CS 259C/Math 250: Elliptic Curves in Cryptography
Homework #3

Due November 9, 2011

Solutions must be handed in in class, handed to Mark (494 Gates), or emailed to Mark (zhandry at cs.stanford.edu) by 4pm on the due date.

Use of SAGE is allowed on any problem. If you use SAGE you must show your work by attaching your computations to the solutions you turn in. Use of LaTeX is encouraged but not required.

1. (8 points) **A variant of Schnorr signatures.** Let Gen be the key generation algorithm for the Schnorr signature scheme. Let $\mathsf{pk} = (P, Q, H)$ and $\mathsf{sk} = a$ be public and secret keys output by $\mathsf{Gen}()$. (See http://cs259c.stanford.edu/lectures/schnorr.pdf for details.) Consider the following variant of the Schnorr signing algorithm:

   Algorithm $\mathsf{Sign}'(\mathsf{pk}, M)$**:**

   1. Choose random $k \xleftarrow{R} [1, r]$ and set $R = [k]P$.
   2. Set $e = H(M \| R)$.
   3. Set $s = k + ae \pmod{r}$.
   4. Output the signature $\sigma = (e, s)$.

   (a) Describe a Verify' algorithm such that $(\mathsf{Gen}, \mathsf{Sign}', \mathsf{Verify}')$ satisfies the correctness requirement for a signature scheme.

   (b) Show that if the original Schnorr signature scheme is secure, then so is the variant $(\mathsf{Gen}, \mathsf{Sign}', \mathsf{Verify}')$. (For the definition of security, see http://cs259c.stanford.edu/lectures/boneh-shoup-digsigs.pdf.)

   Specifically, show that if there an efficient adversary $\mathcal{A}$ that can forge a signature for the variant with probability ϵ, then there is an efficient adversary $\mathcal{B}$ that uses $\mathcal{A}$ to forge a signature for the original scheme with probability ϵ. Adversary $\mathcal{B}$ plays the security game with a challenger for the original scheme. When interacting with $\mathcal{A}$, adversary $\mathcal{B}$ will act as a challenger for the variant. You will need to show the following:

   - How to transform a signature the challenger gives to $\mathcal{B}$ into a valid signature for the variant.
   - That signatures that $\mathcal{B}$ gives to $\mathcal{A}$ come from the same distribution as real signatures for the variant.
   - How to use a forgery produced by $\mathcal{A}$ to construct a forgery for the original Schnorr scheme.

You may assume that $\mathcal{A}$ and $\mathcal{B}$ have direct access to the hash function H (i.e., there are no "random oracles").

(c) Give a reason for using the variant instead of the original Schnorr signature scheme.

2. (4 points) **Breaking ECDSA.** We use the notation of Washington Section 6.6.

   (a) The case where R has x-coordinate equal to 0 is prohibited in ECDSA signatures. Show that if this check was omitted and if an adversary could find an integer k such that $[k]G$ has x-coordinate zero (for example, if P itself has x-coordinate zero and $k = 1$), then the adversary could forge ECDSA signatures for any message.

   (b) The case $s = 0$ is prohibited in ECDSA signatures since the Verify algorithm fails when s is not invertible. Show that if a signer outputs a signature (R, s) produced by the Sign algorithm with $s = 0$ then an adversary would be able to determine the private key a.

3. (3 points) **Comparable key sizes.** The best algorithms for computing discrete logs in elliptic curve groups $E(\mathbb{F}_p)$ take time approximately $p^{1/2}$. In contrast, the best algorithms for computing discrete logs in $\mathbb{F}_p^\times$ take time approximately

$$e^{1.92(\log p)^{1/3}(\log\log p)^{2/3}}.$$

   Complete the table below that shows (roughly) how large to choose finite fields for different security levels. (Express the field sizes in the form $p \approx 2^n$.)

Time to solve discrete log problem	Size of p for $E(\mathbb{F}_p)$	Size of p for $\mathbb{F}_p^\times$
2^{56}	$p \approx 2^{112}$	$p \approx 2^{383}$
2^{80}		
2^{112}		
2^{128}		
2^{192}		
2^{256}		

   In practice, recommended bit sizes (i.e., $\log_2 q$) for $\mathbb{F}_q^\times$ are about 20% larger than the values computed above. The values in this table suggest why elliptic curve cryptography has become so popular — key sizes can be dramatically shorter for the same level of security.

4. (6 points) **The Pohlig-Hellman method.** Let E be the elliptic curve $y^2 = x^3 + x$ over $\mathbb{F}_p$, where $p = 146150163733090291820368483271628301965973349331$. This curve is supersingular, with

$$\#E(\mathbb{F}_p) = p + 1 = 2^2 \cdot 3 \cdot 41 \cdot 2970531783192892110170091123407079308243848271.$$

   Let $P, Q \in E(\mathbb{F}_p)$ be given by

$$\begin{aligned} P &= (91669588454711134990458302584106127018698603872 1, \\ &\quad 58084128604615510263749292619517712 7992061729856) \\ Q &= (12996358802598536362620445384151263234296275043 20, \\ &\quad 11201399061074762464661447430694900588970776923 30). \end{aligned}$$

(These points are already entered in the SAGE worksheet "Homework3," available on the course website.) Let a be an integer in $[1, q+1]$ such that $Q = [a]P$.

(a) Without using a computer, show that $E(\mathbb{F}_p)$ is cyclic. You may assume the result of problem 3(e) on Homework #2. (*Hint:* -1 is not a square mod p.)

(b) Compute a mod 2.

(c) Use the result of (4b) to compute a mod 4.

(d) Compute a mod 3.

(e) Compute a mod 41.

(f) Suppose you happen to know that

$$a = 186431622114179175383708717890844937406957091 0 \mod 29705317831928921101700911234070793082438482 71.$$

Using this information and the results above, compute a. (The SAGE command `CRT` will be helpful here.) Use SAGE to confirm that $[a]P = Q$.

5. (4 points) **Discrete log in an interval.** Suppose one is given an point P of order r on an elliptic curve $E(\mathbb{F}_q)$ and integers $b, w \in [0, r]$. The *DLP in an interval of length* w is: Given P, Q such that $Q = [a]P$ for some $a \in [b, b+w]$, find a.

(a) Describe a baby step-giant step algorithm for computing a using approximately $\sqrt{w}$ group operations and $\sqrt{w}$ group elements of storage.

(b) Suppose one is given $P, Q \in E(\mathbb{F}_q)$ and integers t, m such that $Q = [a]P$ for some integer a satisfying $a \equiv t \pmod{m}$. Show how to reduce this problem to the problem of solving a DLP in an interval of length approximately r/m.

6. (14 points) **Implementing Pollard's rho method.** Let $p = 1048583$ and $E/\mathbb{F}_p$ be given by $y^2 = x^3 + 183801x + 823458$. Let $P, Q \in E(\mathbb{F}_p)$ be given by

$$P = (580655, 963078), \qquad Q = (774292, 954498).$$

Let a be the discrete logarithm of Q to the base P; i.e., $Q = [a]P$.

The SAGE worksheet "Homework3" contains a function `collision_search` that runs the Pollard rho algorithm, stores all of the computed points, and searches the list for a collision at each step. Using this code as a base, write programs that do the following:

(a) Use Floyd cycle finding to find a collision with almost no storage: at each iteration use P_i and P_{2i} to compute P_{i+1} and P_{2i+2}. Repeat until $P_i = P_{2i}$. Output the discrete logarithm a and the total number of times the `walk` function is called.

(b) Use distinguished points to find a collision with a small amount of storage: at each iteration check to see if the x-coordinate of P_i is 0 mod d, where d is an input to the function. If so, we say P_i is a distinguished point, and we store P_i in a list and start over with a new random P_0. Repeat until a distinguished point is found that is already in the list. Output the discrete logarithm a, the number of distinguished points found, and the total number of times the `walk` function is called. Test your function using $d = 32$.

(c) There is a possibility that the walk in (6b) will enter a cycle that has no distinguished points, in which case the algorithm will not terminate. Add to the algorithm a way of detecting and escaping such cycles at little additional cost.

(d) Run `collision_search` and each of the algorithms from (6a) and (6c) 1000 times and compute the average number of iterations. (Use $d = 32$ in (6c).) Write the number of iterations as $c\sqrt{N}$ where N is the order of P. How do these values compare with the theoretical values of $c = 3.093$ (for (6a)) or $c = 1.253$ (for (6c))?

(e) (Bonus, 2 points.) In your algorithm from (6c), how does the average number of steps vary as you vary the probability $1/d$ of hitting a distinguished point?

7. (10 points) **Improving Pollard rho.** Let P, Q be points on $E(\mathbb{F}_p)$ with $Q = [a]P$. Suppose that when running our pseudorandom walk f, instead of looking for a collision where $P_i = P_j$, we look for a collision where the x-coordinates of P_i and P_j are the same.

If we wish to use this technique to find cycles, our walk function f has to be a function of x-coordinates only. The standard way to address this issue is to use the random walk function

$$\hat{f}(x,y) = \begin{cases} f(x,y) & \text{if } y < -y \pmod p \\ f(x,-y) & \text{if } y > -y \pmod p \end{cases}$$

(Testing whether $a < b \pmod p$ means representing a and b as integers in $[0, p-1]$ and comparing these integers.)

Fix a small s and points $M_0, \ldots, M_{s-1}$. Suppose $f(x,y)$ is defined by $f(x,y) = (x,y) + M_{x \bmod s}$. Let $\hat{P}_0 = P$ and $\hat{P}_{i+1} = \hat{f}(\hat{P}_i)$.

(a) Show that for any $R \in E(\mathbb{F}_p)$, we have $\hat{f}(R) = \hat{f}(-R)$. Thus $\hat{f}$ is a function of x-coordinates only.

(b) Suppose that $\hat{P}_i = u_i P + v_i Q$ and $\hat{P}_j = u_j P + v_j Q$ for known values of u_i, v_i, u_j, v_j. (It is easy to keep track of these values when using the function $\hat{f}$.) Suppose that $\hat{P}_i$ and $\hat{P}_j$ have the same x-coordinates. Show how to use this information to compute (with high probability) the discrete log a.

(c) Let $c\sqrt{N}$ be the average number of iterations needed to find a collision of points in the usual rho algorithm using the function f. Estimate the average number of iterations needed to find a collision in x-coordinates only using the function $\hat{f}$.

(d) Suppose $\hat{P}_i = (x_i, y_i)$ and $\hat{P}_{i+1} = (x_{i+1}, y_{i+1})$ satisfy

$$y_i < -y_i, \qquad y_{i+1} > -y_{i+1}, \qquad x_i \bmod s = x_{i+1} \bmod s.$$

Show that $\hat{P}_{i+2}$ has the same x-coordinate as $\hat{P}_i$, so there is a cycle of size 2.

(e) Write $\hat{P}_i = (x,y)$. Show, under the assumptions that the function $\hat{P}_i \mapsto (x \bmod s)$ is perfectly random and that $y < -y$ with probability $1/2$, that the probability that $\hat{P}_i$ leads to a cycle of size 2 is $1/2s$.

The problem of avoiding such "useless cycles" is a tricky one, and until recently the methods of solving the problem required more computation than was saved by searching on x-coordinates only!

8. (10 points) **Pairings.** Let $E/\mathbb{F}_q$ be an elliptic curve and let n be a prime not dividing q. Suppose that $E(\mathbb{F}_q)$ contains a point of order n. Let ϕ be the q-Frobenius endomorphism.

   (a) Show that ϕ acting on $E[n]$ has eigenvalues of 1 and $q \pmod n$. (*Hint:* Washington Proposition 3.15.)

   Now let $\hat{e}\colon E[n] \times E[n] \to \mu_n$ be a pairing that satisfies properties (1) and (6) on p. 87 of Washington. (In particular, $\hat{e}$ is not assumed to satisfy properties (2)–(5).)

   (b) Let P and Q be points of order n that are eigenvectors for ϕ with eigenvalues 1 and q respectively. Show that if $\mu_n \not\subset \mathbb{F}_q$, then $\hat{e}(P,P) = 1$ and $\hat{e}(Q,Q) = 1$.

   A *distortion map for* P is an endomorphism α such that $\alpha(P) \notin \langle P \rangle$. Suppose $P \in E(\mathbb{F}_q)$ has order n and α is a distortion map for P. Let $\hat{e}$ be as above and assume $\mu_n \not\subset \mathbb{F}_q$.

   (c) Show that $\hat{e}(P, \alpha(P)) = \hat{e}(\alpha(P), P)^{-1}$. (*Hint:* consider the endomorphism $1 + \alpha$.)

   (d) Show that $\hat{e}(T,T) = 1$ for all $T \in E[n]$.

   (e) Show that $\hat{e}(S,T) = \hat{e}(T,S)^{-1}$ for all $S, T \in E[n]$.

   Since supersingular curves always have distortion maps, it follows from these results that in most cases the modified Tate pairing (Washington Section 3.4) is in most cases antisymmetric on supersingular curves

9. (6 points) **Embedding degrees of supersingular curves.** Let $E/\mathbb{F}_q$ be an elliptic curve with $\#E(\mathbb{F}_q) = q + 1 - t$. Let $r \geq 5$ be a prime dividing $\#E(\mathbb{F}_q)$. The *embedding degree of* E *with respect to* r is the minimal k such that $E[r] \subset E(\mathbb{F}_{q^k})$. By Washington Corollary 3.11, this implies that r divides $q^k - 1$.

   Assume that $E[r] \not\subset E(\mathbb{F}_q)$. Show the following:

   (a) If $t = \pm\sqrt{q}$, then E has embedding degree at most 3 with respect to r. (This case can only occur if q is a square.)

   (b) If $t = \pm\sqrt{2q}$, then E has embedding degree at most 4 with respect to r. (This case can only occur if q is a power of 2.)

   (c) If $t = \pm\sqrt{3q}$, then E has embedding degree at most 6 with respect to r. (This case can only occur if q is a power of 3.)

   In addition, we already know that:

   - If $t = 0$, then E has embedding degree 2 with respect to r. (Washington Proposition 5.3.)
   - If $t = \pm 2\sqrt{q}$, then E has embedding degree 1 with respect to r. (Washington Exercise 4.14; this case can only occur if q is a square.)

   Since these are all the possible values of t for supersingular curves (see Washington Theorem 4.3), we conclude that if E is supersingular, then E has embedding degree at most 6 with respect to r.