TOPIC: 6.857 - Computer & Network Security

DATE: 3/13/13

FILE UNDER:

PAGE: L11.1

Admin:

Note: Goldwasser & Micali Turing Award!

Today:

More "crypto math"

- Finite fields
- one-time MAC (Goln)
- Number theory:
  - divisors, gcd, inverses from gcd
  - orders of elements, Euler's Theorem
  - generators
  - discrete log problem

Finite fields:

System $(S, +, \cdot)$ s.t.

- S is a finite set containing "0" & "1"
- $(S, +)$ is an abelian (commutative) group with identity 0

group laws:

$((a+b)+c) = (a+(b+c))$ associative

$a+0 = 0+a = a$ identity 0

$(\forall a)(\exists b)\ a+b=0$ (additive) inverses $b=-a$

$a+b = b+a$ commutative

- $(S^*, \cdot)$ is an abelian group with identity 1

S^* = nonzero elements of S

group laws:

$(a \cdot b) \cdot c = a \cdot (b \cdot c)$ associative

$a \cdot 1 = 1 \cdot a = a$ identity 1

$(\forall a \in S^*)(\exists b \in S^*)\ a \cdot b = 1$ (multiplicative inverses) $b = a^{-1}$

$a \cdot b = b \cdot a$ commutative

- Distributive laws: $a \cdot (b+c) = a \cdot b + a \cdot c$

$(b+c) \cdot a = b \cdot a + c \cdot a$ (follows)

Familiar fields: $\mathbb{R}$ (reals) are infinite

$\mathbb{C}$ (complex)

For crypto, we're usually interested in finite fields,

such as $\mathbb{Z}_p$ (integers mod prime p)

Over field, usual algorithms work (mostly).

E.g. solving linear eqns:

$$ax + b = 0 \pmod{p}$$

$$\Rightarrow x = a^{-1} \cdot (-b) \pmod{p}$$ is soln.

$$3x + 5 = 6 \pmod{7}$$

$$3x = 1 \pmod{7}$$

$$x = 5 \pmod{7}$$

Notation: $GF(q)$ is *the* finite field ("Galois field") with q elements

Theorem: $GF(q)$ exists whenever

$$q = p^k, \quad p \text{ prime}, \quad k \geq 1$$

Two cases:

① $GF(p)$ - work modulo prime p

Z_p = integers mod p = $\{0, 1, \ldots, p-1\}$

$Z_p^* = Z_p - \{0\} = \{1, 2, \ldots, p-1\}$

② $GF(p^k)$: $k > 1$

work with polynomials of degree $< k$
with coefficients from $GF(p)$
modulo fixed irreducible polynomial of degree k

Common case is $GF(2^k)$

Note: all operations can be performed efficiently

(inverses to be demonstrated)

TOPIC:

DATE:

FILE UNDER:

PAGE:

Construction of $GF(2^2) = GF(4)$

Has 4 elements.

Is <u>not</u> arithmetic mod 4. (where 2 has no mult. inverse)

elements are polynomials of degree < 2 with coefficients mod 2 (i.e. in $GF(2)$):

0
1
x
$x+1$

Addition is component-wise according to powers, as usual

$$(x) + (x+1) = (2x+1) = 1 \quad (\text{coefs. mod } 2)$$

Multiplication is modulo x^2+x+1 which is <u>irreducible</u> (doesn't factor)

	0	1	x	$x+1$
0	0	0	0	0
1	0	1	x	$x+1$
x	0	x	$x+1$	1
$x+1$	0	$x+1$	1	x

$x^2 \bmod (x^2+x+1)$ is $x+1$ (note that $x \equiv -x$ coefs mod 2)

"Repeated squaring" to compute a^b in field

(Here b is a non-negative integer)

$$a^b = \begin{cases} 1 & \text{if } b = 0 \\ (a^{b/2})^2 & \text{if } b > 0, b \text{ even} \\ a \cdot a^{b-1} & \text{if } b \text{ odd} \end{cases}$$

Requires $\leq 2 \cdot \lg(b)$ multiplications in field (efficient)

$\approx$ a few milliseconds for $a^b \pmod p$ 1024-bit integers

$\approx \Theta(k^3)$ time for k-bit inputs

Computing (multiplicative) inverses:

Theorem: (For GF(p) called "Fermat's Little Theorem")

In $GF(q)$ $(\forall a \in GF(q)^*)$ $a^{q-1} = 1$

Corollary: $(\forall a \in GF(q))$ $a^q = a$

Corollary: $(\forall a \in GF(q)^*)$ $a^{-1} = a^{q-2}$

Example: $3^{-1} \pmod 7$

$= 3^5 \pmod 7$

$= 5 \pmod 7$

- How to find large (k-bit) random prime #?

  Generate & test: do $p \leftarrow$ random k-bit integer
  until p is prime

- Works because primes are "dense":

  about $2^k/\ln(2^k)$ k-bit primes (Prime Number Theorem)

  $\Rightarrow$ one of every $\approx 0.69k$ k-bit integers is prime.

- To test if a large randomly-chosen k-bit integer is prime, it suffices to test

$$2^{p-1} \stackrel{?}{=} 1 \pmod p$$

  - This works with high probability (w.h.p) for random p; doesn't work for adversarially chosen p.
  - See CLRS for Miller-Rabin primality test (randomized)
  - Technically, above gives "base-2 pseudoprime", but this is almost always prime
  - $\exists$ deterministic poly-time primality test (Agrawal, Kayal, Saxena 2002):

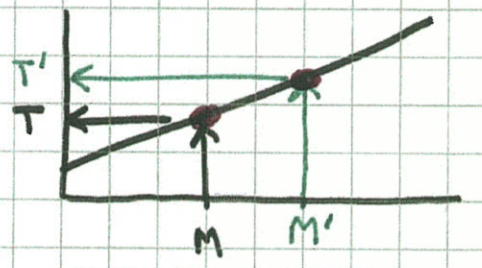
    Test $(x-a)^p = x^p - a \pmod p$ x variable

    which is true iff p is prime

    Test mod p & mod x^r-1 for small r & small a's.

One-time MAC (soln):

Idea:



$K = (a, b)$
p public
K is use-once

$$T = MAC_K(M) = ax + b \pmod p \qquad [x = M] \quad (*)$$

Need two points to determine line; Eve hears just one: (M, T)

p large prime (e.g. $2^{128} + 51$)

key $K = (a, b)$ $\quad 0 \le a < p,\ 0 \le b < p \quad (p^2$ keys$)$

Security:

If adversary hears (M, T) on the line, and replaces it with (M', T') $[M' \neq M]$, then Bob accepts with probability $1/p$.

PF: Hearing (M, T) reduces set of possible keys to those satisfying $(*)$. Nonetheless, for each possible T', there is an (a, b) satisfying both $(*)$ and

$$T = aM' + b \pmod p \qquad (**)$$

all such keys are equally likely; Eve has no way to pick correct T'.

Details:

For fixed M, M' $[M \neq M']$, fixed T s.t.

$$a M + b = T \pmod{p} \qquad (*)$$

For each T', $\exists$ exactly one key (a,b) s.t. $(*)$ and

$$a M' + b = T' \pmod{p} \qquad (**)$$

holds:

$$a = (T - T')/(M - M') \pmod{p}$$

$$b = T - a \cdot M \pmod{p}$$

Thus Eve gains no information on $T' = MAC_k(M')$ by hearing (M, T). Method is information-theoretically secure.

- True even if Eve can control M.
- Note that key K is twice as large as message M.

Divisors

- $d \mid a \equiv$ "d divides a" (evenly)
  $\equiv (\exists k)\ a = d \cdot k$
- d is a divisor of a if $d \geq 0$ & $d \mid a$
- $(\forall d)\ d \mid 0$
- $(\forall a)\ 1 \mid a$
- If d is a divisor of a & a divisor of b, then d is a common divisor of a & b.
- The greatest common divisor of a & b is the largest of their common divisors.
  [But $\gcd(0,0) = 0$ by definition.]
- Examples: $\gcd(24, 30) = 6$
  $\gcd(5, 0) = 5$
  $\gcd(33, 12) = 3$
- Def: a & b are relatively prime if $\gcd(a, b) = 1$

- Euclid's algorithm for computing $\gcd(a,b)$ $[a,b \geq 0]$:

$$\gcd(a,b) = \begin{cases} a & \text{if } b = 0 \\ \gcd(b, a \bmod b) & \text{else} \end{cases}$$

- Example: $\gcd(7,5)$

$= \gcd(5,2)$

$= \gcd(2,1)$

$= \gcd(1,0)$

$= 1$

- Running time is $\approx \lg(a) \cdot \lg(b)$ bit operations

(Polynomial running time, like multiplying.)

Theorem $(\forall a,b)(\exists x,y)\ ax+by=\gcd(a,b)$

Proof "by example" $a=7,\ b=5$

$$7 = 7\cdot 1 + 5\cdot 0$$
$$5 = 7\cdot 0 + 5\cdot 1 \quad \Big\} \text{ initial values}$$
$$2 = 7\cdot 1 + 5\cdot(-1) \quad [\text{subtract 2 eqns}]$$
$$1 = 7\cdot(-2) + 5\cdot 3$$
$$= a\,x + b\,y$$

This is the "extended version of Euclid's algorithm".

Computing modular multiplicative inverses with Euclid's extended alg:

Suppose $a \in \mathbb{Z}_p^*$ (so $1 \le a < p$ & $\gcd(a,p)=1$, p prime(?))

How to compute $a^{-1} \pmod p$?

If p prime: $a^{-1} = a^{p-2} \pmod p$

Otherwise:

Find x,y s.t. $ax + py = 1$

so $ax = 1 \pmod p$

and $x = a^{-1} \pmod p$

Example: $5^{-1} = 3 \pmod 7$