CS-523 Exam 2021

July 2, 2021

1 Anonymous Communication and Censorship

The workers of XenaWarrior, an online retailer, are trying to unionize. XenaWarrior is known to look into their employees electronic communications at work, and the unionizers suspect that XenaWarrior also has a deal with the NSA to check on some employees' electronic communications beyond the company premises – both on the network and by looking at email and messaging providers. In one of their weekly meetings, one of the unionizers suggests that it would be better for the group to protect their electronic communications using anonymous communications to avoid spying from their employer.

The workers contact you to recommend them a good anonymous communication network for their daily electronic communications.

Part 1. From the four systems seen in the class, which one would you recommend if the unionizers are only worried about relationship anonymity?

Part 2. The workers ask if the system you recommended would also help them to avoid XenaWarrior censoring their communication (i.e., Xena Warrior cannot prevent communications among the unionizers). What would you answer?

For both questions, justify your answer. Make sure you specify the adversarial model you are considering (who and with what capabilities). Explain why, under this adversarial model, the system you are recommending does or does not provide the properties the workers require. If the recommended system fulfills the requirements, explain how the unionizers need to use the system to guarantee anonymity and anti-censorship.

Note that the unionizers meet in person every week. If you need it, you can use this fact in your answer.

2 Machine Learning Privacy

The technology company Pear is the largest producer of smartphones in Switzerland and recently launched a new program called Pear4Science that allows scientists to run large-scale user studies on Pear's mobile phone platform. Pear advertises the platform as highly privacy-preserving.

Part 1. A group of EPFL researchers want to test how well Pear4Science works for federated learning by running a simple analysis task. The researchers want to learn whether there is a correlation between users' coffee consumption and how social they are, i.e. they want to learn a linear coefficient a that given a users coffee consumption x predicts the number of social interactions of this user y as $ax = y$. The researchers set up a gradient-based federated learning algorithm that finds the linear coefficient a that minimises the loss function $\mathrm{argmin}_a L(x, y; a) = ax - y$.

Every day, users who participate in the study enter into an app: how many cups of coffee they drank, and how many different people they saw. The algorithm then locally computes the gradient of the loss function as $g = \frac{\partial}{\partial a} L$ and sends it back to the Pear4Science server. The server aggregates the gradients of all users and updates its estimate of the optimal coefficient as $a_{i+1} = a_i - \sum_n g_n$ where g_n is the gradient provided by user n.

Pear promises that this scheme provides full privacy to any user who participates in the study. Concretely, they state that no data about individual users ever leaves their phone.

Question. Is this true? If your answer is no, describe what a potential adversary can learn and how (describe who this adversary is and their capabilities). If your answer is yes, justify why the system is privacy-preserving.

Part 2. The EPFL researchers are satisfied with the results of their first study and decide to test Pear4Science for user surveys. In the new study, they want to analyse whether certain demographic subgroups in Switzerland are more likely to suffer from cardiovascular diseases than others. Users answer a set of questions about their demographics, such as their age, gender, or the country they were born in, and indicate whether they have ever been diagnosed with certain diseases. The researchers consider this data to be highly sensitive but want to make sure that many users participate. Therefore, they advertise the study as highly privacy-preserving because they use Pear4Science' differentially private algorithm to collect the survey results. When using this algorithm, the Pear4Science platform first aggregates all users answers, and computes the counts of users with certain diseases broken down by their demographics. It then adds differentially private noise to each count, and publishes the results.

Question. (a) Validate whether this algorithm provides good privacy for users' sensitive health data. If you think there is any privacy leakage describe the leakage and a concrete privacy threat model that would violate user privacy. (b) Describe one potential disadvantage for the utility of the collected data that might result from the use of differential privacy in this study.

3 Online Tracking

`BestAds.com` and `AwesomeAnalytics.com` are two providers of behavioral ads. They come to an agreement that they will share with each other the user browsing data they collect. This will allow them to build even more accurate profiles and serve better targeted ads!

Julia uses the Lithium browser which does not block third party cookies or fingerprinters. Lithium implements a special type of FLoC called per-observer FLoC (POFLoC). Remember that in FLoC, users are assigned a FLoC ID representing the cohort the user belongs to.

In POFLoc, users have one FLoC ID per tracker, i.e., two trackers on the same site would obtain different FLoC IDs for Julia, but a given tracker obtains the same ID for Julia in all webs this tracker is present on. For example, if `observer1.com` and `observer2.com` are present on the same site `example.com`, and query for the FLoC cohort of the Julia, they obtain different values. At the same time, if `observer1.com` is present on both `anotherexample.com` and `example.com`, it will get back the same ID for Julia from both sites.

Part 1. Julia visits a site on which both `BestAds` and `AwesomeAnalytics` are present. Describe one method by which `BestAds` can inform `AwesomeAnalytics` of the ID `BestAds` sees for Julia. This method should enable `AwesomeAnalytics` to link Julia's `BestAds` ID to Julia's `AwesomeAnalytics` ID. Your description should include all the steps required to perform the linking.

Part 2. On the recommendation of her friends, Julia decides to switch to the WaterWolf browser which blocks third-party cookies but still uses POFLoC. Does this switch impact the method you propose in for `BestAds` and `AwesomeAnalytics` to link the IDs? If your answer is yes, describe an alternative method by which `BestAds` and `AwesomeAnalytics` can circumvent the new protections. If your answer is no, justify why.

4 Private Set Intersection with (Garbled) Circuits

In this question we will construct a special private set intersection (PSI) protocol called *X-PSI*. In an X-PSI protocol, a client with set X and server Y collaborate so that the client learns whether any common elements exists between sets X and Y, i.e., $|X \cap Y| > 0$. The server learns nothing.

In this question we use Yao's garbled circuits to build a X-PSI protocol for the special case where the domain D of set elements is small and ordered: $D = \{d_1, \ldots, d_n\}$. Both the elements in the domain D and their order are known to the client and server. The structure of the X-PSI protocol is as follows:

1. The client and server construct bit vectors $\vec{x} = (x_1, \ldots, x_n)$ respectively $\vec{y} = (y_1, \ldots, y_n)$ to represent their sets. Element x_i (respectively y_i) is 1 iff $d_i \in X$ (respectively $d_i \in Y$).
2. The server creates a garbled circuit C that operates on $\vec{x}$ and $\vec{y}$ and sends it to the client.
3. The client interacts with the server to evaluate the circuit C on the client input $\vec{x}$ and the server input $\vec{y}$. The client learns the output $z = C(\vec{x}, \vec{y})$.
4. The client computes whether there are common elements in sets X and Y based on z.

Part 1. (a) Specify how the server builds the garbled circuit C so that the protocol achieves its goal: the client learns whether there is any intersection between X and Y (and no more).

To specify the circuit, describe all gates in the circuit and how their input and output should be wired. You should specify a circuit that operates on the inputs $\vec{x} = (x_1, \ldots, x_n)$ and $\vec{y} = (y_1, \ldots, y_n)$. The circuit should be described with the following syntax that represents three types of gates:

```
out = AND(in1, in2)
out = OR(in1, in2)
out = NOT(in)
```

No other gate and *no other syntax* can be used. For example, you could express the function $a \vee (b \wedge \neg c)$ as:

```
nc = NOT(c)
t = AND(b, nc)
out = OR(a, t)
```

(b) Specify how the client uses the *output* of the circuit C to determine if there is any common element between sets X and Y.

Part 2. Suppose the client is honest-but-curious in your protocol (i.e., the protocol above instantiated with your circuit from Part 1). Can the client learn anything more about the server's set than the existence of common elements between the client's and server's sets? If yes, give an example. If not, argue why not.