Final

Take home due 11 May 2010 at 3 AM

There are seven problems each worth five points for a total of 35 points. Show all your work, partial credit will be awarded. Space is provided on the test for your work; if you use a blue book for additional workspace, sign it and return it with the test.

This is an open book take-home final. It is released on Sunday, May 9-th, and due at 3 AM on Tuesday morning, May 11-th.

Name: _____

Problem	Credit
1	
2	
3	
4	
5	
6	
7	
Total	

1. **Fast multiples:**

   Describe a polynomial-time algorithm for calculating nP, where n is a positive integer and P is a point on an elliptic curve.

   Describe how to extend the algorithm to the case where n is an integer, not necessarily positive.

2. **Oblivious transfer:**

   In an oblivious transfer protocol, the sender sends two encrypted messages, $E_1(M_1)$ and $E_2(M_2)$ to the chooser. The chooser can pick one and exactly one message to decrypt. The chooser learns nothing about the message it did not pick. The sender does not learn which of the two messages the chooser picked.

   Here is an Discrete Log based implementation of oblivious transfer. Note that the chooser begins by sending a message to the sender. The sender responds with two encryptions based on the information sent.

   (a) Sender and chooser agree on a prime p and a generator α of Z_p^*. Chooser also announces an element $\beta \in Z_p^*$. They also agree on a hash function H which takes elements of Z_p^* to strings over $\{0, 1\}$ for ex-or'ing with messages.

   (b) Chooser picks a random r and sends to chooser either $\gamma = \alpha^r$ or $\gamma = \beta/\alpha^r$.

   (c) Sender receives γ. Sender choses random r_1 and r_2 and sends $E_1 = \big(\alpha^{r_1}, H(\gamma^{r_1}) \oplus M_1\big)$ and $E_2 = \big(\alpha^{r_2}, H((\beta/\gamma)^{r_2}) \oplus M_2\big)$.

   (d) Chooser receives both E_1 and E_2.

   - If the chooser sent $\gamma = \alpha^r$, then chooser computes M_1 from $E_1 = \big(\alpha^{r_1}, X_1\big)$ by $M_1 = X_1 \oplus H\big((\alpha^{r_1})^r\big)$.
   - If the chooser sent $\gamma = \beta/\alpha^r$, then chooser computes M_2 from $E_2 = \big(\alpha^{r_2}, X_2\big)$ by $M_2 = X_2 \oplus H\big((\alpha^{r_2})^r\big)$.

   Based on this protocol, describe an elliptic curve based method protocol for oblivious transfer. Explain why it is correct and secure:

   (a) Why does the chooser learn the message of it's choice.

   (b) Why (under what assumption) does the chooser not learn the message it does not choose.

   (c) Why does the sender not learn which message the chooser chose.

   Argue this for your elliptic curve protocol, although the argument will mostly work for either the elliptic curve or the discrete log versions, as they are abstractly based on similar principals and equations.

3. **Pohlig-Hellman:**

   Bob wants Alice to commit to the choice of a number, 1 through 5. Knowing something about cryptography, they choose a random prime p and a generator $\alpha \in Z_p^*$. Alice picks a random r such that $r \pmod 5$ is equal to her choice. She announces $\alpha^r \pmod p$ as her commitment to her choice.

   Bob notices, however, that $p - 1$ is divisible by 5. Bob is able to discover Alice's choice. Describe how he does it.

4. **Pohlig-Hellman, calculations:**

   This is a continuation of the previous problem. I am thinking of a number x between 1 and 5. I pick a random r and set $y = 5r + x$. I tell you that $3^y = 25 \pmod{31}$.

   What is x?

   First, do it brute force showing all powers of 3 modulo 31, and match the power y that yields $3^y = 25 \pmod{31}$.

   Then show and explain the Pohlig-Hellman approach, since enumerating all powers will take too long if p is large.

5. **Zero-knowledge:**

   Suppose the prover Peggy wants to prove knowledge of x, the square root of $X = x^2 \pmod{n}$ where n is the product of two distinct primes p and q.

   Peggy can present the verifier Vic with a random square R. Vic can then ask for either the square root of R or of XR. Since R is random, Vic does not learn the square root of X, but assuming Peggy can answer both the square root of R and XR, then the quotient of the two is the square root of X.

   Suppose we know that Vic will ask, in this order:

   (a) for the square root of R,

   (b) the square root of XR,

   (c) the square root of XR,

   (d) and the square root of R.

   Suppose $n = 4757$ and $X = 2614$. Give a sequence of four numbers that Peggy can tell Vic so that Peggy can answer Vic's four questions.

   Show work to confirm that your sequence of numbers were, or could have been, generated without knowledge of the square root of X.

   Hint: If you know Vic will ask for the square root of the given random number R, generate R as $R = r^2 \pmod{n}$ of a random r that you choose. If however Vic will ask for a r such that $r^2 = XR$, solve that equation to show what R Peggy should send in order to correctly answer with r.

6. **Square roots:**

   Find the square root of $X = 2614 \bmod 4757$. Show all work.

   Credit for brute force (that is, get a computer to square all numbers until one turns out right). Full credit for a fully algorithmic approach.

   Hint: To factor 2614 using something other than brute force, you can use the fact that $2413^2 = 1 \pmod{4757}$.

   Hint: Both factors of 4757 are $3 \pmod 4$.

7. **Zero-knowedge, *continued:***

   Referring back to the previous two questions ...

   Suppose Vic double-crosses Peggy and for the final question, rather than asking for R, asks for XR.

   Now that you know the square root of X, give the correct answer to Vic. Show how Peggy's new answer combined with her original answer would allow Vic to find the square root of X with a simple (polynomial time) calculation if he had both answers.