Privacy-Enhancing Cryptography to Complement Differential Privacy

Luís T. A. N. Brandão and René Peralta*

†November 03, 2021

Abstract: In this post, we illustrate how various techniques from privacy-enhancing cryptography, coupled with differential privacy protection, can be used to protect data privacy while enabling data utility. Of notable interest is the setting where there are multiple sources of relevant data, each having privacy constraints about data sharing. Privacy-enhancing cryptography is naturally suited to resolve challenges in multi-party and interactive scenarios, avoiding the sharing of data across parties. Its combined use with differential privacy broadens the set of problems that can be handled in a privacy protecting manner. In this post, we consider a use case related to private medical data, but the ideas can easily transfer to myriad other settings.

Protecting data privacy across multiple datasets

Alice is the data steward at hospital H_A, responsible for a database of patient medical records. Similarly, **B**ob is the data steward at another hospital H_B. Alice and Bob learn of **R**ya's ongoing **r**esearch about correlations between patients' age and diagnosis. Alice and Bob would like to help Rya, since the research could provide useful insights to derive better medical practices. However, there are privacy restrictions that prevent Alice and Bob from sharing their databases.

Suppose Rya is interested in learning the number of patients diagnosed with a condition X, by age range. Differential privacy allows Rya to obtain an approximate result from each of the two hospitals, each tweaked by a noise addition in order to protect privacy. However, the restriction to two separate results has shortcomings: (1) when combining the two results, Rya is unable to make corrections that may be required because of possible duplicate entries; (2) the pair of individual replies leaks information about hospital differences, which is unrelated to Rya's goal.

In contrast to the above scenario, **privacy-enhancing cryptography** (PEC) enables Rya to interact with Alice and Bob to obtain a combined response that is corrected with respect to duplicate entries. This is done without Alice and Bob sharing data between themselves, and without Rya learning anything beyond the intended output. PEC can be combined with differential privacy techniques to provide the best tradeoff between privacy and accuracy. Table 1 illustrates how errors may be introduced when not using PEC. The error is in the sum of two correlated counts, which overestimates the true count in the union of the two sets. These errors (with rates above 25 % in the example) can significantly hinder the utility of the results.

[*] Brandão (orcid:0000-0002-4501-089X) and Peralta (orcid:0000-0002-2318-7563) are with the PEC project, Cryptographic Technology Group, at the National Institute of Standards and Technology (NIST). Opinions are by the authors, not to be construed as official NIST views. Brandão is at NIST as a Foreign Guest Researcher (contractor from Strativia).

[†] This is the authors' version of the invited post submitted on 2021-Sep-15 to the NIST Differential Privacy Blog Series, published online on 2021-Nov-03 as post #11. Any mention of commercial products or reference to commercial organizations is for information only; it does not imply recommendation or endorsement by NIST, nor does it imply that the products/organizations mentioned are necessarily the best available for the purpose.

Table 1: Measure (N) of the number of patients with diagnosis X

Age range	Without differential privacy					With differential privacy				
	Exact counts			Error rate r		Differentially private counts			Error rate r	
	A	B	$\cup_{AB}$	If N=A+B (no PEC)	If N=$\cup_{AB}$ (with PEC)	A'	B'	$\cup'_{AB}$	If N=A'+B' (no PEC)	If N=$\cup'_{AB}$ (with PEC)
0–30	8	11	15	26.7 %	0 %	9.2	10.1	16.5	28.7 %	10 %
31–60	123	85	172	20.9 %	0 %	119.5	87.2	168.3	20.2 %	2.2 %
61–120	428	632	660	60.6 %	0 %	433.7	633.1	656.8	61.6 %	0.5 %

Legend: A (counts at hospital H_A); B (counts at hospital H_B); $\cup_{AB}$ (counts at the union of hospitals H_A and H_B); A', B', $\cup'_{AB}$ (differentially private versions of A, B, $\cup_{AB}$); $r = N / \cup_{AB} - 1$.

What is safe to compute? As discussed in previous posts, differential privacy techniques add noise to the exact result of a query, to limit privacy loss while still enabling a useful answer to relevant queries to a database. From a privacy perspective, the approximate result is "safer" than an accurate answer. PEC techniques achieve something different: they circumscribe the disclosure to only the specified final output, even if the source inputs are distributed across various parties. Such disclosure is "safer" (with respect to privacy and accuracy) than replies that would include isolated answers from each separate source. This is achieved with cryptographic techniques that emulate an interaction that would be mediated by a (non-existing) trusted third party. The PEC and the differential privacy paradigms can be composed to enable better privacy protection, namely in scenarios where sensitive data should remain confidential in each individual original source. Differential privacy adjusts the query result into a noisy approximation of the accurate answer, which PEC can compute without exfiltrating additional information to any party.

Privacy-enhancing cryptography techniques

The next paragraphs consider five privacy-enhancing cryptography techniques: **s**ecure **m**ultiparty **c**omputation (SMPC), **p**rivate **s**et **i**ntersection (PSI), **p**rivate **i**nformation **r**etrieval (PIR), **z**ero-**k**nowledge **p**roofs (ZKP), and **f**ully **h**omomorphic **e**ncryption (FHE). We illustrate how they can apply, in composition with differential privacy, to Rya's research setting. The examples include settings that have to handle more than one database, account for privacy restrictions from Rya, and ensure correctness even if some parties misbehave.

SMPC. With **s**ecure **m**ulti**p**arty **c**omputation (SMPC) [1, 2], Rya can learn a statistic computed over the combined databases of Alice and Bob, without actually combining the databases. Alice and Bob do not see each other's data, and Rya learns nothing about the databases, i.e., other than what can be inferred from the (differentially private) obtained statistic (see Figure 1).

The application of SMPC together with **d**ifferential **p**rivacy (DP), as showcased in Figure 1, constitutes a more secure alternative to "**central DP**" and to "**local DP**", where a curator combines the data but also becomes subject to privacy breaches [3], as explained in a previous post. Central DP compromises on security, by requiring a potentially hackable curator to serve as custodian of the data from multiple hospitals, to be able to answer queries in a DP manner. Local DP makes a tradeoff between privacy and accuracy, mitigating the foreseeable case of the hacking of the curator, by requiring that the data sent from each hospital to the curator has been DP-protected. SMPC (of differentially private statistics) enables the best of both worlds: it provides the best-possible

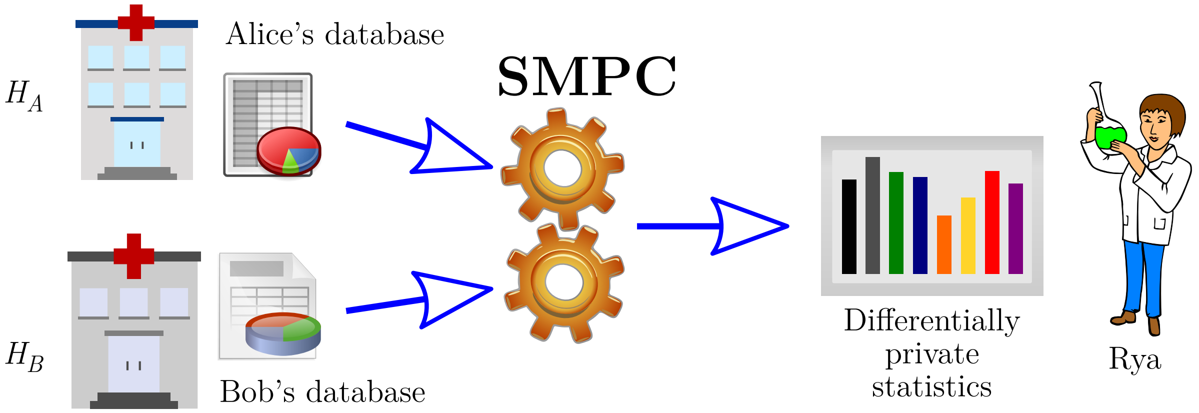

Figure 1: Secure computation of differentially private statistics from combined databases

accuracy (as in central DP), and it avoids the leakage potential from the possible breach of a curator (in both central and local DP).

A different application of SMPC, to avoid requiring online availability of the original sources (the hospitals), is to use a distributed curator (see Figure 2). Here, the data from each hospital is *secret shared*, so that no sole component of the curator knows the data, but a threshold number of them is sufficient to answer any query, i.e., using SMPC to compute over the secret-shared data. Presently, the MPC Alliance is a consortium that joins more than 40 companies interested and actively engaged in developing and implementing SMPC solutions.

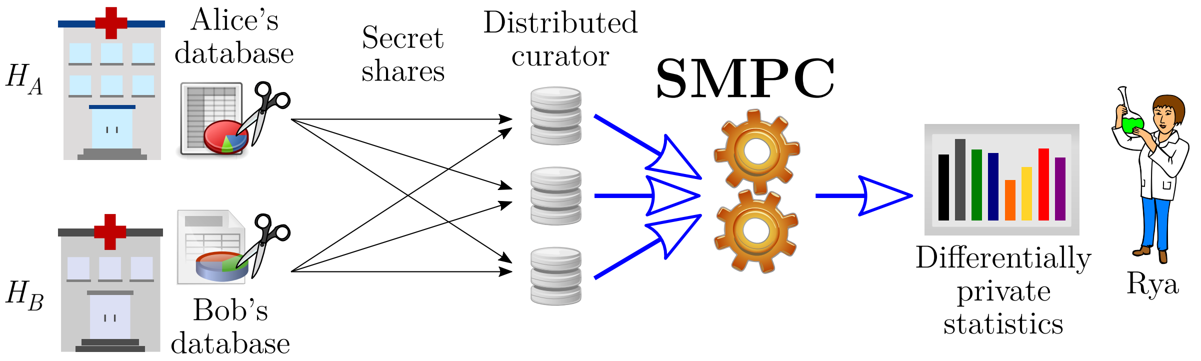

Figure 2: Differential privacy via a distributed curator, using SMPC over secret-shared data

PSI. With **p**rivate **s**et **i**ntersection (PSI) [4, 5], Alice and Bob can determine the set of patients that are common across their databases, without sharing any information about other patients. Naturally, this intersection can be considered sensitive information that should remain private. A variant called PSI cardinality can be used to compute statistics about the common patients, such as how many there are, without divulging the set itself (see Figure 3). The mentioned PSI# can also be considered per age and per diagnosis.

We note that even the cardinality of the intersection may be sensitive information. Therefore, this multi-party statistic can itself be subject to differential privacy protection. From a different angle, the statistic can also be useful for the hospitals to determine how to parametrize their differential privacy protection level for subsequent queries from external researchers. This may improve privacy

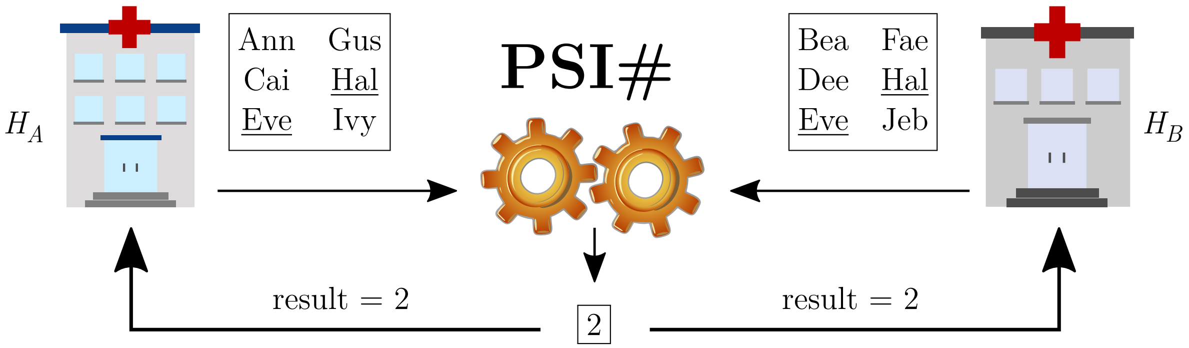

Figure 3: Private set intersection cardinality (PSI#) of patients across two hospitals

and/or accuracy in settings where Rya will later be separately querying both hospitals. Conceivably, this could be useful in a setting related to the COVID-19 pandemic, as considered in an application where a party learns a risk of infection based on the number of encountered persons that have been diagnosed as infected [6].

PIR. With **p**rivate **i**nformation **r**etrieval (PIR) [7, 8], Rya is allowed to learn the result of a query sent to Alice's database, without Alice learning what was queried (see Figure 4). Recalling our example from Table 1, Rya can learn the differentially private approximation ($A' = 119.5$) of the number ($A = 123$) of patients in H_A, with diagnosis X and within the age range 31–60, without Alice learning the queried age-range. Naturally, this can be generalized to hide which diagnosis (X) was queried.

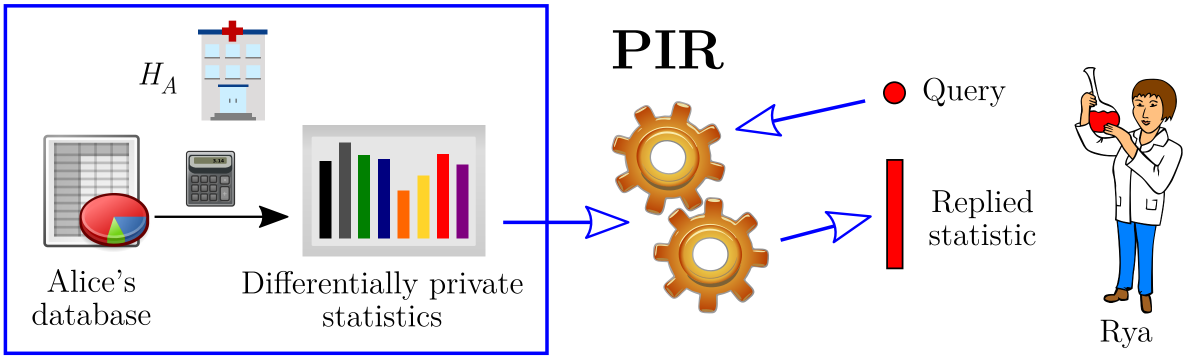

Figure 4: Private information retrieval (PIR) of a differentially private statistic

ZKP. Zero-**k**nowledge **p**roofs (ZKPs) [9] allow making proofs about data that has somehow been "committed" (for example, by disclosing an encrypted version of a database), without revealing the actual data. Thus, once the data has been committed, a database owner can prove that the reply given to a certain query correctly relates to data that has not changed. This is a great tool for allowing accountability while protecing. In particular, ZKPs can also be used to enable other PEC techniques (such as SMPC, PSI and PIR) in the so-called malicious model, where any of the parties (Alice and Bob) could otherwise undetectably deviate from the agreed protocol. For example, a ZKP can be used by Alice to prove to Rya that an answer satisfies an appropriate differential privacy protection, i.e., resulting from a correct noise addition, with respect to the original secret database (see Figure 5). Presently, zkproof.org is an open initiative that seeks to mainstream the development of interoperable, secure, and practical ZKP technology.

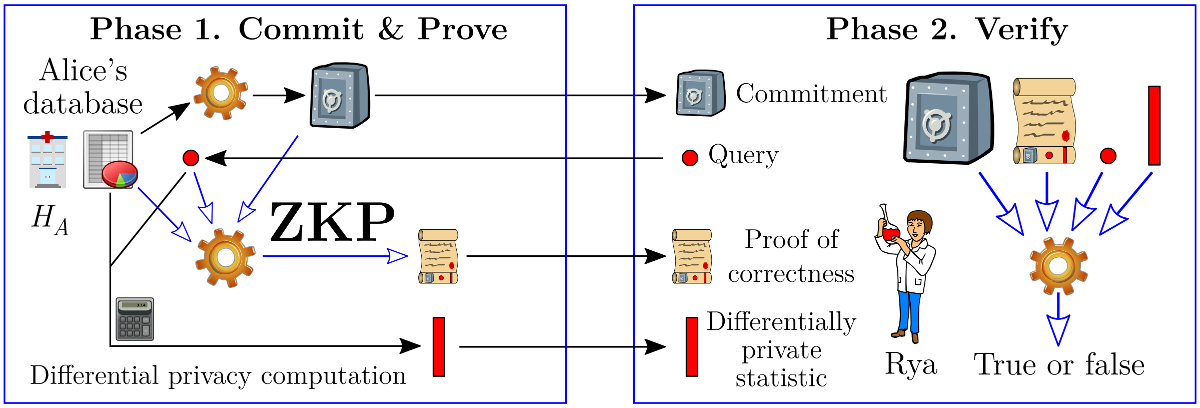

Figure 5: Differentially private answer, with a **z**ero-**k**nowledge **p**roof (ZKP) of correctness

FHE. **F**ully-**h**omomorphic **e**ncryption (FHE) [10] allows computing over encrypted data, without knowing the secret key. In other words, someone without the secret key (needed to decrypt) is able to transform a ciphertext (i.e., the encryption of a plaintext) into a new ciphertext that encrypts an intended transformation of the plaintext. Conceptually, Rya can encrypt the intended query, send it to one or various hospitals, and then let the hospitals transform the encrypted query into an encrypted DP-protected reply, which Rya can later decrypt (see Figure 6). The computation can be made sequentially across hospitals, with each new transformation remaining encrypted until the final stage of decryption by Rya.

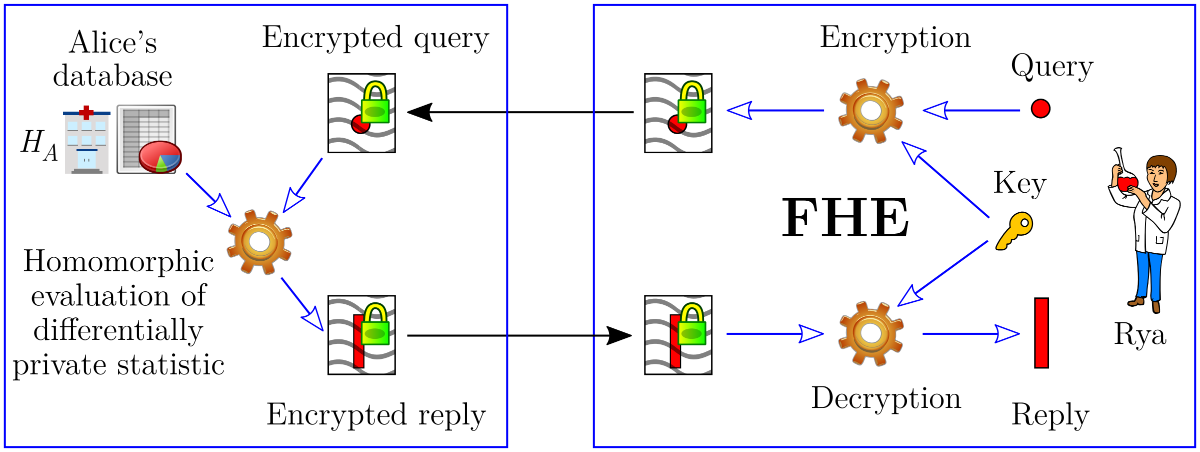

Figure 6: Using FHE for private computation of differentially private statistics

FHE is a natural primitive to enable privacy-preserving delegation of computation. The reader may note that the FHE use-case in Figure 6 is very similar to the PIR use-case in Figure 4. Indeed, FHE can be used as a primitive for a number of other privacy-enhancing cryptography tools, including PIR, PSI, and SMPC. A typical implementation benefit is that it enables solutions with minimal communication complexity. A possible downside of FHE is that it can be computationally more expensive than other solutions. However, there are applications for which FHE is practical, and the field is rapidly improving. The homomorphicencryption.org initiative is promoting the standardization of FHE.

Conclusion

The roles of privacy-enhancing cryptography (PEC) and differential privacy are significantly different, but they are complementary. Both types of techniques are applicable to protect privacy while enabling the computation of useful statistics. In conclusion, the toolkit of the privacy-and-security practitioners should include PEC tools. They provide additional utility while protecting privacy, including in interactive and multi-party settings that are not amenable to being handled by standalone differential privacy. For more PEC details and examples, follow the NIST-PEC project. This is an exciting area of development.

Acknowledgments: The authors thank Katie Boeckl (NIST), David Garais (Galois), Naomi Lefkovitz (NIST), Joe Near (UVM) and Angela Robinson (NIST), for their review comments.

References

[1] A. Yao. *How to Generate and Exchange Secrets.* FOCS 1986. doi:10.1109/SFCS.1986.25

[2] O. Goldreich, S. Micali, A. Wigderson. *How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority.* STOC 1987. doi:10.1145/28395.28420

[3] A. Evfimievski, J. Gehrke, R. Srikant. *Limiting privacy breaches in privacy preserving data mining.* Symposium on Principles of Database Systems. 2003. doi:10.1145/773153.773174

[4] C. Meadows. *A more efficient cryptographic matchmaking protocol for use in the absence of a continuously available third party.* SP 1988. doi:10.1109/SP.1986.10022

[5] G. Garimella, P. Mohassel, M. Rosulek, S. Sadeghian, J. Singh. *Private Set Operations from Oblivious Switching.* PKC 2021. doi:10.1007/978-3-030-75248-4_21

[6] R. Peralta, A. Robinson. *Encounter Metrics and Exposure Notification.* Journal of Research of the National Institute of Standards and Technology. 2021. doi:10.6028/jres.126.003

[7] B. Chor, E. Kushilevitz, O. Goldreich, M. Sudan. *Private Information Retrieval.* FOCS 1995 and JACM 1998. doi:10.1145/293347.293350

[8] E. Shi, W. Aqeel, B. Chandrasekaran, B. Maggs. *Puncturable Pseudorandom Sets and Private Information Retrieval with Near-Optimal Online Bandwidth and Time.* Crypto 2021. ia.cr/2020/1592

[9] S. Goldwasser, S. Micali, and C. Rackoff. *The Knowledge Complexity of Interactive Proof Systems.* STOC 1985 and SIAM-JC 1989 doi:10.1137/0218012

[10] C. Gentry. *Fully Homomorphic Encryption Using Ideal Lattices.* STOC 2009. doi:10.1145/1536414.1536440

Used cliparts: The author-assembled illustrations make use of fourteen cliparts obtained from clker.com (Accessed in June 2021): 🏥 (Hospital = H_A); 🏥 (Hospital = H_B); 📊 (spreadsheet = database for Alice); 📊 (spreadsheet = database for Bob); ⚙ (gears = PEC computation — local if one gear; interactive if two gears); 📊 (chart = differential-private statistics); 🔬 (researcher = Rya); ✂ (scissors = secret sharing); 🛢 (storage disks = components of distributed curator); 🖩 (calculator = non-PEC computation); 🔐 (vault = commitment); 📜 (scroll = proof); 🔑 (key = FHE key); 🔒 (lock = FHE ciphertext). Their utilization here is made under a free-use license expectation.