The computational Diffie-Hellman problem

- How secure is this system? Of course if someone can compute discrete logs then this is not secure.
- The question of whether there is an algorithm for finding g^{xy} from g^x and g^y is known as the computational Diffie-Hellman problem.
- It is not known if this is an easier problem then solving the discrete log problem.
- It is known that solving the computational DH problem is equivalent to cracking the associated El Gamal cryptosystem.
- That is, suppose we can solve the computational DH problem for a field F_q and generator g. Now consider the ElGamal cryptosystem (F_q, g, b) and Alice's transmission (r, t). We know $b = g^a$ and $r = g^k$ so we can compute g^{ak}.
- But $t = mg^{ak}$ so $tg^{-ak} = m$ and we have decoded the message.

The computational Diffie-Hellman problem, cont'd

- Now suppose we want to solve the computational DH problem for a field F_q and generator g.
- Assume we can crack the ElGamal system with (F_q, g, b) for any b.
- Then let $b = g^x$, $r = g^y$ and $t = 1$.
- If we now decode (r, t) we get $m = g^{xy}$.

The Pohlig-Hellman algorithm

- Given some non-zero $b \in F_q$ and generator g, how can we find x such that $b = g^x$?
- The first observation is that if $q - 1 = p_1^{k_1} \dots p_m^{k_m}$ then by the Chinese remainder theorem, it suffices to determine what x is congruent to mod $p_i^{k_i}$.
- So let's fix some prime ℓ just to avoid possible confusion with the characteristic of the field and suppose that ℓ^k is the largest prime power of ℓ which divides $q - 1$.

Pohlig-Hellman, cont'd

- Suppose that

$$x \equiv x_0 + x_1\ell + x_2\ell^2 + \ldots x_{k-1}\ell^{k-1} \bmod \ell^k.$$

- Now notice that

$$x(\frac{q-1}{\ell}) \equiv x_0(\frac{q-1}{\ell}) + x_1(1\frac{q-1}{\ell})\ell + \ldots \equiv x(\frac{q-1}{\ell}) \bmod q-1.$$

- So we get

$$b^{\frac{q-1}{\ell}} = g^{x\frac{q-1}{\ell}} = g^{x_0\frac{q-1}{\ell}}.$$

- Now we can generate all powers

$$g^{m\frac{q-1}{\ell}} \text{ for } m = 0 \ldots \ell - 1$$

and determine x_0 by direct comparison.

Pohlig-Hellman, cont'd

- Now suppose that we have computed the sequence $x_0, x_1, \ldots, x_{i-1}$ and a sequence of elements of F_q, $b_0 = b, b_1, \ldots, b_{i-1}$. How do we continue?
- Define b_i inductively so that

$$b_i = b_{i-1} g^{-x_{i-1}\ell^{i-1}} = g^{x_i \ell^i + \ldots + x_{k-1}\ell^{k-1}}.$$

- Compute

$$b_i^{\frac{q-1}{\ell^{i+1}}} = g^{x_i \frac{q-1}{\ell}}$$

and again compare this to the list of all $g^{m\frac{q-1}{\ell}}$ to determine x_i.

Analysis of Pohlig-Hellman

- This attack is reasonably effective assuming that you can factor $q - 1$.
- It runs in time proportional to the size of the largest prime divisor of $q - 1$.
- As with the Pollard $p - 1$ algorithm, the take-away here is to make sure that $q - 1$ has some large prime factor.
- For instance, a Mersennes prime is a prime of the form $2^n - 1$. Since there are fields of size 2^n for all n, any n such that $2^n - 1$ is prime would be a good choice. There are not known to be infinitely many such primes but there are such with millions of digits.

Baby step - giant step algorithm

- Again we try to find x from b given a generator g and $b = g^x$. Let $N = [\sqrt{q-1}] + 1$.
- We make two lists:

$$
\begin{array}{lr}
\underline{\text{Baby step}} & \underline{\text{Giant step}} \\
g^0 & b \\
g^1 & bg^{-N} \\
g^2 & bg^{-2N} \\
\vdots & \vdots \\
g^{N-1} & bg^{-(N-1)N}
\end{array}
$$

- We look for a match between the two lists and if we find one, say

$$g^i = bg^{-kN} \text{ then } b = g^{i+kN}$$

and we have found x.

You can always find x

- Note $0 \leq x < q - 1 \leq N^2$ so $x = x_0 + x_1 N$ for some $x_0, x_1 \leq N$.
- This means
$$b = g^x = g^{x_0} \cdot g^{x_1 N}$$
and so
$$g^{x_0} = bg^{-x_1 N}.$$
- This algorithm takes on the order of $\sqrt{q}$ many steps.