CS 380S - 0x1A Great Papers in Computer Security
Fall 2011

Homework #4

Due: 2pm CST (in class), December 6, 2012

NO LATE SUBMISSIONS WILL BE ACCEPTED

YOUR NAME: _____

Collaboration policy

No collaboration is permitted on this assignment. Any cheating (*e.g.*, submitting another person's work as your own, or permitting your work to be copied) will automatically result in a failing grade.

Homework #4 (30 points)

Problem 1

Recall the oblivious transfer protocol between the Sender (S) and the Chooser (C) based on the hard-core predicate of a one-way trapdoor permutation. The Sender chooses a one-way trapdoor permutation F (let T be the trapdoor, and H the hard-core predicate of F). Let $b_{0,1}$ be the Sender's input bits, and let c be the bit indicating the Chooser's choice.

The protocol proceeds as follows:

S	$\rightarrow$	C	F	
S	$\leftarrow$	C	y_0, y_1	where $y_c = F(x_c)$ for a random x_c; $y_{\overline{c}}$ is random
S	$\rightarrow$	C	$m_0 = b_0 \oplus H(T(y_0))$, $m_1 = b_1 \oplus H(T(y_1))$	

The Chooser computes b_c as $m_c \oplus H(x_c) = (b_c \oplus H(T(y_c))) \oplus H(x_c) = (b_c \oplus H(T(F(x_c)))) \oplus H(x_c) = (b_c \oplus H(x_c)) \oplus H(x_c) = b_c$.

Problem 1a (4 points)

Suppose the Sender is *malicious* rather than semi-honest. Is the above protocol secure? If not, explain precisely what a malicious Sender can do to make his view of the real-world protocol unsimulatable in the ideal world.

Problem 1b (4 points)

Suppose the Chooser is *malicious* rather than semi-honest. Is the above protocol secure? If not, explain precisely what a malicious Chooser can do to make his view of the real-world protocol unsimulatable in the ideal world.

Problem 2 (4 points)

Suppose Alice and Bob are evaluating a NAND gate using Yao's "garbled circuits" protocol and the Naor-Pinkas oblivious transfer protocol.

Suppose that (1) Alice is malicious rather than semi-honest, and (2) Alice uses 0 as her input bit. How can she learn Bob's input bit? Explain in detail.

Problem 3 (3 points)

How are "onions" (in the sense of onion routing, *i.e.*, a message wrapped in layers of public-key encryption, one per each router on the path) used in Tor? Explain your answer.

Problem 4 (3 points)

In this problem, we consider *online* query monitoring and auditing, *i.e.*, instead of publishing a perturbed database, the database owner interactively receives queries and, for each query, decides whether it is safe to answer it using some auditing or monitoring algorithm.

Let $X = \{x_1, \ldots, x_n\}$ be the database. Each element x_i is associated with some integer value v_i. The questioner specifies any subset $X' \subseteq X$ as the query.

If the query is safe, the response is the highest value among those associated with the elements of the requested subset. Unsafe queries are denied. A query is unsafe if the responses to all previous queries, taken together with the response to the current query, would reveal the value associated with some element of the database X.

Give an example of a database X and a sequence of queries that, if processed by this auditor, completely reveals the value associated with some element of X.

Problem 5

D is the dataset containing annual salaries of all UT employees. $bsdcount(D)$ returns the number of entries in D that are greater than $\$1,000,000$; $max(D)$ returns the maximum salary in the dataset.

Let San be the standard Laplacian mechanism for ϵ-differential privacy. Given any function f, San generates random ξ from the Laplacian distribution with variance that depends on the sensitivity of function f and the privacy parameter ϵ, and returns $f(D) + \xi$.

Problem 5a (4 points)

What is the sensitivity of *bsdcount* and max? State all assumptions you needed to calculate the answers.

Problem 5b (4 points)

For the "same level of privacy," which function requires "more noise" to be added? Given a function, how does the "noise distribution change" in order to achieve "higher level of privacy"? Your answers should make precise all terms in quotes.

Problem 5c (4 points)

Let $\epsilon = 0.001$, let $p = 0.01$ be your *a priori* probability that Bevo makes $\$10,000$ a year, and let p' be the probability after learning the differentially private values of *bsdcount* and *max*. What is the maximum value of p'?