PROCEEDINGS OF SCIENCE

Responsible Analytics

Luigi Scorzato[a,*]

[a] *Accenture AG,*
12 Route Francois-Peyrot 1218, Geneva, Switzerland

E-mail: luigi.scorzato@accenture.com

I review some trade-offs between deriving insight from the data and protecting its confidentiality. Then, I review a few techniques that have enabled significant progress in this topic and have matured in recent years.

European network for Particle physics, Lattice field theory and Extreme computing (EuroPLEx2023)
11-15 September 2023
Berlin, Germany

[*] Speaker

1. Introduction

One of the greatest challenges of any data project is how to exploit the insight of the data while protecting the confidentiality of the data. This is not a single problem, for which there is one best solution. It is rather a very complex threat surface for which we must combine multiple solutions and make them work together. Here, I focus on two approaches that have become mature only in recent years and have a huge potential. They are currently not yet mainstream approaches, but there are strong reasons to believe that they will be so within the next few years.

2. Multi-Party Confidential Computing

Consider the following problem as an example (many similar ones can be easily envisaged). The Bank A wants to do Anti-Fraud (AF) detection or Anti Money Laundering (AML) detection. It has access to its own customer data & account data. It has also access to transaction data between itself and bank B. But, of course, it has no access to customer & account data from bank B. This is crucial information for preventing fraud. E.g. Was the account open very recently? That piece of information alone would dramatically improve the AF and AML detection systems. Unfortunately, today, in most cases, we can only accept the limitation, to the delight of criminal organizations.

But it does not need to be so. Today, Multi-Party Clean Rooms (MPCR, also called Multi-Party Confidential Computing) offer a solution. The MPCR technology leverages Confidential Computing (CC), which was originally introduced for a very different reason: facilitating cloud migrations. In fact, one of the main historical concerns hampering cloud migrations has been the risk that the cloud administrators could tamper with the data of the cloud customer. In general, the data stored in the file system are encrypted, they are also encrypted while they are transfered between machines and applications, but a cloud administrator could still dump the volatile memory (RAM) of a virtual machine, or even change it. In practice, this risk has been mainly theoretical, but it has blocked many discussions about cloud migrations.

For this reason, all the main chip manufacturers have developed their solutions to this problem. That was not straightforward, because, historically, OS & CPUs had been designed with the idea that the administrator can do everything. This is not a law of nature, but changing this paradigm did require a very low-level redesign of the architecture. Today, all major chip manufacturers are selling confidential computing solutions [1–5]. And all cloud providers offer them for a moderate price and performance penalty [6–8]. These HW architectures include a so-called trusted execution enclave (TEE) that is isolated from everything else, including its own OS, the hypervisor & OS of the host machine. Thanks to this solution, the data are always encrypted outside the TEE enclave and only privileged code has access to the TEE. Moreover, the solution offers a cryptographic confirmation that the enclave is correctly configured and runs the software as expected.

These new capabilities are sometime called *encryption in use*, which complements the standard *encryption at rest* and *encryption in transit*. This solution enables not only a more serene migration to the cloud, but an entire new world of opportunities. Because, if we have a formal guarantee that not even the system administrators with physical access to the machine can interfere with our data, Then we can invite also third parties to process their own data together with our data on those machines, with the guarantee that nobody can see the data of the others. This is the idea of the Multi-Party Data Clean Rooms.

In the specific case introduced above, both banks send their data to a central machine, running a TEE, with a formal cryptographic guarantee that nobody can access the data of the other. And their data are used only for the agreed analysis and only the agreed output is released. This could be an alert that links to data from Bank A only, but with an additional score that reflects the knowledge of the data hold by bank B.

The potential applications of this technology are countless. To mention another one, consider the problem of sharing patients data between hospitals, researchers and pharma companies for clinical trials. These processes are highly regulated, but the regulation is often ambiguous [9], and the industry is plagued by risky practices. In fact, it is claimed [10], that 70% of clinical trials are at high risk of data breach. Note, also, that almost half of the ;argest data breaches in history happened since 2022 [11], which means that the problem is worsening rapidly and worryingly. The risk is high, because very sensitive data are shared across multiple entities and the techniques used to protect the data are often not adequate. Also in this case, multi-party clean rooms with a TEE would represent a great step forward, at least in some cases.

Another very important application is the opportunity to revolutionize end-to-end supply chain analytics. A typical supply chain involves multiple parties: suppliers, manufacturers, carriers, retailers, distribution network, etc. The data that each party can share with the others is very limited in terms of granularity and richness of details. On the other hand, this topic is so important that many multi-billions merge and acquisitions are largely motivated by the need of a better control over the supply chain.

Also in this case, a central TEE would be able to collect the data from all the parties without sharing them with anyone (not even the entity that owns and/or manages the TEE), analyse the combined data and share the agreed outcome that contains the useful information without revealing confidential information.

Until now, we have discussed the problem of protecting confdential data, but we don't need to protect only data. We often want to protect also the Intellectual property of the software (SW) that processes the data. A simple and important example is the following. Consider two parties: a data owner and a SW owner. They want to do analytics by combining their precious data with their powerful algorithm, but neither of them wants to share their most valuable asset with the other. What does happen today? Either nothing, which is a missed opportunity, or someone accepts a big risk that would very much prefer to avoid.

But it does not need to be so. A multi-party clean room offers the best guarantees that each of those parties wishes, while enabling the analytics that they are aiming for. It is irrelevant who owns or manages the TEE, because the technology ensures that even the TEE administrator cannot access the information that is not agreed by both parties.

Although CC is a very recent technology, its adoption is growing fast. Appropriate legal frameworks are also being defined. There are, however, some technical complexities that need to be overcome. First, the process to onborad the data, agree on the processing and the expected output is rather new. Companies like Opaque [12] and Decentriq [13] support the implementation of these processes. Secondly, use cases that require complex processing and/or model training are more challenging. Moreover, data and metadata quality become more crucial.

The last two points lead us naturally to the introduction of another technique that plays a key role to overcome these challenges: Differential Privacy and Synthetic Data.

3. Differential Privacy

Consider again the case of clinical trials. In some cases, the data processing is well defined, which is a perfect use case for multi-party clean rooms, as we have seen.

But in other cases, some level of data exploration will be necessary. How can we enable the scientists to derive insight from the data without putting at risk the privacy of the patient?

One traditional (and still common) approach to deal with this problem is to simply mask the obvious direct identifiers. But this does not prevent reidentification. In fact it well known that 2-3 non sensitive attributes are often sufficient to reidentify an individual. For example, it was famously noted [14] that the 87% of the US population can be reidentified based only on ZIP-code, gender, date of birth. Many other, much less sensitive attributes by themselves, are also sufficient to re-identify a sizeble amount of the population.

It is important to emphasize that reidentification (even with high probability) is explicitly forbidden by GDPR. Although, the same regulations fails to point to a technique that provably prevents reidentification with high probability [9].

K-anonymity [15] has become widely adopted in the past decade. But, besides dramatically spoiling the value of the data, k-anonymity still allows reidentifying targets with high probability [9], which, again, violates GDPR.

It is often argued that big aggregates are safe. But, we should also consider all possible differences of the many big aggregates previously released. If, for example, a large aggregate is updateed, the difference between the latest update and the previous version might reveal exact personal information about an individual [16].

The past two decades have seen a revolution in the science of data privacy protection. The groundbreaking work [17] introduced the concept of Differential Privacy (DP) and it was followed by an intense activity to design optimised mechanisms that implement DP and the development of the corresponding software tools. See [18] for a recent review. By now, DP is recognised by the scientific community as the gold standard of privacy protection and its implementations have reached considerable maturity. This is testified also by the number of prestgious academic awards granted recently to the inventors of DP (Gödel Prize 2017, Knuth Prize 2020, ACM award 2021, RSA award 2022).

DP was officially adopted by the US Census since 2018 and the 2020 release is fully based on DP [19]. The Swiss Federal Statistical Office is evaluating it and contributing to its development [20]. Apple, Google, Facebook, LinkedIn, Microsoft use it to collect reports from the users, and more [21]. Accenture has also applied it in a very large scale project. Wikimedia has adopted it to generate usage statistics, to protect individual contributors from potential threats [22]. Multiple white papers recommend including DP explicitly it into policies, especially to clarify GDPR [9, 23].

Differential Privacy has a reputation of being a complex technique. But, in fact, the idea is rather simple and I will illusrtate it here briefly.

Differential Privacy offers a *definition* of what we want to protect. To introduce it, let us consider first the simple example in Tab. 1. How can I publish statistical reports without violating

Name	**Age**
Pippo	112
Pluto	60
Paperino	81
Topolino	38

$\Longrightarrow$

Name	**Age**
Pluto	60
Paperino	81
Topolino	38

Table 1: We do not violate Pippo's privacy if we drop Pippo's record from the dataset.

the privacy of the individuals that contribute to it? For examples, outliers are at great risk, like *Pippo* in this example.

The main *assumption* adopted by DP is that we will not violate Pippo's privacy if we drop Pippo's records from the dataset. Similarly, we do not violate Pippo's privacy if we add noise to the output of each queries in a way that the probability of each outcome is almost the same whether we keep Pippo in the dataset or remove him. Because then, nobody could infer with certainty whether Pippo was actually in the dataset or not and nobody could infer even with significantly increased probability whether Pippo was in the dataset or not.

Formally, the definition is as follows [17]. A randomized query Q is ε**-Differentially Private** iff:

- $\forall$ datasets D and D' that differs only by one record,
- $\forall$ outputs r in the range of Q,

$$e^{-\varepsilon} \leq \frac{Pr[Q(D) = r]}{Pr[Q(D') = r]} \leq e^{\varepsilon}, \tag{1}$$

where the probability $Pr[]$ is evaluated overe the random noise added to the queries output. In other words, the probability of any outcome r is roughly the same (i.e. within a band $[e^{-\varepsilon}, e^{\varepsilon}]$) whether any individual is included in the dataset or not.

Eq.1 is a *definition*. Importantly, over the years, multiple solutions have been developed *mechanisms* that implement this definition with ever increasing signal to noise ratio. As a result, today we have a rich ecosystem of tools (largely open-source) that make the implementation of DP practical and effective [18].

DP introduces the important concept of *Privacy Budget* ε, which measures the maximum potential information gain that an attacker could obtain about the original data from the knowledge of the published report. It is important to understand that multiple queries to the real data consume more and more privacy budget. For the most common mechanisms, the total privacy budget is the sum of the budgets of the individual queries. For total privacy budget $\varepsilon \lesssim 1$, there cannot be any significant information gain. For any $\varepsilon < \infty$ it remains impossible to draw any conclusion with certainty. However, for $\varepsilon \gg 1$, it might be possible to derive conclusions with high probability. Because DP offers a *worst case* guarantee, also quite large value of ε might be still safe in practice, but the formal guarantees of DP are not useful anymore.

4. Synthetic Data

The examples above addressed the problem of publishing reports or histograms safely. But, once histograms are available, they can be used to generate synthetic data with the same statistical distributions. Synthetic Data are fake data that reproduce some of the statistical properties & format of the real data. There are many ways to generate synthetic data. Gen AI, for example, has expanded considerably the types of data that could be synthetized. However, this does not guarantee by itself privacy protection[1]. But, if we combine any synthetic data generation with DP, the resulting synthetic data inherit the privacy guarantees of DP. Moreover, the *post-processing guarantee* [17], impled by DP, ensures that the synthetic data produced in this way can be used as much as wished, without additional risks (i.e. without consuming further privacy budget).

If one knows exactly the query that is required on the data, then it is usually more efficient to add DP directly in the query, without going through synthetic data. But if we don't, synthetic data are a very flexible way to answer a much broader range of unforeseen queries.

It is important to stress the difference between DP and the multiple alternative techniques that are still being used to generate synthetic data from real data, many of which claim to also protect the privacy of the original data.

It is instructive to compare the history of Data Privacy to the history of Cryptography. The history of cryptography is filled with examples of astute encryption methods that look hard and unbreakable when introduced, but could be broken eventually, with dire consequences for those who trusted them. The turning point that led to modern cryptography was the realization that we can & must prove mathematically that breaking a cryptographic mechanism is sufficiently hard, according to a suitable complexity theory. Today, nobody would take seriously any encryption method that does not come with a such proof. DP represents the same turning point for privacy protection, but its adoption is still inconsistent across vendors and organizations.

DP has become mature in very recent years, but there are still some technical obstacles that delay its wider adoption. First, as we already mentioned, each time we access the same data we cosume some additional privacy budget. However, this is not a critical problem, because DP is mainly needed in the development and model training phases. Simple inference, in production environment, consume much less privacy budget and can also be done safely by using confidential computing, as described in the previous section.

Another problem is the need of clean, well formatted, well documented data and metadata. This topic is discussed in more detail in the next section.

5. Analytic workflows and standard

An analytic workflow requires multiple tasks (and roles) with different challenges, from the point of view of data confidentiality. It is convenient to group these tasks in two main broad categories. On one hand, data analysis, training of machine learning models, report generation require data with realistic statistical properties. The real data must be accessed multiple times by teams that are usually rather small. In these phases, the format of the data has already been

[1] Neural Networks definitely memorize training data and occasionally leak them [24]. This holds also when Federated Leraning is used [25].

processed and homogenized. This category of tasks has been the main focus of research in synthetic data and it is a very mature topic.

On the other hand, other tasks like data ingestion, data mapping, application development, data quality assessment and remadiation, have very different challenges. They demand much lower requirements from the point of view of the statistical properties of the data. Moreover, the real data need to be accessed much more rarely by humans. However they are performed by larger teams that take care of a large number of different datasets. This category of tasks seem more amenable to full automation, which would also eliminate the concerns about data confidentiality. The wish of fullly automated data quality checks is very old, but it has proved much more difficult than expected. To be fair, it is quite easy to develop tools that correct *something*. It is much more challenging to do it systematically with a good coverage of all possible issues. The reason is that there are many possible mistakes, and any data quality detection tool must make strong assumptions of what can go wrong and what cannot go wrong. Reducing too much these assumptions lead very quickly to an explosion of complexity and costs.

Generative AI and Large Language Models (LLMs) do help some of these tasks, but they cannot represent the main solution to this problem, because they are not assumptions-free. On the contrary, they involve their own hidden assumptions, which we are not even aware of [26].

The solution can only come from reliable, widely adopted, standards about data formats and metadata structures. Much progress has been achieved in this respect in recent years. A nice example is the convergence to a widely supported standard for data lineage [27]. The growing popularity of data streaming solution is also helping the convergenge to robust standard, simply because it removes the dependence on legacy formats like .csv and similar loosely standardised formats. A lot, however, remains to be done. If we are serious about preventing data briches, we must converge towards reliable standards about data exchanges.

References

[1] AMD, "Strengthening VM isolation with integrity protection and more." https://www.amd.com/content/dam/amd/en/documents/epyc-business-docs/solution-briefs/amd-secure-encrypted-virtualization-solution-brief.pdf, 2020.

[2] Intel, "Intel Trust Domain Extensions." https://www.intel.com/content/dam/develop/external/us/en/documents/tdx-whitepaper-final9-17.pdf, 2022.

[3] G. Dhanuskodi, S. Guha, V. Krishnan, A. Manjunatha, M. O'Connor, R. Nertney et al., *Creating the First Confidential GPUs: The team at NVIDIA brings confidentiality and integrity to user code and data for accelerated computing.*, *Queue* **21** (2023) 68.

[4] X. Li, X. Li, C. Dall, R. Gu, J. Nieh, Y. Sait et al., *Design and verification of the arm confidential compute architecture*, in *16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22)*, pp. 465–484, 2022.

[5] J. Moreira, J.H. Tseng, M. Kumar, E. Ekanadham, J. Jann, P.C. Pattnaik et al., *FlexSEE: a Flexible Secure Execution Environment for protecting data-in-use*, in *Proceedings of the*

20th ACM International Conference on Computing Frontiers, CF '23, (New York, NY, USA), p. 329–336, Association for Computing Machinery, 2023, DOI.

[6] Azure, "Confidential Compute." https://azure.microsoft.com/en-us/solutions/confidential-compute, 2019.

[7] GCP, "Confidential Computing." https://cloud.google.com/security/products/confidential-computing, 2021.

[8] AWS, "Confidential Computing." https://aws.amazon.com/confidential-computing, 2020.

[9] A. Cohen and K. Nissim, *Towards formalizing the GDPR's notion of singling out*, *Proceedings of the National Academy of Sciences* **117** (2020) 8344 [https://www.pnas.org/doi/pdf/10.1073/pnas.1914598117].

[10] Bioboston-Consulting, "70% of clinical trials face data breaches.." https://linkedin.com/pulse/70-clinical-trials-face-data-breaches-why-your-research-ifrze, 2024.

[11] E. Kost, "14 Biggest Healthcare Data Breaches." https://upguard.com/blog/biggest-data-breaches-in-healthcare, 2024.

[12] Opaque-Systems, "Confidential AI platform for trusted AI." https://www.opaque.co, 2024.

[13] Decentriq, "Data clean room provider." https://www.decentriq.com, 2024.

[14] L. Sweeney, "Simple Demographics Often Identify People Uniquely." https://dataprivacylab.org/projects/identifiability/paper1.pdf, 2000.

[15] L. Sweeney, *k-anonymity: A model for protecting privacy*, *International journal of uncertainty, fuzziness and knowledge-based systems* **10** (2002) 557.

[16] A. Wood, M. Altman, A. Bembenek, M. Bun, M. Gaboardi, J. Honaker et al., *Differential privacy: A primer for a non-technical audience*, *Vand. J. Ent. & Tech. L.* **21** (2018) 209.

[17] C. Dwork, F. McSherry, K. Nissim and A. Smith, *Calibrating noise to sensitivity in private data analysis*, in *Theory of Cryptography*, S. Halevi and T. Rabin, eds., (Berlin, Heidelberg), pp. 265–284, Springer Berlin Heidelberg, 2006.

[18] R. Cummings, D. Desfontaines, D. Evans, R. Geambasu, Y. Huang, M. Jagielski et al., *Advancing differential privacy: Where we are now and future directions for real-world deployment*, *Harvard data science review* **6** (2024) 1.

[19] J.M. Abowd, R. Ashmead, R. Cumings-Menon, S. Garfinkel, M. Heineck, C. Heiss et al., "The 2020 Census Disclosure Avoidance System TopDown Algorithm." https://ideas.repec.org/p/arx/papers/2204.08986.html, Apr., 2022.

[20] Federal-Statistical-Office, “The Swiss FSO innovates with OpenDP to protect citizen privacy.” https://www.bfs.admin.ch/bfs/en/home/dscc/blog/2024-02-opendp.html, 2024.

[21] D. Desfontain, “A list of real-world uses of differential privacy.” https://desfontain.es/blog/real-world-differential-privacy.html, 2024.

[22] T. Adeleye, S. Berghel, D. Desfontaines, M. Hay, I. Johnson, C. Lemoisson et al., “Publishing Wikipedia usage data with strong privacy guarantees.” https://arxiv.org/abs/2308.16298, 2023.

[23] R. Cummings and D. Desai, *The role of differential privacy in GDPR compliance*, in *FAT’18: Proceedings of the Conference on Fairness, Accountability, and Transparency*, vol. 20, 2018.

[24] V. Feldman and C. Zhang, *What neural networks memorize and why: Discovering the long tail via influence estimation*, *Advances in Neural Information Processing Systems* **33** (2020) 2881.

[25] O.D. Thakkar, S. Ramaswamy, R. Mathews and F. Beaufays, *Understanding unintended memorization in language models under federated learning*, in *Proceedings of the Third Workshop on Privacy in Natural Language Processing*, pp. 1–10, 2021.

[26] L. Scorzato, *Reliability and Interpretability in Science and Deep Learning*, *Minds and Machines* **34** (2024) 27.

[27] OpenLineage, “An Open Framework For Data Lineage Collection And Analysis.” https://openlineage.io, 2023.