Edoardo Persichetti Lecture 3: Discrete Logs 04/03/2014

Let G be algebraic group over $\mathbb{F}_q$ of order N known. For example $\mathbb{F}_q^*$.

Definition 1 (DLP) Given $g, h \in G$ find a s.t. $h = g^a$.

Definition 2 (DDH) Given g and (g^a, g^b, g^c) determine whether or not $c = ab$.

Definition 3 (CDH) Given g and (g^a, g^b) compute g^{ab}.

Lemma 1 *DDH≤CDH≤DLP*

ElGamal Signature Scheme: G, g, N known. Use function $F : G \to \mathbb{Z}_N$ (e.g. $F(x) = x \pmod N$).

- **Keygen:** $pk = h = g^a$, $sk = a < N$.
- **Sign:** Hash the message, $H(m) \in \mathbb{Z}_N$. Choose $k < N$ at random then compute $s_1 = g^k$ and $s_2 = k^{-1}(H(m) - aF(s_1)) \pmod N$. Return signature $\sigma = (s_1, s_2)$.
- **Ver:** Check $s_1 \in \langle g \rangle$ and $0 \leq s_2 < N$ and $h^{F(s_1)} s_1^{s_2} = g^{H(m)}$.

Security: choose s_1 and try to compute corresponding $s_2 \Rightarrow$ need to solve $\log_{s_1} g^{H(m)} h^{-F(s_1)}$. Other possibility: choose s_2 and try to compute s_1, but solving $h^{F(s_1)} s_1^{s_2} = g^{H(m)} \pmod N$ in the unknown s_1 is hard. Choosing at the same time s_1 and s_2 implies trying to compute $\log_g h^{F(s_1)} s_1^{s_2}$. *More efficient variant (standardized): DSA.*

Pohlig-Hellman: Observation on the nature of N. Suppose g has order $N = \prod p_i^{e_i}$. Consider group homomorphism $\phi_i(g) = g_i$ where $g_i = g^{N/p_i^{e_i}}$ and consider $\phi_i(h) = h_i = h^{N/p_i^{e_i}}$. Note that g_i has order $p_i^{e_i}$. Can solve $g_i^b = h_i$ with algorithms for prime power order (easier). Get solution b_i and then use CRT to solve $a \equiv b_i \pmod{p_i^{e_i}}$. It is possible to extend the method to reduce to algorithms for prime order. Let $g = g_0$ of order p^e, and $h = h_0 = g_0^a$. Write $a = a_0 + a_1 p + \cdots + a_{e-1} p^{e-1}$ for $a_i < p$. Then set $g_1 = g_0^{p^{e-1}}$ and $h_0^{p^{e-1}} = \cdots = g_1^a$ and we can solve for prime order p. Next step $h_1 = h_0 g_0^{-a_0} = \ldots$ and solve $h_1^{p^{e-2}} = g_1^{a_1}$. Next step $h_2 = h_1 g_0^{-p a_1}$ etc.

Theorem 1 *Let B such that N is B-smooth. We can solve DLP with above method in $\mathcal{O}(\log^2 N + B \log N)$ group operations.*

Using exhaustive search. Hence want large prime factor of N.

Baby Step-Giant Step (Shanks): usual DLP. Let $m = \sqrt{N}$. Then there exist $i, j < m$ s.t. $i + jm = a$. Proof: $j = \frac{a-i}{m} < \frac{N}{m} = m$ so we can rewrite $g^a = h$ as $g^{i+jm} = h$. Hence $g^i = h g^{-jm}$ with $i < m$ and $j < m$. Write two lists and look for *collision*.

- **List 1:** $1, g, g^2, \ldots, g^m$
- **List 2:** $h, hg^{-m}, hg^{-2m}, \ldots, hg^{-m^2}$

For what we saw above LHS is in list 1 and RHS is in list 2 so a collision always exists. How to compute lists? Baby steps: multiplication by g in list 1. Giant steps: multiplication by $u = g^{-m}$ in list 2. Now, brute force $\mathcal{O}(N)$ while BSGS is $\mathcal{O}(\sqrt{N} \log^2 N)$ bit ops. Problem storage: $\mathcal{O}(\sqrt{N} \log N)$ bits.
Idea of collisions good, need clever way to reduce storage.

Definition 4 (Pseudorandom Walk) Function $f : S \to S$ with random behavior (explain later). The sequence of values $x_{i+1} = f(x_i)$ is a pseudorandom walk.

Pollard ρ: again usual DLP. Suppose we have a function f and a pseudorandom walk. If set is finite, at some point there will be index T s.t. $x_T = x_{T+M}$ so (draw picture) we have ρ shape. The goal is to find this collision, so need to find the cycle. In original algorithm, use *Floyd's cycle finding* method: write sequence y with $y_0 = x_0$ and $y_{i+1} = f(f(y_i))$, so in practice $y_i = x_{2i}$, hence our goal is to find i s.t. $x_i = x_{2i}$. Note that in general $x_j = x_i \Leftrightarrow i \geq T$ and $j \equiv i \pmod{M}$. In our case $2i \equiv i \pmod{M}$ implies $M|i$.
Store only current value of x_i and y_i.

Theorem 2 *If f is "sufficiently random", then $\mathbb{E}[T + M] \approx 1.2533\sqrt{N}$ (Birthday Paradox)*

So $\mathcal{O}(\sqrt{N})$ steps.

Proof Consider the event D_k that the first k values $x_0, \ldots, x_{k-1}$ are distinct. Then $Pr[D_k] = \prod_{i=1}^{k-1} Pr[x_i \neq x_j \ \forall j < i | D_i] = \prod_{i=1}^{k-1}(\frac{N-i}{N}) \prod_{i=1}^{k-1}(1 - \frac{i}{N})$ (need randomness here). Approx the last quantity with popular estimate $1 - t \approx e^{-t}$ so we have $\prod_{i=1}^{k-1}(e^{-i/N}) = e^{-(1+\cdots+(k-1)/N)} \approx e^{-k^2/2N}$ since $\sum i = 1^{k-1} i = \frac{k^2-k}{2} \approx k^2$ for large k. Now simple probability computation. Call C_k the event that there is a collision at x_k. Then $Pr[C_k|D_k] = k/N$. Now, if C'_k is the event that x_k is the *first* collision, we have $Pr[C'_k] = Pr[C_k \wedge D_k] = Pr[C_k|D_k] \cdot Pr[D_k] \approx k/N \cdot e^{-k^2/2N}$. Finally, $\mathbb{E}[T + M] = \sum_k (k \cdot Pr[C'_k]) \approx \sum_k (k^2/N \cdot e^{-k^2/2N})$.

Lemma 2 *Let $F : \mathbb{R} \to \mathbb{R}$ such that F' is continuous and $\int_0^\infty F(t)dt$ converges. Then*

$$\sum_{k=1}^{\infty} F(k/N) \approx N \int_0^\infty F(t)dt.$$

We use the lemma to approximate. Define $F(t) = t^2 e^{-t^2/2}$ so $\mathbb{E}[T + M] \approx \sum_k (k^2/N \cdot e^{-k^2/2N}) = \sum_k F(k/\sqrt{N}) \approx \sqrt{N} \int_0^\infty t^2 e^{-t^2/2} dt$ and by numerical integration we obtain $1.2533\sqrt{N}$.

When we apply it to DLP look for collisions $g^i h^j$ and $g^k h^\ell$ so $g^{i-k} = h^{\ell-j}$ and we can take roots. Problem (ask): choice of f. In original work set $x_0 = 1$ and $f(x) = gx(0 \leq x \leq q/3), x^2(q/3 \leq x \leq 2q/3), hx(2q/3 \leq x \leq q)$ for x modulo q (we are in $\mathbb{F}_q^*$), so that $x_i = g^{\alpha_i} h^{\beta_i}$. Can compute $\alpha_{i+1} = \alpha_i + 1(0 \leq x \leq q/3), 2\alpha_i(q/3 \leq x \leq 2q/3), \alpha_i(2q/3 \leq x \leq q)$ and $\beta_{i+1} = \beta_i(0 \leq x \leq q/3), 2\beta_i(q/3 \leq x \leq 2q/3), \beta_i + 1(2q/3 \leq x \leq q)$, where $\alpha_0 = \beta_0 = 0$. Stored modulo $q - 1$ since $g^{q-1} = 1$. In similar way we proceed for $y_0 = 1$ and $y_{i+1} = f(f(x_i))$ and we get $y_i = g^{\gamma_i} h^{\delta_i}$ using two repetitions. When we find collision j we obtain $g^u \equiv h^v \pmod{q}$ where $u \equiv \alpha_j - \gamma_j \pmod{q-1}$ and $v \equiv \delta_j - \beta_j \pmod{q-1}$ so $v \log_g h \equiv u \pmod{q-1}$ and if $(v, q-1) = 1$ we can just multiply by v^{-1}. Else find gcd d (Euclid), so $sv \equiv d \pmod{q-1}$ and mult. by s get $d \log_g h \equiv su \pmod{q-1}$. Solutions $\{su/d + k \cdot q-1/d, k = 0, \ldots, d-1\}$. Since d small (for Pohlig-Hellman) so check manually.

Index calculus: again usual DLP. Ideas from sieves. Want to solve $g^a \equiv h \pmod{q}$. Choose factor base $\mathbb{B}$ and solve $g^a \equiv p \pmod{q}$ for all the primes in $\mathbb{B}$. Having done that, look at hg^{-k} for $k = 1, 2, \ldots$ until we find a B-smooth quantity. In this case, we have $hg^{-k} = \prod p_i^{e_i}$ and thus $\log_g(h) \equiv k + \sum e_i \log_g(p) \pmod{q-1}$, from which we can recover the discrete logarithm we seek. How to compute the discrete logs for the primes? Look for numbers $g_j \equiv g^j \pmod{q}$. If not B-smooth, discard, else store the relation $g_j = \prod p_i^{e_i^{(j)}}$. This results in $j \equiv \log_g(g_j) \equiv \sum e_i^{(j)} \cdot \log_g(p_i) \pmod{q-1}$ so when we have $\pi(B)$ relations we can solve the system for the "unknowns" $\log_g(p_i)$ (since they are congruences, use CRT).
Complexity $L_q(1/2, 1 + o(1))$ can be lowered to $L_q(1/3, c + o(1))$ with techniques similar to Number Field Sieve.