Lecture 8 – Group theory, Diffie-Hellman key exchange

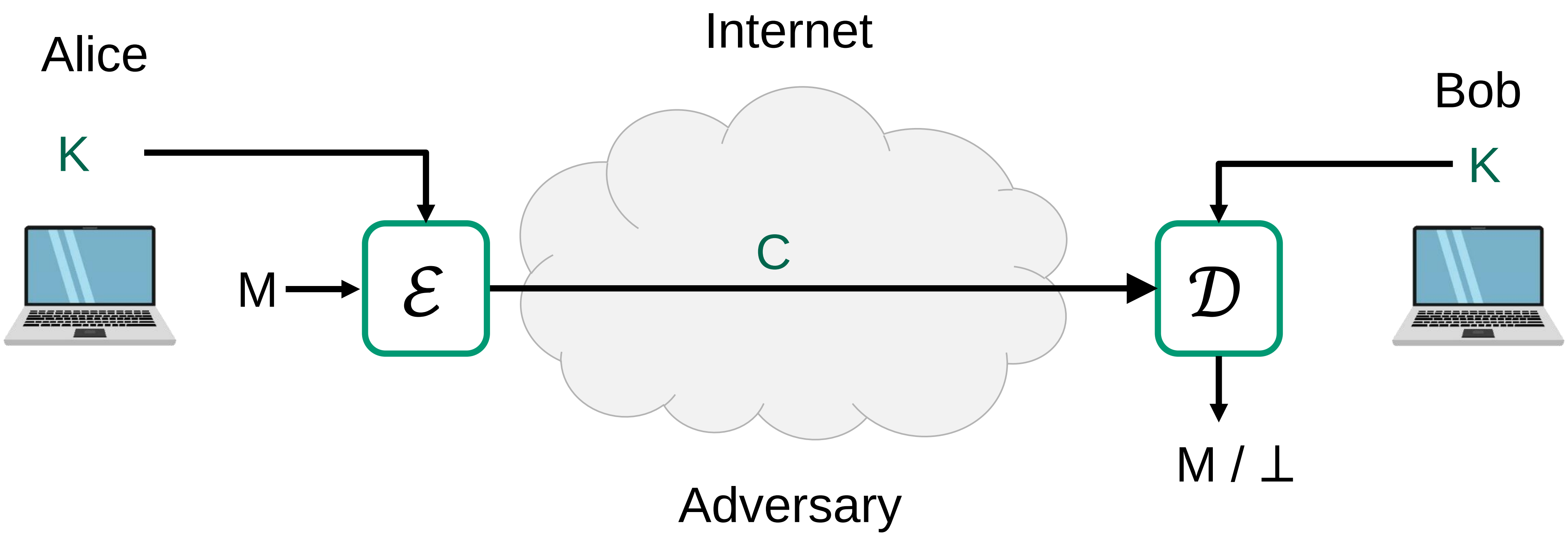
TEK4500

12.10.2022

Håkon Jacobsen

hakon.jacobsen@its.uio.no

Creating secure channels: encryption schemes

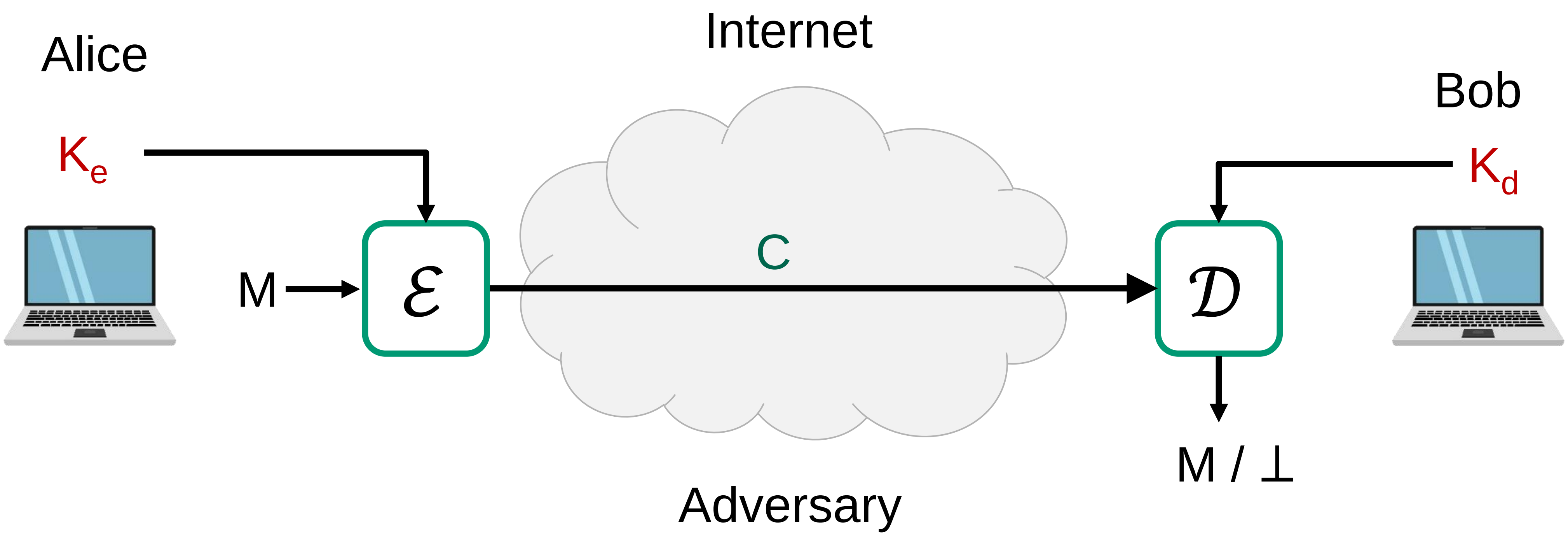


$\mathcal{E}$ **:** encryption algorithm (public)

$\mathcal{D}$ **:** decryption algorithm (public)

K : encryption / decryption key (secret)

Creating secure channels: encryption schemes



$\mathcal{E}$ **:** encryption algorithm (public)

$\mathcal{D}$ **:** decryption algorithm (public)

$\mathbf{K_e}$ **:** encryption key (public)

$\mathbf{K_d}$ **:** decryption key (secret)

Basic goals of cryptography

	Message privacy	**Message integrity / authentication**
Symmetric keys	Symmetric encryption	Message authentication codes (MAC)
Asymmetric keys	Asymmetric encryption (a.k.a. public-key encryption)	Digital signatures

(Key exchange)

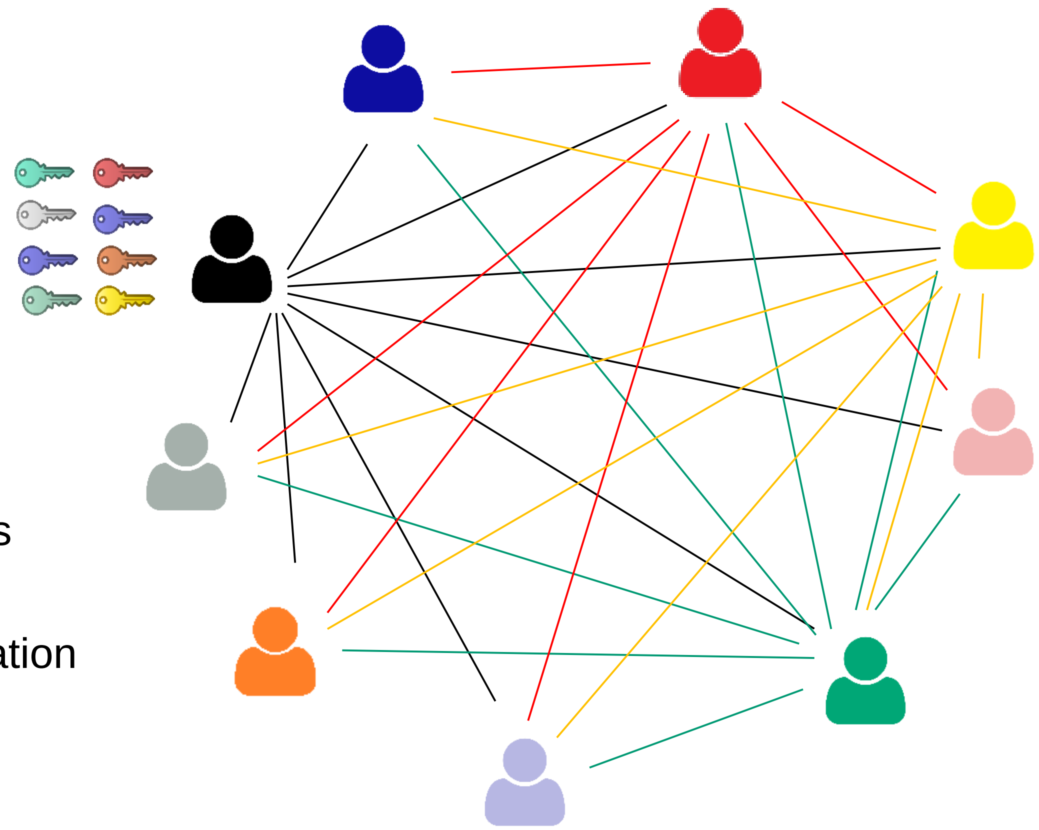
Symmetric key distribution problem

- One user needs to store N symmetric keys when communicating with N other users

- $N(N-1) = \mathcal{O}(N^2)$ keys stored in total

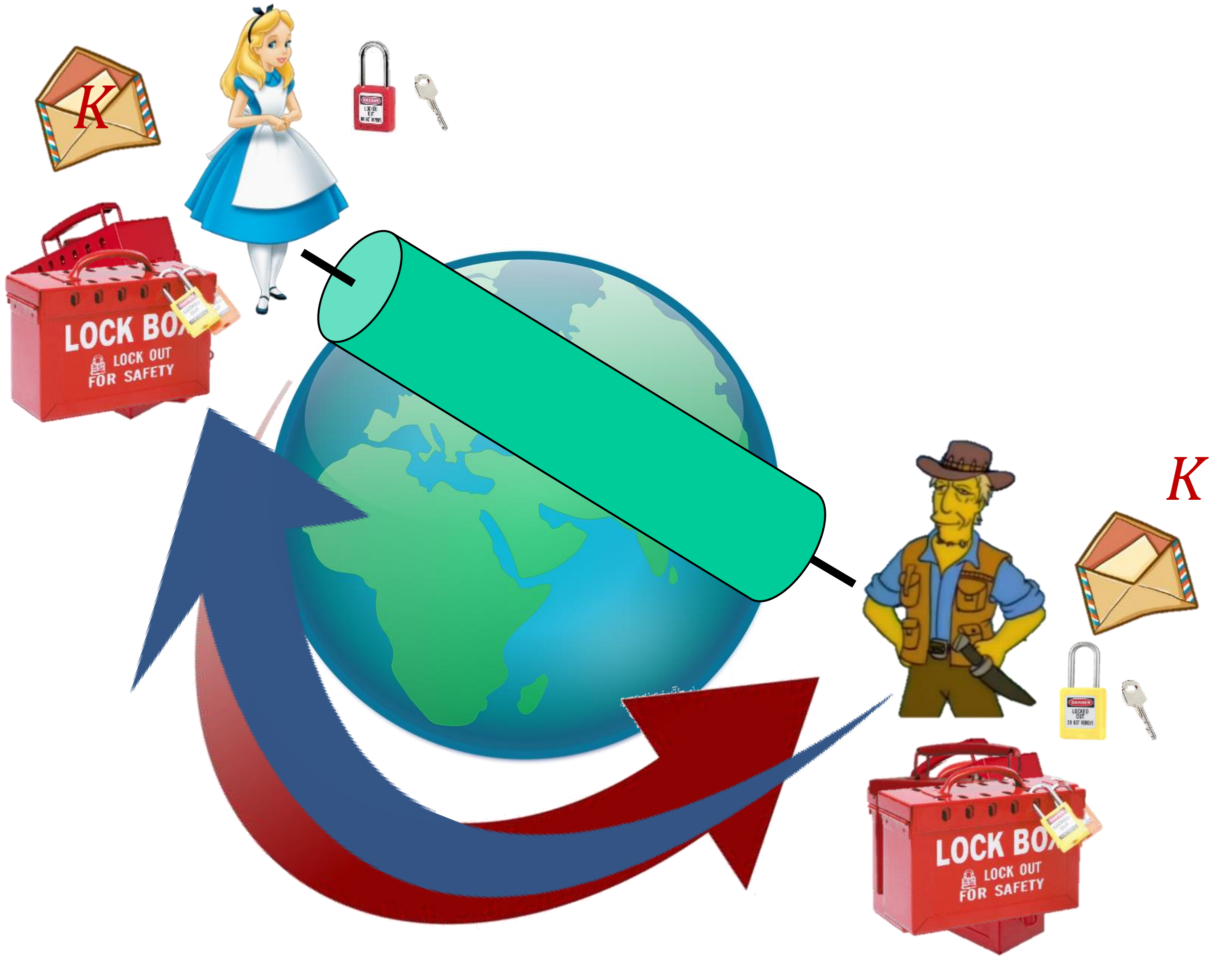
- Difficult to store and manage so many keys securely

- Partial solution: **key distribution centers**
  - One central authority hands out temporary keys
  - $\mathcal{O}(N)$ (long-term) keys needed (to the KDC)
  - Might be a feasible solution in a single organization
  - Single point of failure
  - **What about the internet?**



The public-key revolution

Diffie-Hellman key exchange – idea



Public-key encryption

Diffie-Hellman key exchange

- Discovered in the 1970's

- Allows two parties to establish a shared secret without ever having met

- Diffie & Hellman paper also introduced:
  - Public-key encryption
  - Digital signatures

Ralph Merkle Whitfield Diffie
Martin Hellman

New Directions in Cryptography

Invited Paper

Whitfield Diffie and Martin E. Hellman

Abstract Two kinds of contemporary developments in cryptography are examined. Widening applications of teleprocessing have given rise to a need for new types of cryptographic systems, which minimize the need for secure key distribution channels and supply the equivalent of a written signature. This paper suggests ways to solve these currently open problems. It also discusses how the theories of communication and computation are beginning to provide the tools to solve cryptographic problems of long standing.

1 INTRODUCTION

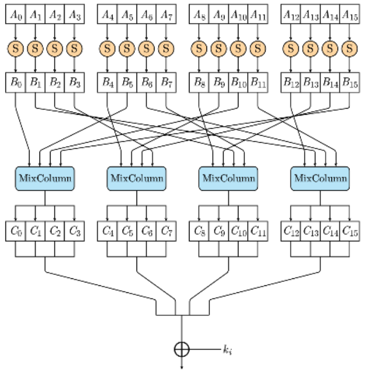
We stand today on the brink of a revolution in cryptography. The development of cheap digital hardware has freed it from the design limitations of mechanical computing and brought the cost of high grade cryptographic devices down to where they can be used in such commercial applications as remote cash dispensers and computer terminals. In turn, such applications create a need for new types of cryptographic systems which minimize the necessity of secure key distribution channels and supply the equivalent of a written signature. At the same time, theoretical developments in information theory and computer science show promise of providing provably secure cryptosystems, changing this ancient art into a science.

The development of computer controlled communication net- communications over an insecure channel order to use cryptography to insure privacy, however, it currently necessary for the communicating parties to share a key which is known to no one else. This is done by sending the key in advance over some secure channel such a private courier or registered mail. A private conversation between two people with no prior acquaintance is a common occurrence in business, however, and it is unrealistic to expect initial business contacts to be postponed long enough for keys to be transmitted by some physical means. The cost and delay imposed by this key distribution problem is a major barrier to the transfer of business communications to large teleprocessing networks.

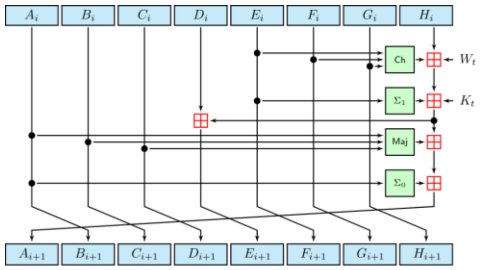
Section III proposes two approaches to transmitting keying information over public (i.e., insecure) channel without compromising the security of the system. In *public key cryptosystem* enciphering and deciphering are governed by distinct keys, *E* and *D*, such that computing *D* from *E* is computationally infeasible (e.g., requiring 10^{100} instructions). The enciphering key *E* can thus be publicly disclosed without compromising the deciphering key *D*. Each user of the network can, therefore, place his enciphering key in a public directory. This enables any user of the system to send a message to any other user enciphered in such a way that only the intended receiver is able to decipher it. As such, a public key cryptosystem is multiple access cipher. A private conversation can therefore be

A different kind of primitives

- Symmetric crypto boils down to a few *primitives*
  - Block ciphers/PRFs, hash functions
  - Why are these considered secure?
    - Lots and lots of cryptanalysis (*well-studied!*)
    - Artificial and man-made



AES



SHA2-256

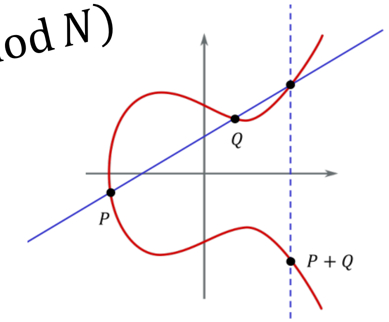
- Want asymmetric crypto to be based on a few well-studied primitives too

- Candidates come from a different place:
  - Hard *mathematical* problems
  - Good candidates: discrete logarithm problem, factoring
    - Much more algebraic structure

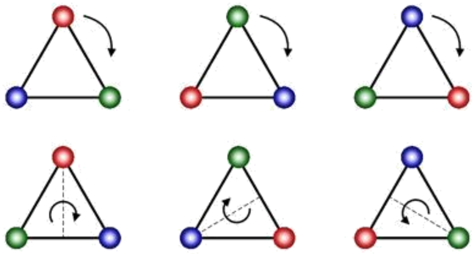
$$\boldsymbol{Z}_n^* \simeq \boldsymbol{Z}_{p_1}^* \times \boldsymbol{Z}_{p_2}^* \times \cdots \times \boldsymbol{Z}_{p_t}^*$$

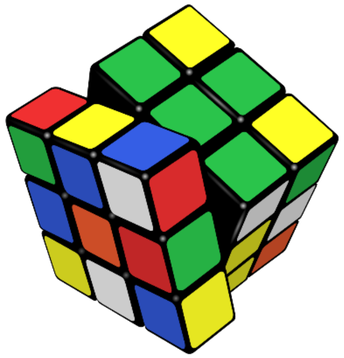
$$C \leftarrow M^e \pmod N$$

$$e : G_1 \times G_2 \to G_T$$



$y^2 = x^3 - 5x + 2$





$$\begin{tikzcd}[column sep=small] & \mathbb{Z}_{10} \arrow[dl, dash] \arrow[dr, dash] & \\ \mathbb{Z}_5 \arrow[dr, dash] & & \mathbb{Z}_2 \arrow[dl, dash] \\ & 0 & \end{tikzcd}$$

Group theory + number theory

Preliminaries

(integers) $\boldsymbol{Z} = \{\dots, -2, -1, 0, 1, 2, 3, \dots\}$

An integer $p > 1$ is **prime** if it's only divisible by 1 and p

(reals) $\boldsymbol{R}$ = the real numbers

$\boldsymbol{R}^* = \boldsymbol{R} \setminus \{0\}$

(integers "mod n") $\boldsymbol{Z}_n = \{0, 1, 2, \dots, n-1\}$

p prime

(integers "mod p") $\boldsymbol{Z}_p = \{0, 1, 2, \dots, p-1\}$

$\boldsymbol{Z}_p^* = \boldsymbol{Z}_p \setminus \{0\}$

Examples:

$\boldsymbol{Z}_{11} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

$\boldsymbol{Z}_{11}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

Groups – motivation

Groups

$(\boldsymbol{Z}, +)$ $(\boldsymbol{R}, +)$ $(\boldsymbol{R}^*, \cdot)$

Associativity

$(1+2)+3 = 1+(2+3)$

$\left(\sqrt{2}+e\right)+\pi = \sqrt{2}+(e+\pi)$

$\left(\sqrt{2}\cdot e\right)\cdot\pi = \sqrt{2}\cdot(e\cdot\pi)$

Identity

$6+0=6$

$3\sqrt{2}+0=3\sqrt{2}$

$3\sqrt{2}\cdot 1 = 3\sqrt{2}$

Inverses

$6+(-6)=0$

$3\sqrt{2}+\left(-3\sqrt{2}\right)=0$

$3\sqrt{2}\cdot\frac{1}{3\sqrt{2}}=1$

Groups – definition

Definition: A **group** $(G,\circ)$ is a set G together with a binary operation $\circ$ satisfying the following axioms

$\mathcal{G}1$: $(a \circ b) \circ c = a \circ (b \circ c)$ for all $a, b, c, \in G$ (associativity)

$\mathcal{G}2$: $\exists e \in G$ such that $e \circ a = a \circ e = a$ for all $a \in G$ (identity)

$\mathcal{G}3$: $\forall a \in G$ there exists $a^{-1} \in G$ such that $a \circ a^{-1} = a^{-1} \circ a = e$ (inverse)

A group is **abelian/commutative** if: $a \circ b = b \circ a$ for all $a, b \in G$

The **order** of a group is the number of elements in G, denoted $|G|$

Groups – examples

Definition: A **group** $(G,\circ)$ …

$\mathcal{G}1$: $(a \circ b) \circ c = a \circ (b \circ c)$ (associativity)
$\mathcal{G}2$: $\exists e \in G$: $e \circ a = a \circ e = a$ (identity)
$\mathcal{G}3$: $\exists a^{-1} \in G$: $a \circ a^{-1} = a^{-1} \circ a = e$ (inverse)

Groups

$(\boldsymbol{Z},+)$ $e = 0$ $"2^{-1}" = -2$

$(\boldsymbol{R},+)$ $e = 0$ $"\pi^{-1}" = -\pi$

$(\boldsymbol{R}^*,\cdot)$ $e = 1$ $\pi^{-1} = 1/\pi$

$(\boldsymbol{Z}_6,+_6)$ $e = 0$ $"2^{-1}" = 4:$ $2 + 4 = 6 \equiv 0 \bmod 6$

$(\boldsymbol{Z}_p^*,\cdot_p)$ $e = 1$

$"3^{-1}" = x$: $3 \cdot x \equiv 1 \bmod p$

Not groups

$(\boldsymbol{Z},\cdot)$ $2^{-1} = ?$

$(\boldsymbol{Z},-)$ $(1-2)-3 \neq 1-(2-3)$

$(\boldsymbol{R},\cdot)$ $0 \cdot x = 1?$

$(\boldsymbol{Z}_n,\cdot_n)$ $2x = 1 \pmod 6?$

$(\boldsymbol{Z}_p,\cdot_p)$

$(\boldsymbol{G},\circ)$

$\circ$	e	a	b
e	e	a	b
a	a	b	e
b	b	e	a

$(\boldsymbol{Z}_3,+_3)$

$+_3$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

$(\boldsymbol{G},.\circ)$

$\circ$	e	a	b	c
e	e	a	b	c
a	a	b	c	e
b	b	c	e	a
c	c	e	a	b

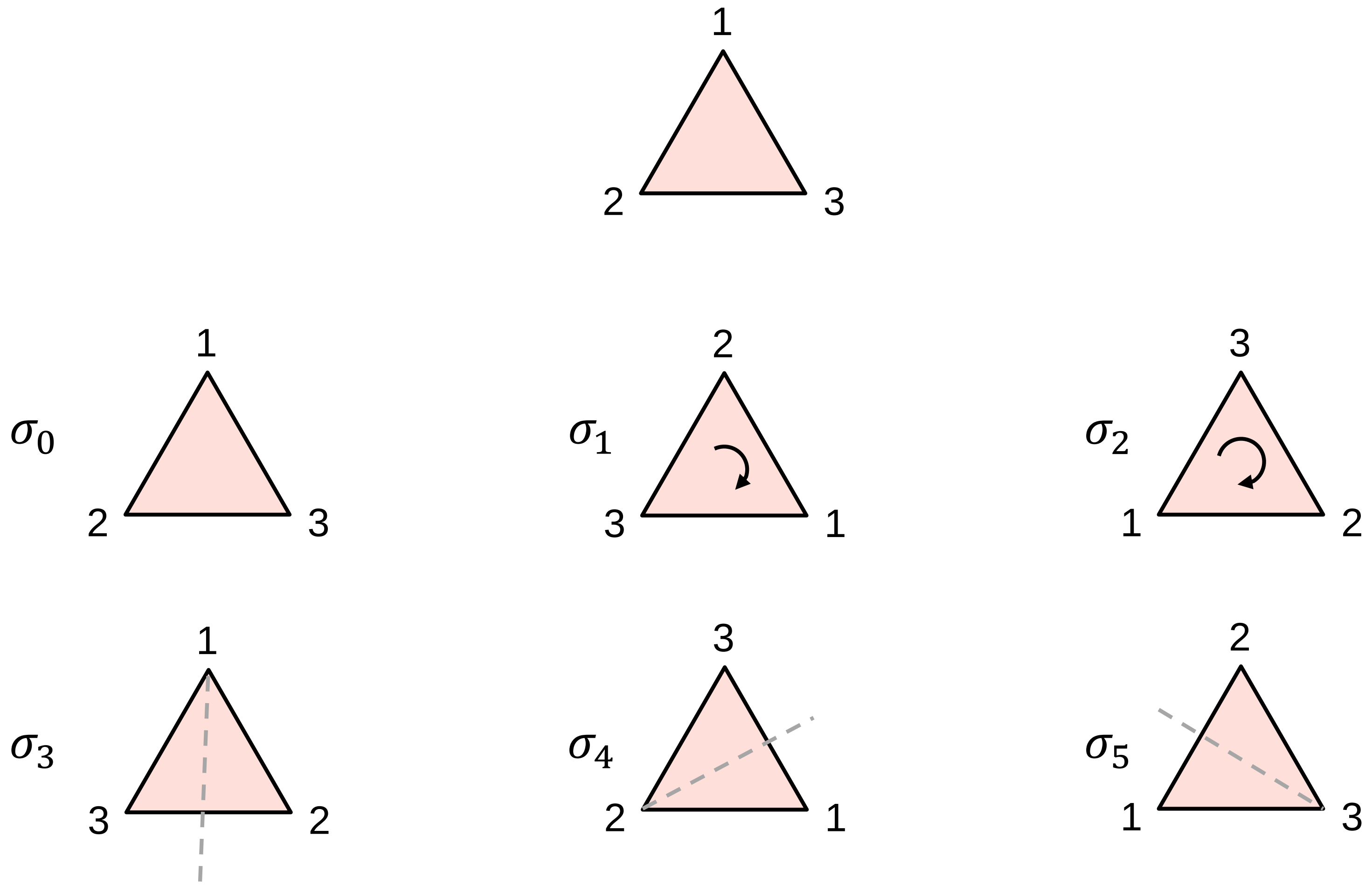
$(\boldsymbol{Z}_4,+_4)$

$+_4$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

$(\boldsymbol{G},\star)$

$\star$	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

More groups



$$\sigma_1 + \sigma_4 = \sigma_3$$

$$\sigma_3 + \sigma_3 = \sigma_0$$

$$\sigma_i + \sigma_0 = \sigma_i$$

Group exponentation

$$g^0 \stackrel{\text{def}}{=} e$$

$$g^n \stackrel{\text{def}}{=} \overbrace{g \circ g \circ \cdots \circ g}^{n}$$

$$g^{-n} \stackrel{\text{def}}{=} (g^{-1})^n$$

Fact: $g^n \circ g^m = \underbrace{\overbrace{g \circ \cdots \circ g}^{n} \circ \overbrace{g \circ \cdots \circ g}^{m}}_{n+m} = g^{n+m}$

Fact: $(g^n)^m = g^{nm} = (g^m)^n$

$(\boldsymbol{Z}, +)$: $\text{"}3^{15}\text{"} = \overbrace{3 + 3 + 3 + \cdots + 3}^{15} = 15 \cdot 3$

Multiplicative notation

$(G, \star)$: $a \star b = ab$
$a \star a \star \cdots a = a^n$

just notation!

Additive notation

$(G, \star)$: $a \star b = a + b$
$a \star a \star \cdots a = n \cdot a$

Cyclic groups

Definition: A **group** $(G,\circ)$ is **cyclic** if there exists $g \in G$ such that

$$G = \{\dots, g^{-2}, g^{-1}, g^0, g^1, g^2, g^3, \dots\}$$

g is called a **generator** for G and we write $(G,\circ) = \langle g \rangle$

Examples:

$(\boldsymbol{Z},+) = \langle 1 \rangle = \langle -1 \rangle$

$(\boldsymbol{Z}_n,+_n) = \langle 1 \rangle$

$$\begin{aligned}(\boldsymbol{Z}_7^*,\cdot) &= \langle 3 \rangle = \{3^0,3^1,3^2,3^3,3^4,3^5\} = \{1,3,2,6,4,5\}\\ &= \langle 5 \rangle = \{5^0,5^1,5^2,5^3,5^4,5^5\} = \{1,5,4,6,2,3\}\\ &\neq \langle 2 \rangle = \{2^0,2^1,2^2,2^3,2^4,2^5\} = \{1,2,4,1,2,4\} = \{1,2,4\}\end{aligned}$$

$(\boldsymbol{Z}_p^*,\cdot)$ cyclic for all primes p

Not cyclic groups:

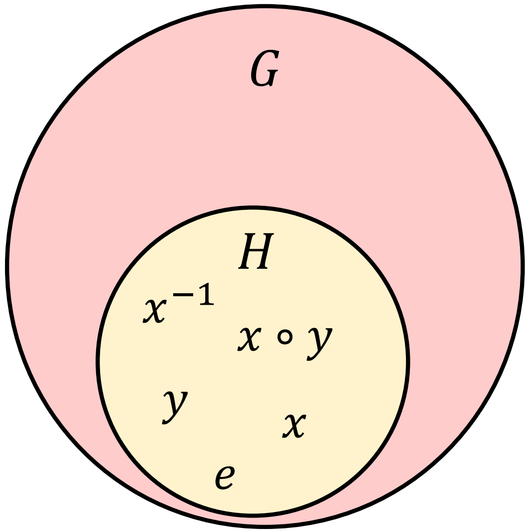
$(\boldsymbol{R},+)$ $(\boldsymbol{R}^*,\cdot)$

$(\boldsymbol{G},\star)$

$\star$	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

Subgroups

Definition: A set $H \subseteq G$ is a **subgroup,** written $H < G$, if H is a group under the binary operation inherited from G



Examples:

$\{e\} < G$ (for all groups)

$G < G$ (for all groups)

$2\boldsymbol{Z} = \{\dots, -2, 0, 2, 4, 6, \dots\} < (\boldsymbol{Z}, +)$

$3\boldsymbol{Z} = \{\dots, -3, 0, 3, 6, 9, \dots\} < (\boldsymbol{Z}, +)$

positive real numbers

$(\boldsymbol{R}_{>0}, \cdot) < (\boldsymbol{R}^*, \cdot)$

$(\{1, -1\}, \cdot) < (\boldsymbol{R}^*, \cdot)$

$\langle 20 \rangle < \langle 10 \rangle < \langle 5 \rangle < (\boldsymbol{Z}_{40}, +)$

$\langle 5 \rangle = \{0, 5, 10, \dots, 35\}$

$\langle 10 \rangle = \{0, 10, 20, 30\}$

$\langle 20 \rangle = \{0, 20\}$

Cyclic groups

Theorem: if $(G, \circ)$ is a finite group, then for all $g \in G$:

$$g^{|G|} = e$$

Proof (finite cyclic groups):

$|G| = |\langle g \rangle| = n$

$$\overbrace{e \quad g^1 \quad g^2 \quad g^3 \quad \cdots \quad g^{n-1}}^{G} \quad g^n \quad g^{n+1} \quad g^{n+2} \quad \cdots$$

$g^n = g^3 \implies g^{n-3} = e \implies g^j = e \quad j < n$ *contradiction!*

Theorem: $g^i = g^{i \,(\mathrm{mod}\, n)} = g^{i \,(\mathrm{mod}\, |G|)}$

Theorem (Lagrange's theorem): if $H < G$ then $|H|$ divides $|G|$

Groups of prime order

Theorem (Lagrange's theorem): if $H < G$ then $|H|$ divides $|G|$

Fact: any non-trivial element ($\neq e$) in a prime-order group is a generator

Fact: any prime-order group is cyclic

Warning: $(\boldsymbol{Z}_p^*, \cdot)$ is *not* a prime-order group! $|\boldsymbol{Z}_p^*| = p - 1$

Suppose $p = 2q + 1$, with q being prime; what are the possible sub-groups of $(\boldsymbol{Z}_p^*, \cdot)$?

$$|\boldsymbol{Z}_p^*| = p - 1 = 2q$$

$$\boldsymbol{Z}_p^* = \begin{cases} \{1\}, \\ \{1, -1\}, \\ H, & |H| = q \\ \boldsymbol{Z}_p^* \end{cases}$$

Example: $\boldsymbol{Z}_{11}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

$11 = 2 \cdot 5 + 1$

$\{1\} < \boldsymbol{Z}_{11}^*$

$\{1, -1\} = \{1,10\} < \boldsymbol{Z}_{11}^*$

$H = \langle 3 \rangle = \langle 4 \rangle = \langle 5 \rangle = \langle 9 \rangle = \{1, 3, 4, 5, 9\} < \boldsymbol{Z}_{11}^*$

$\boldsymbol{Z}_{11}^* < \boldsymbol{Z}_{11}^*$

Why is $(Z_p^*, \cdot)$ a group?

- $\boldsymbol{Z}_p^* = \{1, 2, \ \ldots, p-1\}$
- Associativity ✓ $(a \cdot b) \cdot c = a \cdot (b \cdot c) \bmod p$
- Identity ✓ $1 \cdot a = a \cdot 1 = a \bmod p$
- Inverses ?

Definition: A **group** $(G, \circ)$ …

$\mathcal{G}1$: $(a \circ b) \circ c = a \circ (b \circ c)$ (associativity)
$\mathcal{G}2$: $\exists e \in G$: $e \circ a = a \circ e = a$ (identity)
$\mathcal{G}3$: $\exists a^{-1} \in G$: $a \circ a^{-1} = a^{-1} \circ a = e$ (inverse)

Given $a \in \boldsymbol{Z}_p^*$ can we always find $a^{-1} \in \boldsymbol{Z}_p^*$?
Need to solve: $x \cdot a = 1 \bmod p$

How do we actually find a^{-1}?
Extended Euclidian Algorithm

Claim: $\exists m, n \in \boldsymbol{Z}$ such that $ma + np = 1 \implies ma = 1 - np = 1 \bmod p \implies a^{-1} = m \bmod p \in \boldsymbol{Z}_p^*$

Proof: $I = \{sa + tp \mid s, t \in \boldsymbol{Z}\}$

1) I contains a *positive* integer (e.g. a and p)
2) I contains a *smallest* positive integer $d = ma + np$
3) $p = qd + r \qquad 0 \le r < d$
4) $r = p - qd = p - q(ma + np) = -qma + (1 - qn)p \in I$
5) $r = 0$ (since d is the smallest positive integer in I)
6) $p = qd \implies d = 1$ (since p is prime and $d \le a < p$)
7) $1 = ma + np$

NON-CONSTRUCTIVE

Q.E.D

Diffie-Hellman

$G = \langle g \rangle$ ← public

$a \xleftarrow{\$} \{1, \ldots, |G|\}$

$\xrightarrow{A = g^a}$

$\xleftarrow{B = g^b}$

$b \xleftarrow{\$} \{1, \ldots, |G|\}$

$Z \leftarrow B^a = (g^b)^a = \underline{g^{ab}}$

$Z' \leftarrow A^b = (g^a)^b = \underline{g^{ab}}$

Claim: $Z = Z'$

Diffie-Hellman – example

$$\boldsymbol{Z}_{1019}^{*} = \langle 2 \rangle$$

$$493 \xleftarrow{\$} \{1, \dots, 1018\}$$

$$570 \leftarrow 2^{493} \bmod 1019$$

$$\xrightarrow{\quad 570 \quad}$$

$$\xleftarrow{\quad 72 \quad}$$

$$901 \xleftarrow{\$} \{1, \dots, 1018\}$$

$$72 \leftarrow 2^{901} \bmod 1019$$

$$Z \leftarrow 72^{493} \bmod 1019 \equiv \underline{\mathbf{531}}$$

$$Z' \leftarrow 570^{901} \bmod 1019 \equiv \underline{\mathbf{531}}$$

Diffie-Hellman

public

$G = \langle g \rangle$

$a \xleftarrow{\$} \{1, \ldots, |G|\}$

$A = g^a \longrightarrow$

$\longleftarrow B = g^b$

$b \xleftarrow{\$} \{1, \ldots, |G|\}$

$Z \leftarrow B^a = (g^b)^a = g^{ab}$

$Z' \leftarrow A^b = (g^a)^b = g^{ab}$

Security:

- Must be hard to compute $Z \leftarrow g^{ab}$ given g, A, B (DH assumption)
- Must be hard to find a (or b) given g, A, B (DLOG assumption)

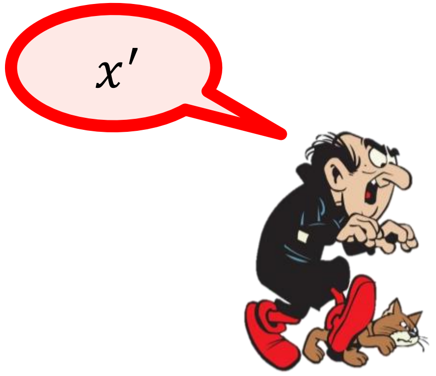
Doesn't work: $A \circ B = g^a \circ g^b = g^{a+b} \neq g^{ab}$

Discrete logarithm (DLOG) problem

$\mathbf{Exp}^{\mathrm{dlog}}_{G,g}(A)$
1. $x \xleftarrow{\$} \{1,2,\ldots,\lvert G\rvert\}$
2. $X \leftarrow g^x$
3. $x' \leftarrow A(X)$
4. **return** $x' \stackrel{?}{=} x$

public

$G = \langle g \rangle$



$\xleftarrow{\quad X \quad}$

Challenger

$x \xleftarrow{\$} \{1,2,\ldots,\lvert G\rvert\}$

$X \leftarrow g^x$

Adversary wins if $x' = x$

In other words: $x' = \log_g X$

Definition: The **DLOG-advantage** of an adversary A is

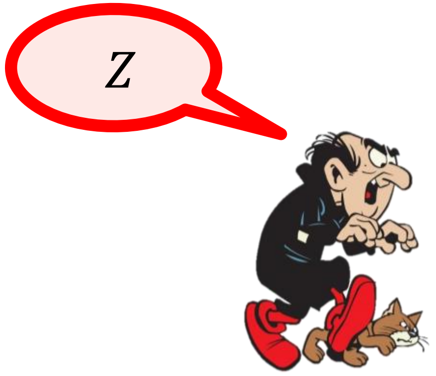
$$\mathbf{Adv}^{\mathrm{dlog}}_{G,g}(A) = \Pr\left[\mathbf{Exp}^{\mathrm{dlog}}_{G,g}(A) \Rightarrow \mathrm{true}\right]$$

Diffie-Hellman (DH) problem

$\mathbf{Exp}^{\mathrm{dh}}_{G,g}(A)$
1. $x, y \xleftarrow{\$} \{1,2,\ldots,\lvert G\rvert\}$
2. $X \leftarrow g^x$
3. $Y \leftarrow g^y$
4. $Z \leftarrow A(X,Y)$
5. **return** $Z \stackrel{?}{=} g^{xy}$

public

$G = \langle g \rangle$



X, Y ←

Challenger

$x, y \xleftarrow{\$} \{1,2,\ldots,\lvert G\rvert\}$

$X \leftarrow g^x$

$Y \leftarrow g^y$

Adversary wins if $Z = g^{xy}$

Definition: The **DH-advantage** of an adversary A is

$$\mathbf{Adv}^{\mathrm{dh}}_{G,g}(A) = \Pr\left[\mathbf{Exp}^{\mathrm{dh}}_{G,g}(A) \Rightarrow \mathrm{true}\right]$$

DLOG vs. DH

$\mathbf{Exp}^{\mathrm{dlog}}_{G,g}(A)$
1. $x \xleftarrow{\$} \{1,2,\dots,\lvert G\rvert\}$
2. $X \leftarrow g^x$
3. $x' \leftarrow A(X)$
4. **return** $x \stackrel{?}{=} x$

$\mathbf{Exp}^{\mathrm{dh}}_{G,g}(A)$
1. $x, y \xleftarrow{\$} \{1,2,\dots,\lvert G\rvert\}$
2. $X \leftarrow g^x$
3. $Y \leftarrow g^y$
4. $Z \leftarrow A(X,Y)$
5. **return** $Z \stackrel{?}{=} g^{xy}$

DLOG security $\stackrel{?}{\Longrightarrow}$ DH security

DLOG security $\Longleftarrow$ DH security

$\Updownarrow$

DLOG *insecurity* $\Longrightarrow$ DH *insecurity*

Algorithms for solving DLOG

- **Generic algorithms:** works for *all* (cyclic) groups
  - Brute-force
    1. Given g and $X \in G$
    2. for $i = 1, 2, \ldots, |G|$ check if $g^i = X$

    running time: $\mathcal{O}(|G|) = (2^n)$, given $|G| \approx 2^n$

  - Are there better algorithms?

- **Group-specific algorithms:** exploits algebraic features of given group

Solving DLOG: the baby-step giant-step algorithm

Given: $X \leftarrow g^x$ $\quad n \leftarrow \left\lceil \sqrt{|G|} \right\rceil$ $\quad Y \leftarrow g^n$

Find: x

Look for i, j such that:

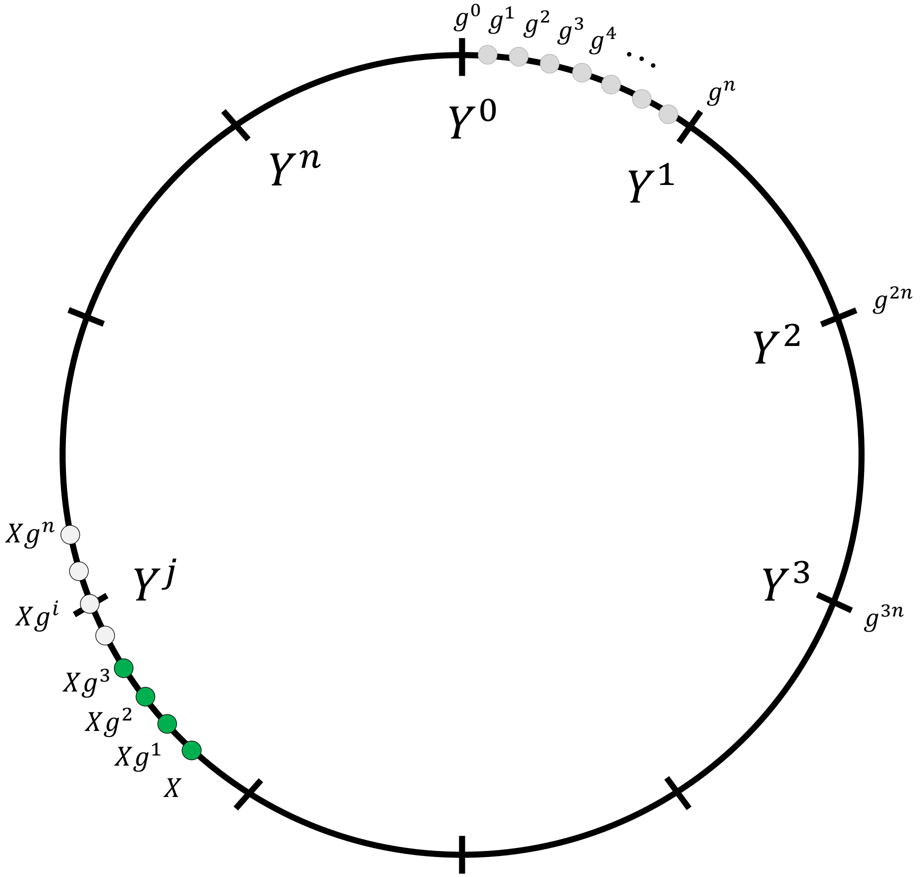
$$Xg^i = Y^j$$

$$X = Y^j g^{-i}$$

$$X = g^{nj} g^{-i}$$

$$g^x = g^{nj} g^{-i}$$

$$g^x = g^{nj-i}$$



$$\overbrace{Y^0, Y^1, \dots, Y^n}^{\mathcal{O}\left(\sqrt{|G|}\right)}$$

$$\underbrace{Xg^0, Xg^1, \dots, Xg^n}_{\tilde{\mathcal{O}}\left(\sqrt{|G|}\right)}$$

$\boxed{x = nj - i}$

Time + memory: $\tilde{\mathcal{O}}\left(\sqrt{|G|}\right)$

Generic algorithms for solving DLOG

- Baby-step, giant-step: time $\mathcal{O}\left(\sqrt{|G|}\right)$ memory $\mathcal{O}\left(\sqrt{|G|}\right)$
- Pollard's rho: time $\mathcal{O}\left(\sqrt{|G|}\right)$ memory $\mathcal{O}(1)$
- Pohlig-Hellman: time $\max_p \mathcal{O}\left(\sqrt{p}\right)$ memory $\mathcal{O}(1)$ $\left(|G| = p_1^{e_1} p_2^{e_2} \cdots p_t^{e_t}\right)$

- **Consequence:** for DLOG to be hard $\sqrt{|G|}$ must be large enough!
  - $|G| \approx 2^{128}$ only gives $\sqrt{2^{128}} = 2^{64}$ security
  - $|G| \approx 2^{256}$ only gives $\sqrt{2^{256}} = 2^{128}$ security
  - $|G| \approx 2^{512}$ only gives $\sqrt{2^{512}} = 2^{256}$ security
  - etc...

- **Nechaev'94 & Shoup'97:** Solving DLOG requires $\Omega\left(\sqrt{|G|}\right)$ time in *generic* groups

Non-generic algorithms for DLOG

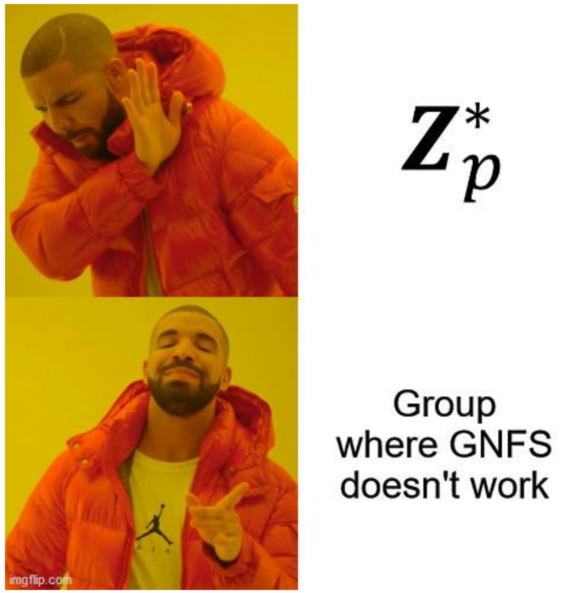
- Unfortunately, $(\boldsymbol{Z}_p^*, \cdot)$ is *not* a generic group!

- *Much* faster specific algorithms exist for solving DLOG in $\boldsymbol{Z}_p^*$
  - Index-calculus
  - Elliptic-curve method
  - Special number-field sieve (SNFS)
  - General number-field sieve (GNFS)

  exceptionally complicated algorithms, requiring very advanced mathematics!

- Current DLOG-solving record: $|\boldsymbol{Z}_p^*| \approx 2^{795}$ using GNFS (Heninger et al. '19)
  - Previous records: https://en.wikipedia.org/wiki/Discrete_logarithm_records

- $|\boldsymbol{Z}_p^*| \geq 2^{2048}$ typically required as a minimum today



Next week: better alternatives to Z_p^*?