Cryptography, winter 2014/15

Diffie-Hellman key exchange

Dr. Daniel Loebenberger

Groups!

Definition

A *group* is a nonempty set G with a binary operation $\cdot : G \times G \to G$ satisfying

Associativity: For all $a, b, c \in G$ it holds that $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.

Identity: There is an element $1 \in G$ such that for all $a \in G$ we have $a \cdot 1 = 1 \cdot a = a$.

Inverse: For every $a \in G$ there is an element $a^{-1} \in G$ with $a \cdot a^{-1} = a^{-1} \cdot a = 1$.

Most groups used in cryptography are *commutative*, which means that we have for all $a, b \in G$ that $a \cdot b = b \cdot a$.

Definition

A group G is *cyclic* iff there is an element $g \in G$ which generates the whole group. We then write $G = \langle g \rangle$.

Basic results

Lagrange's Theorem

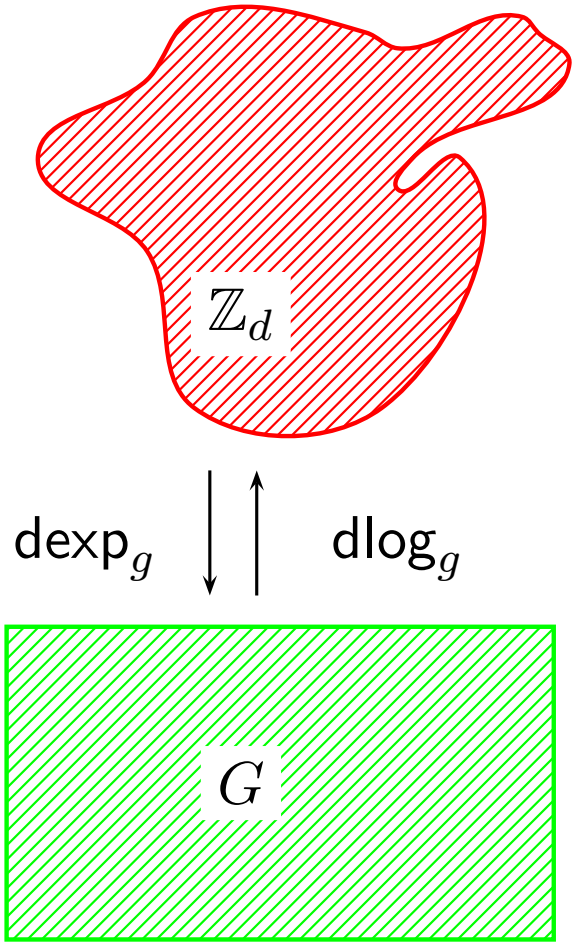
Let G be a finite group and $H \subseteq G$ a subgroup.

1. The order $\#H$ of H divides $\#G$.
2. If $H \neq G$, then $\#H \leq \#G/2$.
3. For any element x in G, we have $x^{\#G} = 1$.

Fermat's Theorem

Let N be prime and $x \in \mathbb{Z}_N$ with $x \neq 0$. Then $x^{N-1} = 1$ in $\mathbb{Z}_N$.

Back and forth



Repeated squaring

ALGORITHM. Repeated squaring.

Input: A group G, a base $x \in G$, and an exponent $e \in \mathbb{Z}$ with $1 \leq e < \#G$.

Output: $x^e \in G$.

1. Let $\sum_{0 \leq i < n} e_i 2^i$ be the binary representation of e with $e_0, \ldots, e_{n-1} \in \{0, 1\}$, $e_{n-1} = 1$, and n its bit length.
2. Set $y \leftarrow x$.
3. For i from $n - 2$ downto 0 do steps 4-5
4. Compute $y \leftarrow y^2$.
5. If $e_i = 1$, then compute $y \leftarrow y \cdot x$.
6. Return y.

Discrete logarithms

Discrete Logarithm Problem (DL_G)

Given a cyclic group $G = \langle g \rangle$ of order d with generator g, compute $a \in \mathbb{Z}_d$ with $g^a = x$.

Diffie-Hellman key exchange

PROTOCOL. Diffie-Hellman key exchange.

Key generation.

Input: security parameter n.
Output: G, g and d as below.

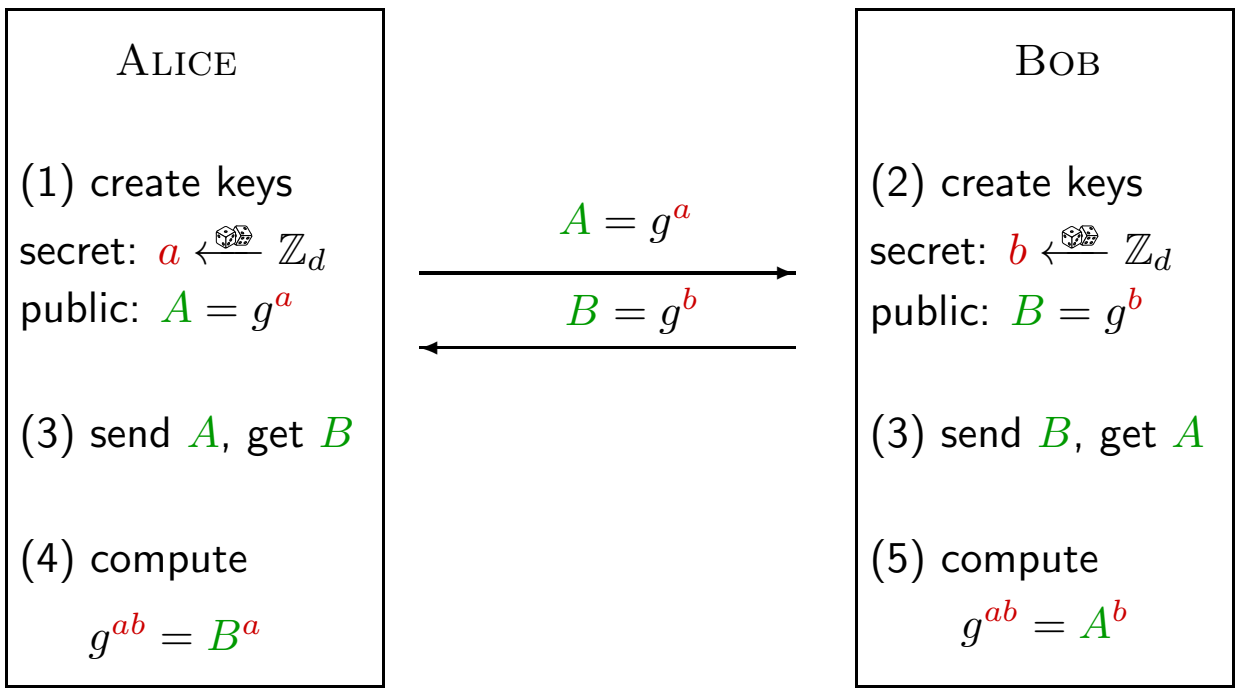
1. Determine a description of a finite cyclic group $G = \langle g \rangle$ with $d = \#G$ elements and a generator g, where d is an n-bit integer.

Key exchange.

2. Alice chooses her secret key $a \xleftarrow{\$} \mathbb{Z}_d$. She computes her public key $A \leftarrow g^a \in G$.
3. Bob chooses his secret key $b \xleftarrow{\$} \mathbb{Z}_d$. He computes his public key $B \leftarrow g^b \in G$.
4. Alice and Bob exchange their public keys A and B.
5. Alice computes the shared secret key $k_A = B^a$.
6. Bob computes the shared secret key $k_B = A^b$.

Diffie-Hellman key exchange

public: finite cyclic group $G = \langle g \rangle$ with
a generator $g \in G$ and $d = \#G$ elements.



Example

Take $G = \mathbb{Z}_{2579}^{\times}$ and $g = 2 \in G$.

- Alice chooses her secret key $a = 765$ and computes her public key $A = 2^{765} = 949$ in G.
- Bob chooses his secret key $b = 853$ and computes his public key $B = 2^{853} = 435$ in G.
- Alice and Bob exchange their public keys $A = 949$ and $B = 435$.
- Alice computes the shared secret key $k_A = B^{765} = 2424$ in G.
- Bob computes the shared secret key $k_B = A^{853} = 2424$ in G.

And lo and behold, the system works not only in general, but also in this particular case: Alice and Bob share the key $k_A = k_B$.

Man-in-the-middle attack

public: finite cyclic group $G = \langle g \rangle$ with $d = \#G$.

Alice		Eve		Bob
2. create keys secret: $a \xleftarrow{\$} \mathbb{Z}_d$ public: $A = g^a$		a. create keys $e \in \mathbb{Z}_d$ public: $E = g^e$		3. create keys secret: $b \xleftarrow{\$} \mathbb{Z}_d$ public: $B = g^b$
	$\xrightarrow{A}$		$\xrightarrow{E}$	
4. send A, get E	$\xleftarrow{E}$	b. send E, get A and B	$\xleftarrow{B}$	4. send B, get E
5. compute: $g^{ea} = E^a$		c. compute: $g^{ae} = A^e$ $g^{be} = B^e$		6. compute: $g^{eb} = E^b$

Diffie-Hellman Problem

Diffie-Hellman Problem (DH_G)

Given a group $G = \langle g \rangle$ of order d and A and B in G, compute $C \in G$ so that

$$A = g^a, B = g^b, C = g^{ab},$$

for some a and b in $\mathbb{Z}_d$.