FAQ

Task 1:

========== question asked on 05/25/2017 ==========

1. About the input/output behavior.

We assume there are n centers. and each has m_i record for i = 1, 2, .. n during the evaluation. Then, records can be sent to the servers in several batches (e.g., 100 records per batches) at different time. Here is (I guess) an example of what would happen:

- **Center 1 sends records A,B,C to the servers;**
- **Center 2 sends records A,C,D to the servers;**
- **Servers should respond with "#2, {A,C}", meaning that in the new records from Center 2, A and C already exist before, so Center 2 should delete them to remove duplicates;**
- **Center 3 sends records A,D,E,F to the servers;**
- **Servers should respond with "#3, {A,D}", so that Center 3 will remove A and D. Here I assume that Center 2 has already removed its duplicates, so the servers wouldn't include C in the answer.**
- **…**

(a) Is the guess correct? If not, what is wrong?

> Yes, your guess is correct.

(b) For Center i, are the m_i records all given to the center at the beginning, or they might be given at different time (in the former case, the center could do some preprocessing)? In either case, is it guaranteed that all the m_i records are de-duplicated as given?

> Within each center, you can assume the records are de-duplicated before sending the servers. In other words, each center will de-duplicate their records before sending them to the server and the task of the servers (and the challenge) is to de-deduplicate records between different centers.

(c) What is the format of the output? I assume that the output per request consists of a center number and a set of IDs. Should it also be specified where the duplicates are from? In the above example, should the second response also specify "A from Center 1&2, D from Center 2"?

> You should give the output to each center, which can be either empty (indicating no duplication) or a set of IDs in the records from the center that are duplicated from the other centers. In a real scenario (which you do NOT need to implement), the centers will be prioritized such that only the output to the centers with new batches of data will be sent.

Do you mean that in the example the first response should be {ID of A, ID of C} and sent to all centers (not just Center 2)?

A separate questions: In the example is it possible that Center 3 sends the request before the servers respond to Center 2's request? Or can we assume this never happens (namely, a new request always comes after the servers have responded to all previous requests)?

2. About the security model.

We can assume that every party is honest but curious, and that the servers are honest majority.

(a) Can the centers collude with each other? If so, then what is the maximum number of centers that can collude together?

> Note the as each center may infer the presence of a record in one of the other centers if it is given the output that the record is duplicated. We hope there is no additional information leak than this. Does this make sense?

Yes it explains clearly what information can be revealed to the centers. But what is the collusion model? Can two or more centers collude? And can they collude with the server(s)?

> No center will collude with each other, or collude with servers.

(b) Can one server collude with one or more centers?
> Yes, for a m-server protocol, we allows for at most m/2-1 servers to collude. (the servers are honest majority)
But it was claimed in the website that "the servers do not collude with each other". I suppose it means no two servers can collude?
> I want to modify my previous response by that we allow at most half of the k servers may collude with each other. We will modify the FAQ to reflect this.

(c) What information can be revealed to the centers? Is the output revealed to all the centers, or only to the requested center? Should the output be in permuted (or alphabetic) order to avoid additional information leakage?
> please see my answers above.

(d) What information can be revealed to the servers? Is the output be revealed? Can any information of the records (e.g., ID, hash(ID), name, date-of-birth, etc.) be revealed? If the output is not revealed, can any other information be leaked? For instance, could the number of duplicated records be leaked? In the above example could it be leaked to the servers that all the three centers have a common record?
> All input (hash ID etc) should be kept on a server and cannot be revealed to the other servers. The number and IDs of duplicated records can be revealed to all servers so that the servers holding the records knows, based on its own records, which center has a duplicated records. However, the associated centers of these duplicated records cannot be reveal to all servers. As a result, if a record is common in all centers, each server knows the corresponding record is duplicated but does not know if any of the other centers has the duplicated record or not.
Do you mean that every record (consisting of its ID, Name, DOB, etc.) can be revealed to at most one server?
In the above example for the first query the two records {A, C} (consisting of all the information in the records) can be revealed to all the servers? So every server that holds the record will notice A is a duplicate?

> I am saying the ID of the record (but not other information such as Name, DOB etc) will be revealed to all servers so that the server holding the record ID will notice it is a duplicate.

3. About the dataset samples. I take the first dataset (random-batch-1per) as an example.
The dataset is generated in the following way: First, generate all the records randomly; Second, remove all the duplicate records; Third, pick a random file (file781 in the sample), and pick 99 records at random from that file to randomly copy to other files.
(a) It is claimed in the webpage that "~0.1% duplicated records between most centers, but some pairs of centers may contain up to 10% duplications". And it is clarified in FAQ that "the duplication percentage is in the interval [0.1, 10]% between any pair of centers with more concentration around the 0.1%". But in the sample, duplicates only appear between file781 and some other files (and the duplication percentage is ~0.01%). There is no duplicates between any other pair. It doesn't match the requirement.

> Sorry for the mis-understanding. The example file is designed this way to mimic the real scenario that often time there is only one center has cumulated enough number of records to be compared with the existing records in all other centers that have be previously de-deduplicated. But we ask the participating teams not to make this assumption that only one center has potential duplicated records so that it can be applied to a broader range of cases.

(b) The last line of file746 consists of 2 records. Is it in the wrong format?

> Sorry this is a formatting issue. We will fix it. Each line should contain only one record.

(c) When copying the records from file781 to other files, the same ID is used. Can we assume that the same record will have a unique ID (even in different centers)? If not, then what is the answer to 1(c)?

How about this question? If the same record may have different IDs, then what would be the output?

> Each record has a unique ID. Duplicated records (from different centers) have different IDs. They will know their record is a duplicate based the output ID.

4. About the communication model.

(a) What is the communication model? I assume there is communication between every pair of servers, and between every server and every center. Is there communication between every pair of centers? Is every pair of parties connected by "100Mbps network"?

> The communication between the each center and the servers is limited by sending the data (hash strings & IDs of the records) from the to the server, and the output from servers to each center. We do NOT plan to evaluate the communication cost between centers and servers as it will be a one time communication (but if the input data is too big, this could be an exception, although we do not expect that). However, we will evaluate the communication between servers -- as you assumed, there can be communications between every pairs of servers depending on the protocol.

Is there any communication between centers? Do they even know each other?

> No communication between centers. Centers only communicate with one (or more) servers.

(b) Are all the parties guaranteed to be online and can respond immediately during the entire protocol?

> All servers are guaranteed online during the computation and thus will respond immediately.

5. About the evaluation standards.

The communication cost, computation cost, and overall turnaround time will be evaluated.

(a) Do we simply count the total communication / computation cost? Do we also measure maximum or average communication / computation cost? Do we treat them differently for different types? For example, communication between servers is cheaper than communication between a server and a center? Computation at server side is cheaper than center side?

> We will count the communication and computation cost separately. We will measure multiple computation times on the same computation and use the average cost for the comparison.

(b) Do we also measure communication rounds?

> Yes. Fewer rounds are preferred.

(c) I assume the servers respond to every request in an online fashion (rather than responding at the very end). Do we add up the response time for all the requests, or take the average, or measure the maximum response time? Which one do we care the most?

> Yes. Responding time will be measured and the average time will be used in the comparison.

(d) Do we measure memory cost? Do we simply count the total memory cost? Or do we treat the memory cost differently at centers and servers? For example, memory at server side is cheaper than center side? Do we care more about the total memory cost, or maximum memory cost among all the parties? Or it doesn't matter as long as it doesn't exceed the memory limit (8G per party)?

> We will measure the memory cost on servers; but it will be not be considered as a priority as long as it does not exceed the memory limit.

(e) Do the centers and servers run machines of the same configuration?

> The servers will be run machines of similar configuration (if not identical) in our evaluation. The computation on the centers should be light (i.e., only for computing hash functions and other relatively simple computations if needed).

========== question asked on 04/11/2017 ==========

1. Can we assume that the centers are honest but curious?
Yes, that sounds good.
2. Can we submit multiple solutions of different flavors?
Up to two solutions per team.
3. What will you be measuring?
The performance will be measured according to the following order
#1 Turnaround time
#2 Communication overhead,
#3 Memory cost
a. The speed by which a response is given back to the center on whether its record is a duplicate when a batch of 100 is submitted?
Yes, this will be measured. We will average the performance over several batches.
b. You had stated in your previous letter that the system comes with some initial records already held by the centers (assuming that de-duplication has already been done for those). Do you count the processing time that will be done on the initial state of the system? Or can we assume that this is for free?
We assume the de-duplication on the existing records are done. This part is for free.
4. The batch of 100 records that will be given to the servers will originate from a single center?
Yes, when a single center first accumulated 100 records, it will send to the servers on a first-in-first-serve base
5. Can we assume an off-line computing period for the center?
yes, we can allow some off-line computing. But, the team should make it clear whether the offline work is one-time or needs to be done before running every computing tasks. Also we will measure the performance impact of this step. Latter we can decide whether to ignore or take into account the overheads of this step.
6. Can we assume public parameters for the system, such as a public key?
yes, a public key or some reasonable public parameters should be okay
7. what kinds of networks (latency and throughput) are there between the servers and between the servers and the centers? E.g., 10Gbps ethernet between the servers?
I would imagine these will be connected with 100Mbps network to represent the reality situation.
8. On what machines will the test be running? In particular, are the centers as powerful as the servers?
You can imaging these are typical instances like EC2 M4 large (2 core, 8G memory). We will provide a Ubuntu VM on our website.
https://aws.amazon.com/ec2/instance-types/

========== question asked on 03/29/2017==========

1. For the bounds on the duplication percentage between centers, for the 0.1% you mean that all pairs of centers will have at least 0.1% duplications and some pairs will have up to 10% duplications? This means that the duplication percentage is in the interval [0.1, 10]% between any pair of centers with more concentration around the 0.1%?
+Yes

2. What is the meaning of patients records are entered at a random pace locally? does that mean the computation is done in an online fashion? That is, the servers will not receive the whole input from the centers at the beginning of the computation, instead they will

receive the input in small chinks from the centers (or a single record at a time) over the course of computation? but this mean that the centers do not have the full list of their patients when the computation/protocol starts which conflicts with the problem setup as you will give us a set of patents records from the beginning.
+We assume there are n centers. and each has m_i record for i = 1, 2, .. n during the evaluation. Then, records can be sent to the servers in several batches (e.g., 100 records per batches) at different time. The server is required to be able to handle the secure de-duplication task based on either the existing received records or any future coming batches.

3. What is the exact form of a patient's record? Assume that we have for each patient the following info only: ID, first and last name, and gender. Is the record will be tuple of three hashes for each patient (hash of its ID, hash of first/last name, and hash of gender)? and then it is mentioned that the output is a list of unique IDs of all duplicated records, does that mean we need to output the hashes of patients IDs only (anything else is not required).
+ The input will be ID, first/last name, gender, DOB, and Zip codes, etc. We will release the sample data very soon. Output should be hashes of patients IDs.

4. For the input also, we will not have different patients with same ID, right? I mean there will be no two or more patients from different centers have the same ID but different first/last name for example. So, we should think of the ID as SSN, once two IDs are the same then we know for a fact that this is the same patient? or we need to match all attributes to see if it is the same patient or not?
+ We will not have different patients with the same ID from different centers. We need to match all attributes to see if it is the same patient.

5. For the final output, should the same list of duplicated records among all centers be given to each center? I mean all centers will get the same output even if some of the duplicated records are not part of their original set of inputs/patients?
+ Yes. For a three-center setup, one can assume the intersection is alway occurring between one center (with newly added batch) with the other two centers. That is, it can be assumed, if A is the center with the new batch data, then B \cap C = \emptyset, the report should be A \cap (B \cup C) that should be shared with every center. In that case, A knows some of its records are already in either B or C, and thus should be removed; but it does not know the duplicated records are in B or in C. Similar idea can be extended to the setup with more than 3 centers.

6. What is the meaning of turnaround time with increasing number of records at each center?
+ The turnaround time is defined as the overall computation time to securely de-duplicate all the records from all centers. We will use different number of records at each center to test the scalability of each solution, where the number of records can vary from 10K to 100K.

========== question asked on 03/28/2017==========

1. Do the records come one by one, or do we get a bunch of records at the same time?
+We assume there are n centers. and each has m_i record for i = 1, 2, .. n during the evaluation. Then, records can be sent to the servers in several batches (e.g., 100 records per batches) at different time. The sever is required to be able to handle the secure de-duplication task based on either the existing received records or any future coming batches.

2. Should we assume that there are three servers that talk to all the centers? Should the solutions scale to more servers? Should we assume a cap say of 11?
+all centers talk to all servers with up to 5 severs.

3. Do we have memory limit assumptions on the side of the servers?
+up to 16 GB

4. What are the communication channels between the three servers? And between the centers?
+We can assume the SSL communication channel between servers and centers.

Task 2:

We have a question on the cryptographic standard of the desired solution. As demonstrated in the problem statement, the desired solution requires 'at least 128-bit security level'. Our team members have questions on this. First, how do you define 'at least 128-bit security level'?
Here we have two different understandings on it.
1. If the desired solution is a 'localized' solution (enclaves run on local machine which stores the data), SGX's encryption mechanism guarantees it.
2. If the desired solution is a 'remote' solution, which means that the machine storing raw data is different from the machine runs the enclaves, does it means that the data transmission channel between these machines need to be encrypted using a key with at least 128-bit in length?

Response: We are talking about a secure outsourcing scenario, you need to ensure 128 bit security for transmission, storage and computation. For example, you need to include MAC and IV in AES to protect the authenticity and integrity of the data. We also need to enable data seal for secure storage on the remote cloud. All above steps require at least 128-bit level of security.

Another question is about the definition about security compliance. Before defining the security compliance, we need to define the adversary model. In security research, the attacker can launch varieties of attacks to a system and researchers always develop mitigation to one attack in a single project. Here are some common adversary models and we are curious about the adversary model in this task. Does the attacker has the ability to
1) run arbitrary code anywhere other than SGX enclave?
2) get access to the local/remote input data files?
3) has enough control of the server and launches side-channel attack to guess the data?
4) get the private signing key of Intel SGX service and forge a 'virtualized' enclave?
5) attack the foundation of Intel SGX (Intel SGX is based on Intel ME, which is a blackbox to user and developer but recently proved to be exploitable from remote.)?

Response: You do not need to consider advanced attack models like controlled side-channel, Hyper thread attack and cache timing attack. This task is asking for a "naked" SGX implementation with malicious model. Ideally, data should reside in SGX but due to the memory limitation, you will have to swap data in and out using data sealing or/and software paging in linux. You need to ensure the security of data outside enclave during this process. We need to ensure the authenticity, integrity and freshness of the data.

1) The VCF file format allows several degrees of liberty. Can we assume that all VCF files will be exactly in the same format as in the example dataset? In particular, that the VCF files will not contain more than one SNP per line, or additional fields.

Response: I am not sure what you meant "exactly the same". But we can ensure there will be no more than one SNP per line, and no additional fields.

2) Do we consider that the application will run on one hardware node (e.g., one CPU with 4 hardware cores and hyperthreading), or is it planned to run the application in a distributed multi-node environment (e.g., SGX clouds)? Different scenarios can lead to different decisions.

Response: We plan to run the applications in one hardware node. Precisely, we will use Intel® Xeon® E3-1280 v5 CPU, 1TB SSD, 64GB memory, 4 cores.

3) How much computation is the local machine allowed to do before sending the data to a remote environment? If the answer is none, then the local machine would simply send the encrypted VCFs to the remote machine as is. Otherwise, the remote machine could, for example, compress the VCF file and encrypt it, then send it.

Response: You can preprocess individual VCF files (such as indexing and compression), but not with any combination of multiple VCF files. The pre-processing time will be evaluated separately as the actual computation on SGX.

1. What operating system will be used by the evaluation machine?

ubuntu 16.04

2. What SGX SDK version is available in the evaluation machine?

SDK Version 1.7

3. Should we consider all the attributes of the given VCF file sensitive (POS, ID, REF, ALT, QUAL, FILTER, TYPE) ? I mean, all the attributes should be encrypted when the records reside in the untrusted disk?

You can preprocess individual VCF files (such as indexing and compression), but not with any combination of multiple VCF files. The pre-processing time will be evaluated separately as the actual computation on SGX. All attributes need to be encrypted.

Task 3:

The following parameters can be assumed as public information.

N (Number of record)
K (Number of features)
p (ratio of positive / negative classes)