant step algorithm for discrete log takes $O(\mathrm{sqrt}(2^{1024})) \sim 2^{512}$, still greater than the number of atoms in the universe.

Forget hardware types

int = 32 bits
long = 64 bits
long long = 124 bits

long long long long long long long
long long long = 1024 bits
(hypothetical)

Use mplib.org

Be warned:

"Attempting computations of more than 41 billion digits will cause overflow in the mpz type."

Luckily, we only need ~ 1000 digits.

So what about Alice and Bob?

1. Alice and Bob agree on a Group G (including parameter p), and on generator g.

2. Alice picks a secret a,
which is a random integer
between 1 and the size of G.

3. She computes, using repeated squaring, g^a, and broadcasts it to Bob. Her secret is safe, because Obama can't do discrete log.

4. Bob picks a secret b, which is a random integer between 1 and the size of G.

5. He computes, using repeated squaring, g^b, and broadcasts it to Alice.

6. Alice takes Bob's secret, and uses repeated squaring to compute $(g^b)^a$

7. Bob takes Alice's secret, and uses repeated squaring to compute $(g^a)^b$

8. Bob and Alice have established a common secret $(g^a)^b = (g^b)^a$, which can be then used as an encryption key for a symmetric encryption algorithm.

If Diffie-Hellman assumption holds, and discrete log is hard.

Diffie-Hellman assumption:
1. Computing g^{ab} from g^a, g^b is as hard as computing a from g^a and b from g^b.

Discrete log assumption:
2. Computing a from g^a is hard.