CS 7880 Graduate Cryptography October 22nd, 2015

Lecture 12: Crypto in Cyclic Groups, DL, CDH, DDH Assumptions

Lecturer: Daniel Wichs *Scribe: Schuyler Rosefield*

1 Topic Covered

- Primes
- Groups, Subgroups, Cyclic Groups
- Crypto in Cyclic Groups
- Discrete Logarithm and Diffie-Hellman Assumptions

2 Primes

Fact 1 *Distribution of Primes*

There are infinitely many of them, and we can define the function

$$\pi(x) := \text{number of primes} \leq x$$

By the Prime Number Theorem we can bound from below π with

$$\pi(x) \geq \frac{x}{3\log_2(x)} \approx \frac{x}{\log(x)} \tag{1}$$

Following (1) we can approximate the probability that a specific number is a prime with

$$Pr[x \text{ is prime} : x \in \{1, \dots 2^n - 1\}] \geq \frac{1}{3n}$$

Theorem 2 *Miller-Rabin '80, AKS '02*

We can test if a number is prime in polynomial time. With MR this process is probabalistic, and refined in AKS to be deterministic.

Corollary 1 *Sampling Prime Numbers*

We can efficiently sample a random n-bit prime in $poly(n)$ time.

The algorithm is roughly as follows:

1. sample $x \leftarrow \{0, \dots 2^n - 1\}$
2. test if x is prime, else goto 1

This has probability of success as

$$Pr[\text{algorithm does not output after } t \text{ iterations}] \leq (1 - \frac{1}{3n})^t)$$

If $t = 3n^2$ this probability is $\leq (1/e)^n$.

3 Groups, Subgroups, Cyclic Groups

Theorem 3 *Lagrange's Theorem.* If H subgroup G, then $|H| \mid |G|$.

Let $g \in G, \langle g \rangle := \{g^0 = 1, g, g^2, \ldots g^{q-1}\}$ where g^{q-1} is the last distinct element, and $g^q = 1$. Then we say that q is the order of the element g, which is the same as the order of the sub-group $\langle g \rangle$. By Lagrange's theorem $q||G|$.

If $h \in G$ then $h^m \equiv h^{m \mod q}$ where $q = |G|$.

Corollary 2 *Modular Exponentiation*

$\forall n \in \mathbb{N}, a \in \mathbb{Z}_n^*$

1. $a^x = a^{x \mod \varphi(n)} \mod n$
2. $a^{\varphi(n)} = 1 \mod n$
3. if $n = p$ is prime, $a^{p-1} = 1 \mod p$

Fact 4 *Generators*

$\mathbb{Z}_p^*$ with p prime is a cyclic group $\Rightarrow \exists g \in \mathbb{Z}_p^*$ is a generator.
i.e. $\langle g \rangle = \mathbb{Z}_p^* = \{g^0, g^1, \ldots g^{p-2}\}$

4 Crypto in Cyclic Groups

We're going to abstractly build cryptosystems over some cyclig group $\mathbb{G}$, without specifying exactly what the representation of the group is. This allows us to later instantiate these cryptosystems in a variety of different ways. We abstractly assume we have some algorithm $(G, g, q) \leftarrow Groupgen(1^n)$

1. G is a description of a cyclic group. Multiplication $g \cdot h$ and inverses g^{-1} can be computed in polynomial time in n.
2. g is a generator of G: $\langle g \rangle = G$
3. $q = |G|$ is the order of G.

E.g. One instantiation is the following.

1. Let $G = \mathbb{Z}_p^*$ where p is a random n-bit prime
2. $q = p - 1$ (notably is even)
3. g is a generator of G (we know it exists, and it turns out we can even sample it efficiently).

Note, if G is a cyclic group of order q then $(G, \cdot) \simeq (Z_q, +)$ are isomorphic. If g is a generator of G then the map $\pi : \mathbb{Z}_q \to G$ given by $\pi(x) = g^x$ is an isomorphism. Therefore, we can think of these as just different representations of the same mathematical object. However, for cryptography, we'll rely on the fact that some operations are easy given the representation $(Z_q, +)$ but (hopefully) hard given the representation $(G, \cdot)$.

We summarize three different "computational hardness assumptions" that we will use to build cryptosystems. They are listed in order from weakest to strongest.

5 Discrete Logarithm and Diffie-Hellman Assumptions

Definition 1 Discrete Lograrithm (DL) Problem
Given $g^x \mod p$, calculating x is hard
Formally $\forall$ PPT A :

$$Pr[A(G, g, q, g^x) = x : (G, g, q) \leftarrow Groupgen(1^n), x \leftarrow Z_q] = negl(n)$$

Definition 2 Computation Diffie-Hellman (CDH)
Given g^x, g^y, hard to compute g^{xy}
Formally $\forall$ PPT A :

$$Pr[A(G, q, g, g^x, g^y) = g^{xy} : G \leftarrow Groupgen(1^n), x, y \leftarrow Z_q] = negl(n)$$

Definition 3 Decision Diffie-Hellman (DDH)
Given g^x and g^y, the value g^{xy} is indistinguishable from random.
Formally:

$$(g, g^x, g^y, g^{xy}) \approx (g, g^x, g^y, h) :$$

where $(G, g, q) \leftarrow Groupgen(1^n), x, y \leftarrow Z_q, h \leftarrow G$.

Equivalently, expanding the definition of computational indsitinguihsability $\approx$, we require that for all PPT A:

$$|Pr[A(G, g, q, g^x, g^y, g^x y) = 1] - Pr[A(G, g, q, g^x, g^y, h) = 1]| \leq negl(n)$$

where the probabilities are over $(G, g, q) \leftarrow Groupgen(1^n), x, y \leftarrow Z_q, h \leftarrow G$.

Diffie-Hellman Key Agreement. We now show a protocol that allows Alice and Bob to agree on a shared secret key k via a public discussion. In other words Alice and Bob exchange messages and, even if an eavesdropper Eve sees all of the messages exchanged between them, she does not learn anything about the shared key k.

We first create public parameters $(G, g, q) \leftarrow Groupgen(1^n)$. We can think of this as being done by a "trusted party" and everyone in the future uses these parameters, or they can be chosen by (say) Alice in the first round of the protocol.

The rest of algorithm is as follows:

1. Alice generates $x \leftarrow \mathbb{Z}_q$, sends $h_A = g^x$ to Bob
2. Bob generates $y \leftarrow \mathbb{Z}_q$, sends $h_B = g^y$ to Alice
3. Alice generates $k = h_B^x = g^{xy}$ and Bob generates $k = h_A^y = g^{xy}$

The security of the protocol follows immediately from the DDH assumption, which tells us that even if Eve sees g, h_A, h_B the shared key k looks uniformly random to her.

DDH does not hold in $\mathbb{Z}_p^*$. Although the group $\mathbb{Z}_p^*$ is a good candidate for the discrete log and the CDH assumptions, it is not a good candidate for the DDH assumption.

Let the quadratic residues mod p be defined as follows:

$$QR_p = \{h : h = f^2 for f \in G\} = \{h : g^z, zeven\}$$

We can test if $h \in QR_p$ by checking $h^{(p-1)/2} = 1$. If $h = g^{2z'}$ for some z', then $h^{(p-1)/2} = g^{(p-1)z'} = 1$. On the other hand if $h = g^{2z'+1}$ for some z', then $h^{(p-1)/2} = g^{(p-1)z'+(p-1)/2} = g^{(p-1)/2} \neq 1$.

Furthermore $g^{xy} \in QR_p$ if $g^x \in QR_p$ or $g^y \in QR_p$ since xy is even if x or y is even. Therefore $g^{xy} \in QR_p$ with probability 3/4. However a random $h \leftarrow G$ is in QR_p with probability only 1/2. This lets us distinguish g^{xy} from random.

DDH in a subgroup of $\mathbb{Z}_p^*$. A Sophie-Germain prime is a number $p = 2q + 1$ such that p, q are both prime. We can set $G = QR_p$. This is a cyclic group is of prime order q and every element $g \in G$ with $g \neq 1$ is a generator of G. The DDH assumption is conjectured to hold in this group.

Algorithms for DL. There are sub-exponential algorithms to solve DL (and therefore aslo CDH, DDH) in time $2^{\tilde{O}(n^{1/3})}$ in $G = \mathbb{Z}_p^*$ (and therefore also if G is a subgroup of Z_p^*) where $p \leq 2^n$.

A generic algorithm can break DL in any group G in $O(2^{n/2})$ time (baby step, giant step) when $|G| \leq 2^n$:

1. Given generator g, calculate the giant steps $\{g, g^{\sqrt{q}}, g^{2\sqrt{q}}, \ldots g^{(\sqrt{q}-1)\sqrt{q}}\}$
2. For $h = g^x$, calculate the baby steps $\{h, h \cdot g, h \cdot g^2, \ldots h \cdot g^{\sqrt{q}}\}$
3. At some point we must get a collision between one of the baby steps and giant step so that $h \cdot g^j = g^{i\sqrt{q}}$ where we know j, i. At this point we can solve $g^{i\sqrt{q}-j} = h$.

There is also a polynomial time algorithm that solves the discrete logarithm problem using a quantum computer.