
FINITE FIELDS AND DISCRETE-LOG BASED CRYPTOSYSTEMS

CIS 400/628 — Spring 2004
Introduction to Cryptography

This is based on §3.10 and Chapter 7 of Trappe and Washington

FIELDS

EXAMPLE

$(\mathbb{Z}_p, +_p, \times_p)$ acts like a miniature version of \mathbb{Q} .

DEFINITION

A field \mathbb{F} is a set with operations $+_{\mathbb{F}}$ and $\times_{\mathbb{F}}$ \ni

- ▶ The usual assoc. and comm. laws hold for $+_{\mathbb{F}}$ and $\times_{\mathbb{F}}$.

$$(a + b) + c = a + (b + c). \quad a + b = b + a.$$

$$(a \times b) \times c = a \times (b \times c). \quad a \times b = b \times a.$$

- ▶ The distributive law holds

$$(a + b) \times c = a \times c + b \times c.$$

- ▶ There is an additive identity ($0_{\mathbb{F}}$) and inverses ($-x$).
- ▶ There is a multiplicative identity ($1_{\mathbb{F}}$) and inverses (x^{-1})

EXAMPLES

$\mathbb{R}, \mathbb{C}, \mathbb{Z}_p$ (or \mathbb{F}_p).

We usually drop the \mathbb{F} subscript on things.

FIELDS, IT GETS WORSE!

- ▶ A vector space over a field \mathbb{F} is defined just like a vector space over \mathbb{R} or \mathbb{C} .
- ▶ The number of elements in the smallest basis for a v.s. is the **dimension** of the v.s.
- ▶ $\mathbb{F}' \supsetneq \mathbb{F}$ is called an **extension field**. \mathbb{F}' is automatically a v.s. over \mathbb{F} .
E.g., \mathbb{C} is a 2-dimensional v.s. over \mathbb{R} .
E.g., \mathbb{R} is a ∞ -dimensional v.s. over \mathbb{Q} .
- ▶ \mathbb{F}' is a **finite extension** of \mathbb{F} **iff** \mathbb{F}' is a finite dim. v.s. over \mathbb{F} .
- ▶ The **degree** of \mathbb{F}' is the dimension of the v.s. over \mathbb{F} .

DEFINITION Suppose \mathbb{F} is a field and $n \cdot 1 = \underbrace{1 + \dots + 1}_{n \text{ many } 1\text{'s}}$.

- ▶ \mathbb{F} has **characteristic 0** **iff** $n \cdot 1 \neq 0$, for all n .
- ▶ \mathbb{F} has **characteristic m** **iff** $m = \min(\{n \in \mathbb{Z}^+ \mid n \cdot 1 = 0\}) < +\infty$.

FIELDS AND PRIMES

FACT If \mathbb{F} has char. $n > 0$, then n is prime. (Why?)

FACT For each q , there is at most one field with q elements.

FACT If \mathbb{F} is a finite field, then $|\mathbb{F}| = p^d$ for some prime p and $d \geq 1$.

DEFINITION

- (a) \mathbb{F}^* $\stackrel{\text{def}}{=}$ the nonzero elements of \mathbb{F} .
- (b) The **order** of a $a \in \mathbb{F}^*$ is $\min\{n \in \mathbb{Z}^+ : a^n = 1\}$.
- (c) $\mathbb{F}_q \stackrel{\text{def}}{=}$ the finite field with q elements (if any).

PROPOSITION Suppose $a \in \mathbb{F}_q^*$. Then (the order of a) $| (q - 1)$.
proof on board

DEFINITION A **primitive element** α of \mathbb{F}_q is an element of \mathbb{F}_q^* with order $q - 1$.
($\therefore \mathbb{F}_q^* = \{\alpha^k : 1 \leq k \leq q - 1\}$.)

THE DISCRETE LOG

DEFINITION Suppose \mathbb{F}_q has prim. elem. α and $y \in \mathbb{F}_q^*$.

The **discrete log** of y to the base α (**notation:** $L_\alpha(y)$) is the solution for x of: $y = \alpha^x$.

EXAMPLE 6 is the discrete log of 2 in \mathbb{F}_7 .

Fact $(\alpha, x) \mapsto \alpha^x$ is easy.

BUT these seem hard:

1. Given α and y , find $x \ni y = \alpha^x$

basis of many cryptosystems

2. Given x and y , find $\alpha \ni y = \alpha^x$

basis of RSA (although $\mathbb{Z}_{p \cdot q}$ is not a field).

ALGORITHMS FOR DISCRETE LOG (FOR \mathbb{Z}_P)

METHOD	Worst-Case time complexity
Pohlig-Hellman	$O(2^{\frac{1}{2} \log p})$
Index Calculus	$O(e^{(\frac{1}{2} + o(1)) \sqrt{\log p \log \log p}})$

BIT SECURITY

Q: Computing $L_\alpha(y)$ seems hard, but are all the bits of $L_\alpha(y)$ hard to determine? (and why would you care?)

A: No! Computing $y \mapsto L_\alpha(y) \bmod 2$ is polytime. (See T&W.)

\therefore If x is our plaintext and $y = \alpha^x$ is our ciphertext then everyone can read the last bit in x !!

Q: What about the next to last bit?

LEMMA Suppose

▶ p is a prime with $p \equiv 3 \pmod{4}$.

▶ $\alpha \in \mathbb{Z}_p^* \quad \exists \quad \gamma = \alpha^{2^r} y. \quad \blacktriangleright \quad r \geq 2 \text{ and } y \in \mathbb{Z}$

Then: $\gamma^{(p+1)/4} \equiv \alpha^{2^{r-1}} y \pmod{p}. \quad \text{proof on board}$

So what?

COMPUTING DISCRETE LOGS MOD 4, CONTINUED

Suppose p is prime, $p \equiv 3 \pmod{4}$, and α is a gen. of \mathbb{Z}_p^* .

Suppose we have a cheap way of computing:

$$\text{low2bits}(y) \stackrel{\text{def}}{=} L_\alpha(y) \pmod{4}.$$

Suppose $\beta \equiv \alpha^x \pmod{p}$, where $x = \sum_{i=0}^n x_i 2^{n-i}$.

Here is how to cheaply compute x .

▶ $\text{low2bits}(\beta)$ gives us x_1 and x_0 .

▶ To find x_2 .

$$\text{Set } \beta_2 = \beta \alpha^{-(x_0 + 2x_1)} = \alpha^{2^2(x_2 + x_3 2 + \dots + x_n 2^{n-2})}.$$

$$\text{Set } z = \beta_2^{(p+1)/4}. \quad (\text{Why?})$$

CLAIM. $\text{low2bits}(z) = 2 \cdot x_2$. proof on board

▶ Iterate this process and find x_3, x_4, \dots, x_n . (See T&W.)

Therefore: If discrete log is hard to compute,
then computing low2bits must be hard too.

BIT COMMITMENT: FLIPPING COINS OVER THE PHONE

A Problematic Protocol

ALICE: Calls Bob, flips a coin, asks Bob to pick head or tails.

BOB: “Tails”

ALICE: “You loose.”

Why is Bob unhappy?

Solution 1 Locked boxes & UPS.

Solution 2 Discrete log

Setup Pick p , a prime with $p \equiv 3 \pmod{4}$ & α , a prim. elem. of \mathbb{Z}_p^*

ALICE: Chooses $x \stackrel{\text{ran}}{\in} \mathbb{Z}_p^*$, $x = (x_n x_{n-1} \dots x_1 x_0)_2$. x_1 is the coin flip.
Computes $y = \alpha^x$ (in \mathbb{Z}_p^*) and sends y to Bob.

BOB: Receives y and calls head or tails.

ALICE: Sends x to Bob.

BOB: Checks that $y = \alpha^x$.

THE ELGAMAL CRYPTOSYSTEM

Setup

Each user picks a key (p, α, a, b) :

▶ p , a prime such that the discrete log problem for \mathbb{Z}_p is hard.

▶ α , a prim. elem. of \mathbb{Z}_p^* ▶ a and b such that $b \equiv \alpha^a \pmod{p}$

plaintexts = \mathbb{Z}_p^* ciphertexts = $\mathbb{Z}_p^* \times \mathbb{Z}_p^*$

public: p, α, b private: a

BOB wants to send $m \in \mathbb{Z}_p^*$ to Alice

Alice's key = (p, α, a, b)

Chooses $k \stackrel{\text{ran}}{\in} \{1, \dots, p-1\}$.

Computes $r \equiv \alpha^k \pmod{p}$ and $t \equiv b^k m \pmod{p}$

Sends (r, t) to Alice

ALICE

Computes $t \cdot r^{-a} \pmod{p}$.

CLAIM $m \equiv t \cdot r^{-a} \pmod{p}$.

proof on board

Eves What does Eves have to know to crack this?

MERKLE-HELLMAN

The Subset Sum Problem

Given: $s_1, \dots, s_n, T \in \mathbb{Z}^+$ (s_i 's : sizes, T : target)

Question: Is there a $\{s'_1, \dots, s'_m\} \subseteq \{s_1, \dots, s_n\} \ni$
 $s'_1 + \dots + s'_m = T?$

Alt. Question: Is there $(x_1, \dots, x_n) \in \mathbb{Z}_2^n \ni \vec{x} \cdot \vec{s} = T?$

FACT: The Subset Sum Problem is NP-complete.

But there are easy special cases.

MERKEL-HELLMAN, CONTINUED

DEFINITION s_1, \dots, s_n is **super increasing** iff
 $s_j > (s_1 + \dots + s_{j-1})$, for $j = 2, \dots, n$

Example: 2, 5, 9, 21, 45, 103, 215, 450, 940

FACT: For super-increasing \vec{s} , there is an obvious linear-time greedy alg.

A not-so-good cryptosystem

Suppose \vec{s} is super-increasing of length n .

Message: $(x_1, \dots, x_n) \in \mathbb{Z}_2^n$.

$e_{\vec{s}}(\vec{x}) = \vec{s} \cdot \vec{x}$. (This is 1-1.)

$d_{\vec{s}}(y) =$ the result of the greedy alg.

Therefore: transform \vec{s} so it is not super-increasing.

MERKEL-HELLMAN, CONTINUED

SETUP

- ▶ $\vec{s} = (s_1, \dots, s_n)$ is super-increasing.
 - ▶ p , a prime $> \sum s_i$
 - ▶ $a \in \mathbb{Z}_p^*$
 - ▶ $\vec{t} = (t_1, \dots, t_n) \ni t_i = a \cdot s_i \bmod p$.
 - ▶ **Private:** \vec{s}, p, a . **Public:** \vec{t}
 - ▶ Plaintexts = \mathbb{Z}_2^n Ciphertexts = $\{0, \dots, n \cdot (p - 1)\}$
- Keys : (\vec{s}, p, a, \vec{t}) $e(\vec{x}) = \vec{x} \cdot \vec{t}$
 $d(y)$ = the solution of the subset sum problem for (\vec{s}, z) ,
where $z = a^{-1} \cdot y \bmod p$

MERKEL-HELLMAN, CONCLUDED

GOOD POINTS

- ▶ Based on an NP-complete problem
- ▶ Reasonably fast

BUT “There are easy special cases of Subset Sum”

... and the cryptosystems produces one of them.

Shamir in 1982 found a poly (in n) time algorithm for solving this scrambled version of the subset sum problem.