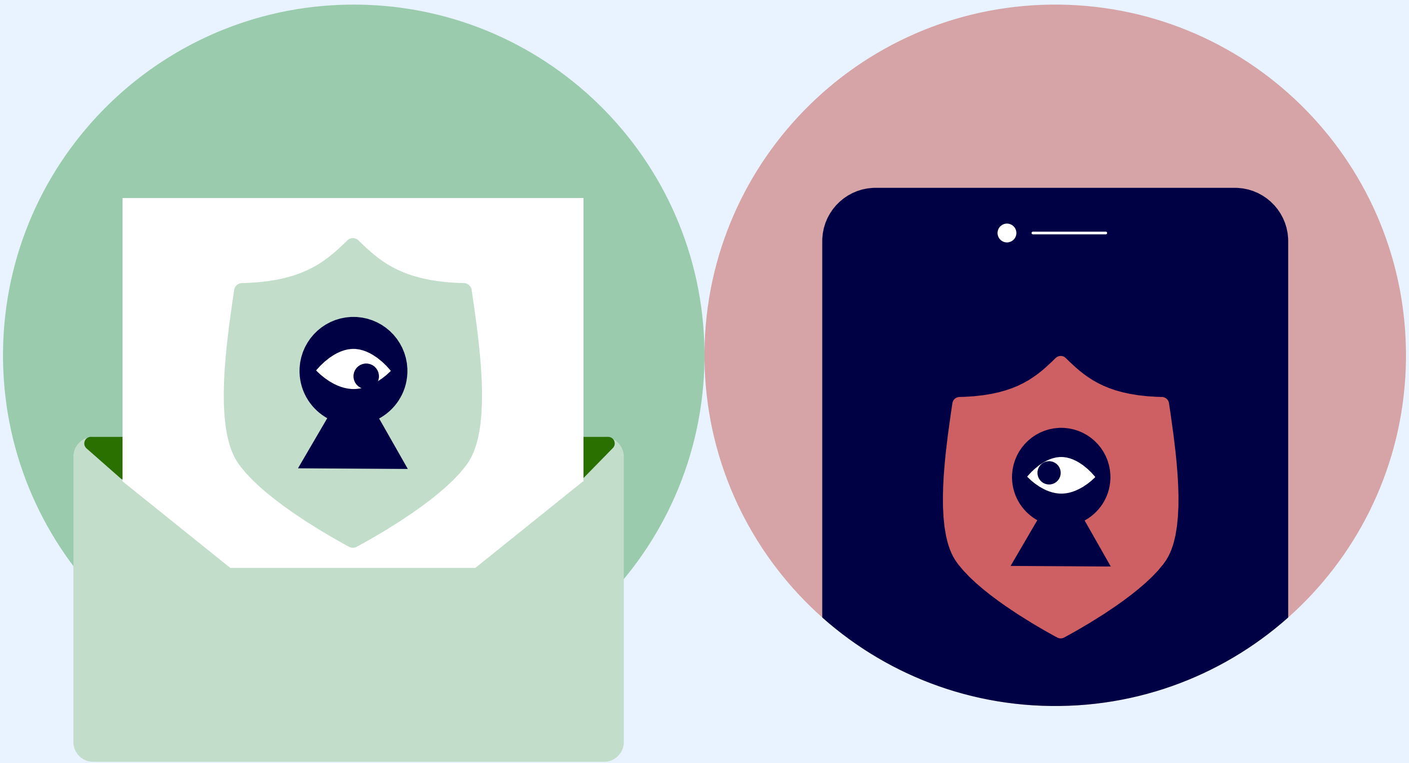
Brasil Uruguay

Open Loop | Meta

Technical Playbook for Advancing Data Protection Principles Using PETs



About Open Loop

Meta's Open Loop is a global program that connects policymakers and technology companies to help develop effective and evidence-based policies for AI and other emerging technologies.

Through a structured methodology, Open Loop participants co-create policy "prototypes" and test new or existing AI policies, regulations, laws, or voluntary frameworks. These multi-stakeholder efforts support rulemaking processes and improve the quality of guidance and regulations on emerging technologies, ensuring that they are understandable, effective and feasible in practice.

About the Open Loop program on PETs:

In September 2022, Open Loop launched a new policy prototyping programme intended to guide and enable companies in Brazil and Uruguay to leverage and apply Privacy Enhancing Technologies (PETs). The program is aimed at incentivising companies to develop and adopt PETs while gathering participants' experience in implementing this technical playbook prototype, and testing its "clarity, effectiveness and actionability".

The policy prototype program engages start-ups providing B2C & B2B products and services in Brazil and Uruguay, across different sectors.

In Brazil, this program is led by a consortium including the Meta Open Loop team and a multidisciplinary research team within the Instituto Liberdade Digital of Brazil, in collaboration with the Brazilian Data Protection Authority – ANPD and the Executive Secretariat of the National AI Strategy – EBIA - Aixis I, ABES AI WG (at the Ministry of ICTs) as observers, and with the support of the WEF's Center for the Fourth Industrial Revolution in Brazil – C4IR.

In Uruguay, the program is led by a consortium including the Meta Open Loop team and the Eon Resilience Lab of C Minds, in collaboration with Uruguay's e-government agency ("Agencia de Gobierno Electrónico y Sociedad de la Información y del Conocimiento", AGESIC) and Uruguay's privacy enforcement authority ("Unidad Reguladora y de Control de Datos Personales").

Disclaimer

The information and materials presented in this technical playbook prototype should only be used in the context of this specific Open Loop policy prototyping program on Privacy Enhancing Technologies (PETs) and, going beyond such program, should not be acted upon without specific legal advice based on particular circumstances, including the laws and regulations of a jurisdiction where a participating company is sitting.

> **The material provided in this document is a prototype of a playbook on Privacy Enhancing Technologies (PETs) and is produced ad hoc for testing purposes only. It does not represent the official position of any of the participants and should not be used for any other purpose.**

This playbook provides an introduction to various Privacy Enhancing Technologies (PETs) and their potential applications in mitigating privacy risks. However, it is important to note that PETs are dynamic and may change or become outdated over time. In addition, their effectiveness in mitigating privacy risks may vary based on the specific scenario and risk profile of an organization.

Most of the PETs presented in this document are not yet mature enough to be deployed broadly, and are not recommended for their immediate implementation. Therefore, while this playbook serves as a roadmap to help organizations understand the options for technologies that could be developed and deployed in the future, it should not be relied upon as a comprehensive or definitive solution for all privacy risks. Rather, it should be considered as a starting point for organizations looking to explore the use of PETs in their privacy risk management strategies.

This playbook is provided for informational and educational purposes only and should not be construed as legal, technical, or other professional advice. It does not reflect the practices or views of Meta or any other member or participant of the Open Loop consortium. The participants do not make any representations or warranties, express or implied, about the completeness, accuracy, reliability, suitability, or availability with respect to the information contained in this material. Any reliance you place on such information is therefore strictly at your own risk.

As with other technologies, the PETs represented here represent technologies that may change or become outdated and are applicable to certain, limited factual scenarios. Therefore, they may not provide a solution for all (or any) identified privacy risks. In addition, we understand that most of these PETs are not yet mature enough to be deployed at scale in most organizations, including Meta or others represented in this Open Loop policy prototyping program. However, we hope that this will be a roadmap that will help organizations understand options for technologies that could be developed and deployed. While the PETs presented here are not yet mature enough to be deployed broadly, we are optimistic that continued development and refinement will lead to their eventual widespread adoption.

In no event will the participants be liable for any loss or damage including, without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising from loss of data or profits arising out of, or in connection with, the use of this material.

1 Introduction

As technology continues to advance, the issue of privacy has become more pressing than ever. With the increasing concern over the misuse and exploitation of personal information, it's crucial that data controllers and developers prioritize privacy in the design and development of their products and services. The goal of this playbook is to assist data controllers and developers in improving the privacy of their data processing activities 'by design' while educating stakeholders on Privacy Enhancing Technologies (PETs), by providing a 3-Step Process for Advancing Data Protection Principles using PETs.

Privacy by design (or data protection by design, PbD) principles are a legal requirement and/or expectation under many jurisdictions[1]. Data controllers are generally required to take technical and organizational measures to ensure that data protection requirements are integrated in the design of their products and services. How to implement 'privacy by design' is rarely a simple or linear process. This playbook is intended to be leveraged by both data controllers and developers or product managers within an organization. There are risks and strategies relevant to the entire organization and risks and strategies relevant to programmers or product managers who will be engaging with privacy issues closer to their domains of work, which we provide in the technical appendices.

Over the past years many PETs have been developed to help preserve and enhance privacy and data protection in information systems. The list of PETs has grown over the years and now includes a wide range of techniques, including tokenization, k-anonymization, global and local differential privacy, federated learning, homomorphic encryption, synthetic data, secure multiparty computation, and trusted execution environments.

Some advantages of using appropriate PET techniques are:

- PETs can help to demonstrate privacy by design and by default.
- PETs can help to comply with the data minimization principle.
- PETs can provide an appropriate level of security.
- PETs can help with anonymization or pseudonymization.
- PETs can minimize the risk of personal data breaches by rendering the personal data unintelligible to anyone not authorized to access it.

Further, under some legal regimes, principles of data protection shall not apply to data rendered anonymous in such a way that the data subject is no longer identifiable. PETs can help with achieving anonymisation. Implementing PETs by a controller does not preclude it from taking other steps to ensure that the personal data processing is fair, lawful and transparent. PETs cannot be viewed as a 'silver bullet' which can single-handedly obviate risk – it must be considered in tandem with other measures to ensure data protection compliance.

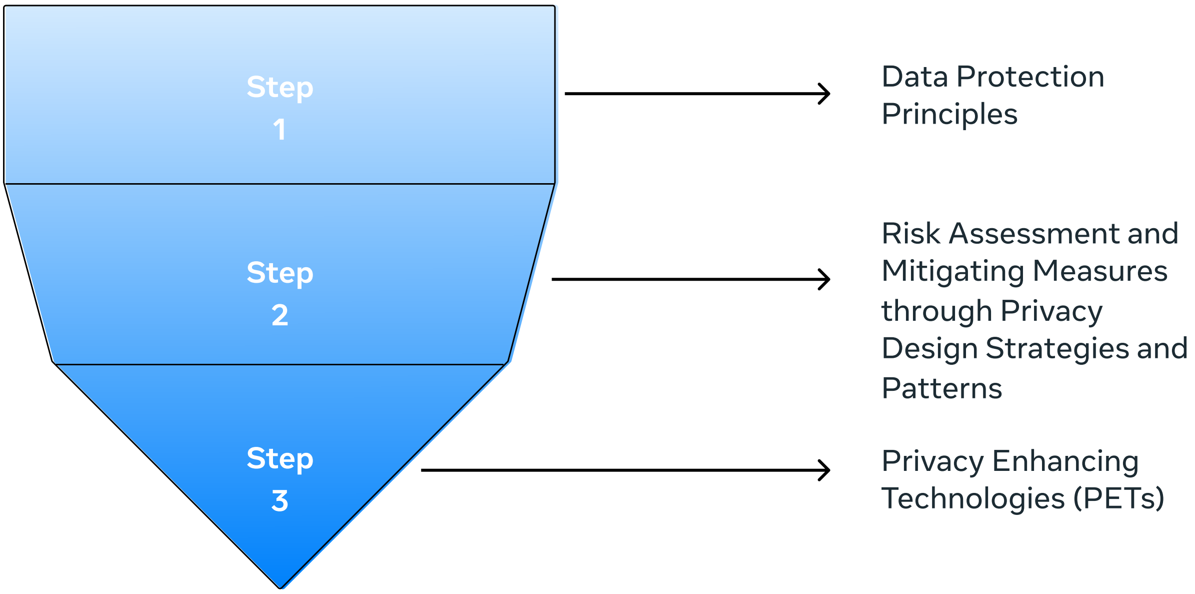
Besides applying PETs, other procedures to reduce risk could include, among other things: (i) legal or contractual measures obligating parties to not attempt to re identify the data; (ii) controls (technical or otherwise) on who can access the data and associated data that might be used to re identify the data; and (iii) protocols for deleting or otherwise sequestering the data after a certain period time.

PETs do play a valuable role in operationalizing privacy by design requirements. Therefore, the aim of this playbook is to advance the conversation on how to supplement privacy by design requirements with privacy-enhancing technical measures, so as to help data controllers and developers develop their systems in a privacy-friendly way.

Visualization of the Funnel Approach

This playbook utilizes a ‘funnel approach’ towards risk mitigation. The approach is may be used to:

1. Identify broad privacy requirements[2] that could be implemented.
2. Evaluate privacy risks, connect them to potential privacy requirements and attribute appropriate design strategies thereafter to remedy them
3. Simplify the risk mitigation strategies into technical measures, using Privacy Enhancing Technologies (PETs).


Step 1
Step 2
Step 3
Data Protection Principles
Risk Assessment and Mitigating Measures through Privacy Design Strategies and Patterns
Privacy Enhancing Technologies (PETs)

This approach is meant to help data controllers and developers meet privacy standards while developing their products and services. Applying the ‘funnel approach’ throughout the data lifecycle can help ensure that the outcome of ‘privacy by design’ can be achieved holistically, and that risks are appropriately identified across each stage/component. The latter is particularly crucial in aiding controllers in determining the appropriateness of the PET(s) they would like to implement.

1.1 Reading Guide

This playbook aims to bridge the gap between privacy expectations and technology solutions by associating Privacy Enhancing Technologies (PETs) with data protection risk assessment process through privacy by design strategies. By doing so, it hopes to empower data controllers to process data in a privacy-centric manner.

This guide starts by setting out the data protection principles in Section 2.

In Section 3, we will describe a 3-step process for operationalizing privacy by design principles while connecting them to the adoption of PETs, namely:

1. **identify and assess privacy-related risks in the proposed design of the system (product / service),**
2. **propose risk reducing measures using design strategies and patterns across the data lifecycle/information system components, and**
3. **select concrete risk mitigation approaches using suitable PETs.**

To provide further guidance, Appendix I is a glossary of terms that will be useful when reading the playbook. Appendix II describes in more detail commonly used PETs, how they work in practice, their main limitations and where engineers can find further resources. Appendix III provides technical considerations when adopting PETs. Appendix IV provides additional core technical use cases for engineers to consider with privacy risks mitigated by using PETs. Finally, Appendix V, provides further consideration when applying PETs in an AI context. It also discusses specific privacy harms and risk dimensions across the AI system lifecycle and provides examples of specific issues related to this.

2 Data protection principles

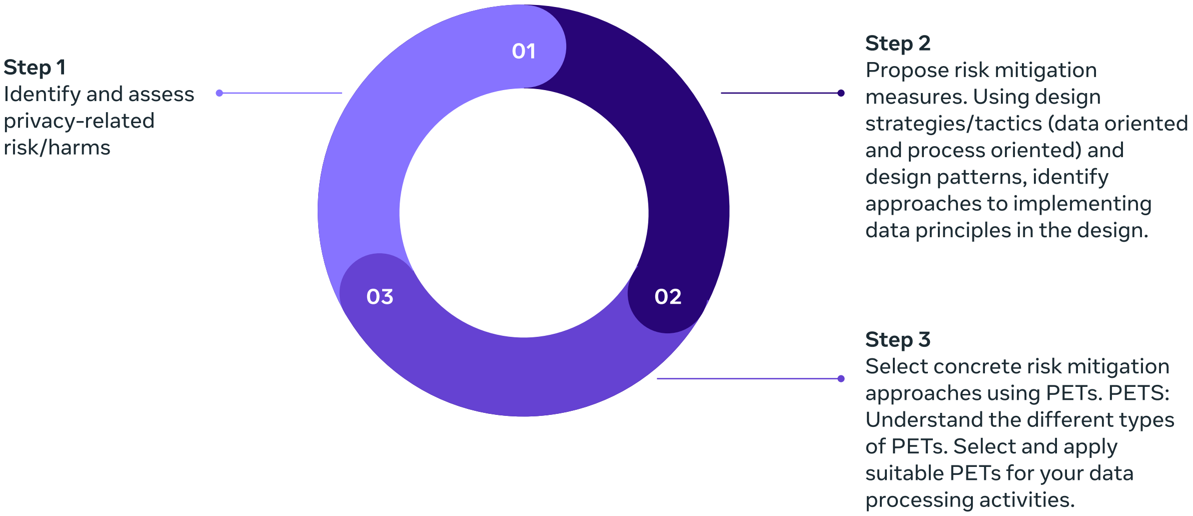
Understanding how personal data processing can affect the privacy of individuals is key to designing and developing trustworthy systems from the point of view of data protection.

Different data protection laws and privacy related frameworks (e.g. OECD Basic principles on national application, the EU's General Data Protection Regulation, Brazil's LGPD, Uruguay's privacy law and guidance on data protection, the Ibero American Network Standards, the Inter-American Bank Principles for Digital Development) list the basic principles to be adhered to when processing data, such as: Lawfulness, fairness, and transparency, Purpose specification and limitation, Data minimization, Accuracy, Storage limitation, Integrity and confidentiality, and Accountability. These frameworks also require data controllers to be accountable, as they are responsible for meeting these requirements.

3 The 3-step process for advancing data protection principles using PETs

While the data protection principles help determine when the processing of personal data is legitimate, it is hard translating these abstract principles to concrete functional requirements / specifications and design solutions. In this section we will describe a 3-step process that could help data controllers achieve compliance with the data protection principles, namely: Lawfulness, Fairness, and Transparency, Purpose specification and limitation, Data minimization, Accuracy, Storage limitation, Integrity and confidentiality, and Accountability.

To help design teams, product owners, engineers, and others to translate data protection principles to concrete design solutions including PETs, this playbook proposes the following 3-step approach:



Step 1
Identify and assess privacy-related risk/harms

Step 2
Propose risk mitigation measures. Using design strategies/tactics (data oriented and process oriented) and design patterns, identify approaches to implementing data principles in the design.

Step 3
Select concrete risk mitigation approaches using PETs. PETS: Understand the different types of PETs. Select and apply suitable PETs for your data processing activities.

Step 1: Risk Assessment: Privacy and data protection risks

3.1.1 Assessing privacy/data protection risk

The first step for advancing data protection principles using PETs is identifying and assessing the risks of a data processing activity. Risk may be understood as the likelihood of a particular harm happening. Whether something is a big or a small risk is dependent on the impact of the risk and the likelihood of that risk manifesting itself.

When looking at privacy risks we must therefore assess:

- what the negative impact of an unwanted / unintended consequence of our data processing activity is (i.e., the privacy harm), and
- what the chance is of that impact manifesting itself by looking at the root causes of privacy harms.

3.1.2 Privacy-related harms

Privacy harms refer to the negative consequences that can arise when an individual's personal information is collected, used, or disclosed in a way that infringes on their privacy rights.

Understanding the potential impact of data processing on individuals requires consideration of the various types of privacy-related harm that may result from the misuse or abuse of personal data. This could include harm to an individual's physical, psychological, economic, reputational, autonomy, relationships, or potential for discrimination. Examining these different types of harm can provide insight into the potential risks and consequences of improper handling of personal data.

It is important to note that these categories of harm are not mutually exclusive and that a person may experience multiple harms as a result of a privacy breach. The harms experienced by an individual may also depend on their particular circumstances, such as their age, gender, race, or social status. By understanding the potential harms associated with data processing, data controllers and developers can take steps to minimize the risks and ensure that privacy is protected in a way that meets the needs and expectations of individuals.

These harms can occur as a result of unwanted, unintended, or illegal processing or breach of personal data. To understand the chance that these harms might occur, it would be useful to look at the potential root causes of harm.

3.1.3 Root causes of privacy and data protection risk

The privacy harm that can result from data processing activities can occur without intention and regardless of the structure of the processing. The specifics of the harm could depend on the individual involved and its circumstances. For example, consider an online retailer that collects credit card information from its customers only for payment purposes. However, if the retailer fails to implement proper security measures to protect this data, it could result in a data breach caused by malicious actors exploiting the vulnerabilities in the system. This could lead to economic harm, as well as relationship harm between the retailer and its customers. That's why it's crucial to consider "privacy by design" in information systems. This involves designing the system in a way that minimizes the potential for privacy harm.

By critically assessing the design of your system, and by properly implementing PETs, you may remove many of the root causes of privacy and data protection risk, ultimately lowering the chance of a privacy harm manifesting itself, and the potential negative impacts on the data subject.

3.1.1 Assessing privacy/data protection risk

Privacy risks may manifest across all stages of the data life cycle, i.e: collection, storage, processing, transfer, usage and disposal/deletion. The data life cycle is the sequence of events that data go through from their initial generation or collection all the way to their deletion. This can be visualized as follows:

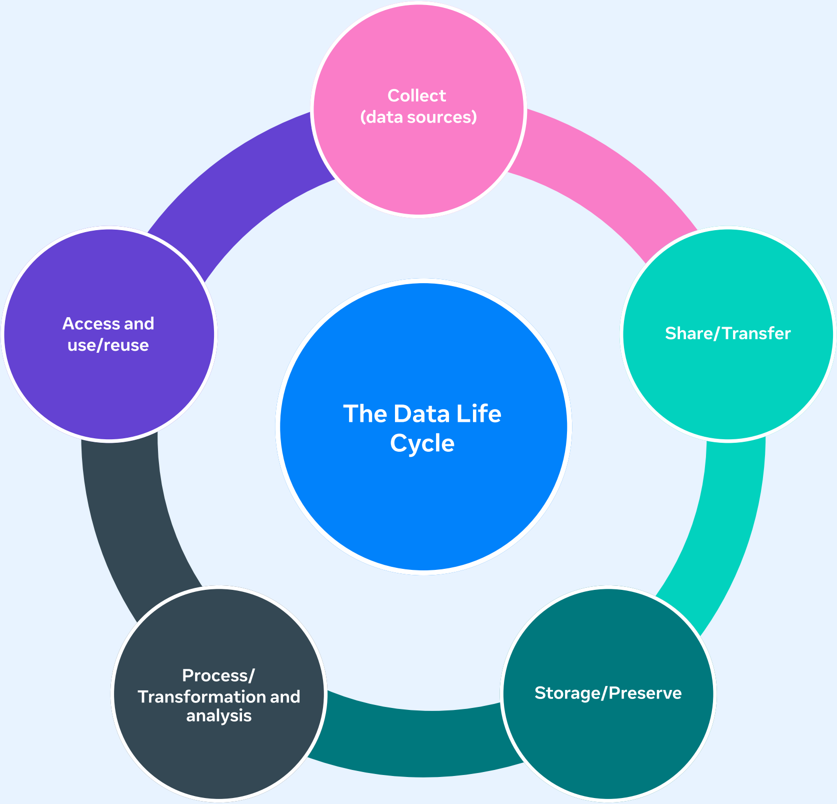


Figure 3: Depiction of the Data Lifecycle

While the data life cycle provides an abstract framework for understanding data processing, a more concrete approach is looking at the actual information system that is processing the personal data.

An information system is a system that can ingest data, store it, use it (i.e., by transforming or analyzing the data) and make it available to other systems or people. An information system can be a data lake, an app, a service, an AI system, etc. The components of an information system can be visualized as follows:

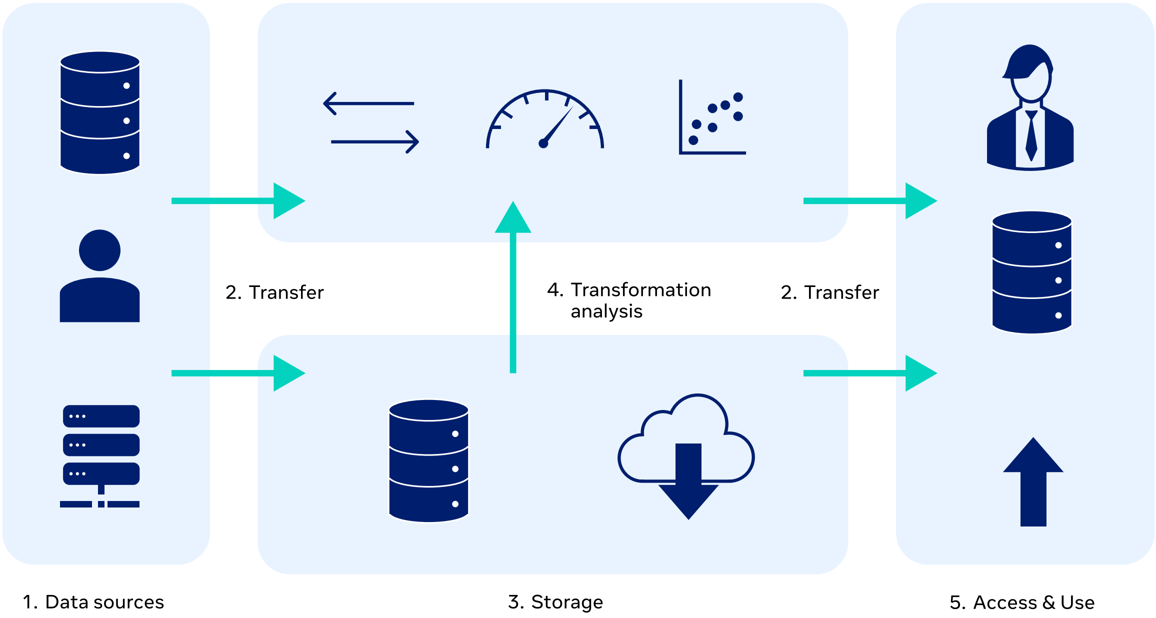


Figure 4: Depiction of an Information System

Note: Information systems generally collect / ingest data from an external data source (1). Data can be collected from third party databases, entered by a user in a webform, observed on a website, gathered by sensors et cetera. These data are transferred (2) from the original data source to the information system. For instance: data can be collected from a third party via an API and then transferred using a secure connection. Once data is ingested, it is (temporarily) stored (3) in the information system. Once the data has been stored it may be used for its actual purposes, which might be either analysis (4), or access and (re)use (5), or both. For instance, a marketeer may access (5) the data in the information system and use it to do a particular analysis of user behavior and preferences. Data from the system may also be transferred (2) to third parties.

These stages could roughly plot the data life cycle on an information system:

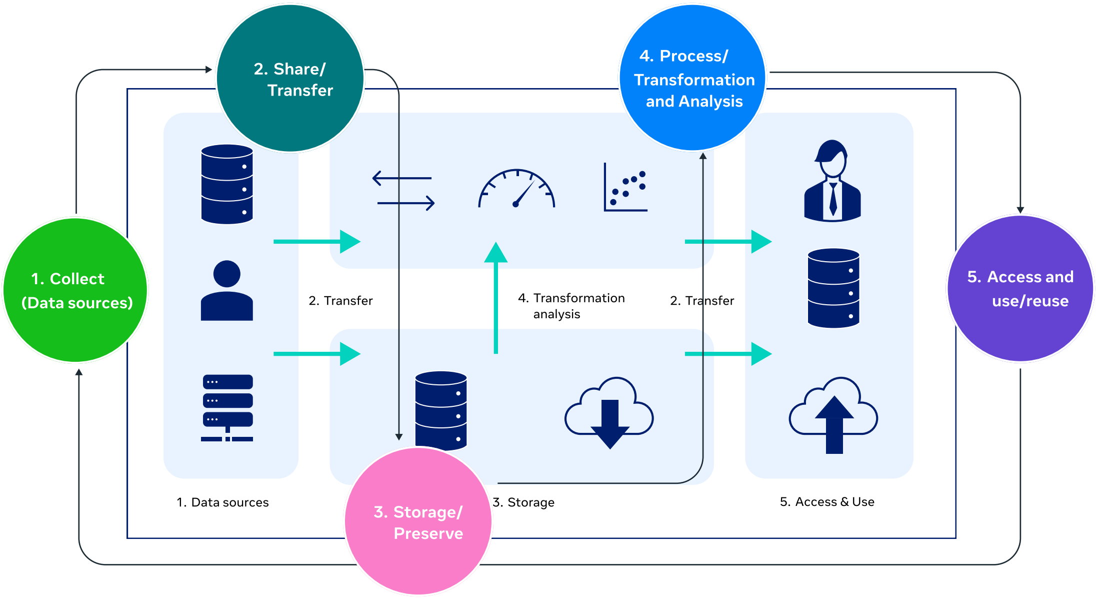


Figure 5: Plotting Data Lifecycle Stages to Components of the Information System.

Note: The first stage of the data life cycle, collection, can be mapped to the first component of the information system, data sources. The life cycle is initiated when generated data located within data sources are collected by the system. The second stage of the data life cycle, sharing/transferring, can be plotted to the data transfer component of the information system. Transferring data allows the data to be stored, so that it may be further processed for different purposes. The third stage of the data life cycle, storing/preserving can be linked to the storage component of the information system, if applicable. Data is (temporarily) stored most commonly through datasets or in databases. This process also includes the management of the data, through organizing, implementing security protocols and retrieving it. The fourth stage of the data life cycle, processing, can be mapped to the transformation analysis component of the system. It is here that raw data is transformed and enriched into data that can provide true value and insight to not only your organization or project, but also to authorized others. Finally, the fifth stage of the data life cycle, accessing and (re)using can be mapped to its counterpart in the information system. Given the wide nature of potential uses the data can have, it can be interpreted, visualized, or further analyzed. The data in this stage can be further relied upon by authorized third parties as part of a data source, to start the data lifecycle process all over again.

By assessing the privacy and data protection risks associated with all of these 'components' you can get an overview of data protection risks (i.e., leaking data, storing data for too long, unauthorized access et cetera).

Below we will set out what potential privacy risks are commonly associated with each of these components and which data protection principle they may be found to infringe upon.

3.1.4.1 Common risks associated with Data sources

The first step for advancing data protection principles using PETs is identifying and assessing the risks of a data processing activity. Risk may be understood as the likelihood of a particular harm happening. Whether something is a big or a small risk is dependent on the impact of the risk and the likelihood of that risk manifesting itself.

When looking at privacy risks we must therefore assess:

- what the negative impact of an unwanted / unintended consequence of our data processing activity is (i.e., the privacy harm), and
- what the chance is of that impact manifesting itself by looking at the root causes of privacy harms.

Data protection principle	**Common risks and considerations**
Lawfulness, fairness, and transparency	• Data source does not have a legitimate purpose (e.g., illegally collected data). • The controller has no legitimate purpose for using data from the source. • The controller does not fulfill its transparency obligations..
Purpose specification and limitation	• The underlying purpose for collecting the data from the source is not specified. • Data source has specified a different purpose for that from which it was originally for
Data minimization	• Ingesting more data than necessary for the purposes of the processing
Accuracy	• Data is not accurate / representative
Storage limitation	• The data is being stored for a period longer than the stated retention policy
Integrity and confidentiality	• Data is not being stored with sufficient security measures
Accountability	• The organization collecting the data should be responsible for ensuring that the data is protected

Table 2: Risks to Data Source Component

3.1.4.2 Common risks associated with Data Transfer

Data protection principle	Common risks and considerations
Lawfulness, fairness, and transparency	• Data transfers are not known by data subjects, or it is known but the data subjects are unaware about how the transfer is done.
Purpose specification and limitation	• More data are ingested than necessary for the purpose • The use of data by third parties for other purposes for which it was originally collected for. If the third parties use the data for other purposes, this may result in unauthorized data processing, which can compromise the privacy of the data subjects.
Data minimization	• More data is ingested than necessary for the purpose.
Accuracy	• Data is incorrectly ingested, which can cause data loss or can adversely affect the quality of input data
Storage limitation	• Data is stored in an intermediary system without retention periods set.
Integrity and confidentiality	• Interfaces are not properly secured, data may be intercepted, accessed, or manipulated by unauthorized parties • Connections are not secure, allowing for eavesdropping on communications
Accountability	• Data ingestion is not properly logged, and data sources not properly cataloged, due to the vast frequency, velocity, size, and different formats of data being ingested by the information system.

Table 3: Risks to Data Transfer Component

3.1.4.3 Common risks associated with Storage

Data protection principle	Common risks and considerations
Lawfulness, fairness, and transparency	• The data controller does not inform the data subject the duration of data retention. • The data controller has no legitimate purpose for storing the data. • The data controller misleads the data subject into believing that storage is necessary to achieve the purpose of processing.
Purpose specification and limitation	• The stored data is further processed, for a purpose that is incompatible.
Data minimization	• Storing more data than necessary for the purposes of the processing. • There are no distributed/decentralized storage and analytics facilities.
Accuracy	• The data that is stored becomes outdated. • Stored data is valid but was entered inaccurately by a user, which cannot be rectified or deleted.
Storage limitation	• The data is stored longer than what is necessary for the purpose of processing.
Integrity and confidentiality	• Cryptographic techniques, such as encryption, are not used to secure data at rest. • Inadequate backup and disaster recovery: Data may be stored without proper backup and disaster recovery procedures in place, leaving it vulnerable to loss in the event of a hardware or software failure, natural disaster, or other unforeseen event. • Data stored in legacy systems with no rule-based access control. • All data is stored on a central server, with no separation of different identifiers of data subjects. • Insufficient physical security: Data storage systems may be physically insecure, making them vulnerable to theft or other physical attacks.
Accountability	• Improperly cataloging of the storage location of different data points of data subjects. • Justification for why technical measures have not been taken to secure the data, have not been recorded.

Table 4: Risks to Storage Component

3.1.4.4 Common risks associated with Transformation / Analysis

Data protection principle	Common risks and considerations
Lawfulness, fairness, and transparency	• Bias within the input datasets manifest themselves in the analytical, output stage, causing legal, or similarly significant effects. • The controller allows mission creep to set in, by allowing data to be processed for purposes other than what they were initially collected for. • No implementation of de-identification, aggregation or pseudonymization policies where appropriate. • Not informing, or inadequately informing the data subjects about the purpose and nature of processing.
Purpose specification and limitation	• The data controller processes data beyond the purposes they were initially collected for. • The data controller improperly classifies/ labels data attributes.
Data minimization	• No implementation of blacklisting, aggregating, or stripping, or destroying policies, to minimize data processing where appropriate.
Accuracy	• No implementation of data lineage to be able to trace and rectify errors and regenerate lost input.
Storage limitation	• Storing the input data, even after the analytical process is complete, and when it might be no longer required.
Integrity and confidentiality	• No implementation of aggregation or de-identification techniques when analyzing data, where appropriate. • No implementation of cryptographic techniques like homomorphic encryption, when analyzing data, where appropriate.
Accountability	• No record, or unspecified record of the kind of analytical/ statistical activities conducted on the data • No data protection impact assessment for high risk processing to data subjects • No logging of file analysis.

Table 5: Risks to Transformation/Analysis Component

3.1.4.3 Common risks associated with Access and Use/Reuse

Data protection principle	**Common risks and considerations**
Lawfulness, fairness, and transparency	• There is no, or inadequate privacy statement, detailing how personal data will be processed. • The reliance on erroneous or biased datasets leads to inconsistent treatment or unfair outcomes to data subjects • Incompatible further processing of data is allowed, without the consent for that processing being acquired.
Purpose specification and limitation	• The data controller allows mission creep to set in, allowing data to be further processed for purposes other than what they were initially collected for.
Data minimization	• The data controller does not use deanonymization techniques, or other PETs, such as differential privacy, tokenization, or multi-party computation, thereby allowing parties to unjustifiably access troves of either raw data or identifiable, analyzed datasets, where appropriate.
Accuracy	• No implementation of ACID Principles where appropriate. • No implementation of CRUD API where appropriate.
Storage limitation	• Data is not labeled properly, thus preventing automatic deletion based on the nature of the data, where applicable. • No implementation of anonymization techniques or deletion, after the stipulated data retention period is complete, where applicable. • No implementation of privacy management tools by the controller, where applicable.
Integrity and confidentiality	• No use of firewalls or up to date cryptographic methods, such as homomorphic encryption, to protect the data where appropriate. • No user rights management mechanisms to monitor data access and activities of privileged users to identify excessive, inappropriate, and unused privileges where appropriate.
Accountability	• No audit trail to be able to trace how/why data was processed, and which parties processed it, where appropriate.

Table 6: Risks to Access and Use/Reuse Component

3.2 Step 2: Propose risk reducing measures

Once the risks have been identified, you can propose risk reducing / mitigating measures to help minimize privacy risks. Ideally, these risk reducing measures are implemented in the design of the system.

One way to make a design that is privacy compliant, is for the data protection principles set out in Section 2 to be incorporated into your design. To help translate privacy requirements to concrete functional and non-functional requirements for information systems, a recommended approach is to use privacy by design strategies.

Privacy by design strategies help design and engineering teams to approach privacy problems in a structured way and come up with solutions. Privacy by Design strategies help translate privacy requirements (Section 2) into concrete design requirements. They provide actionable directions to explore the design of the system. They guide the initial design sketches into a privacy-friendly direction, forcing one to make fundamental design choices early on.[13] They also provide a way to address the risks identified in step 1 of our three-step process to addressing privacy issues. The privacy by design strategies helps in guiding the process of ideation and definition.

When it comes to the actual design of a system, design patterns are helpful. Using the design strategies as guidance, you can look towards relevant design patterns. Design patterns are common ways to approach a design problem. In software engineering a design pattern is defined as:

“a scheme for refining the subsystems or components of a software system, or the relationships between them. It describes a commonly recurring structure of communicating components that solves a general design problem within a particular context.”[14]

So, the privacy by design strategies will point you towards design patterns that you may incorporate in the design of your system.[15]

Finally, when it comes to the actual development of the system, we can turn to PETs which may provide concrete solutions for identified privacy and data protection risks.

To provide further clarification, design strategies are at a higher level of abstraction, and only stipulate a fundamental approach to achieve a particular goal. Therefore, the intended outcome of a privacy design strategy is to achieve privacy and data protection. It does not impose a particular structure, only the goal (the ‘what’) a controller needs to achieve. The tactics within each strategy also help in pointing towards design patterns that can be deployed to achieve an information system that protects privacy. Furthermore, these strategies are fundamental during the ideation (concept development) and definition phases of developing an information system.

Design patterns are at a comparatively lower level of abstraction and provide guidance on how to strategically implement the strategies. They rely on tried and tested solutions to solve generic and recurring issues within a particular context and are therefore most relevant during the design phase of the information system. A design pattern might also help realize multiple design strategies. Within specific departments, or ‘organizational units’ of a controller, these patterns help translate the ‘what’ into the ‘how’, through concrete plans and goals.

PETs further build on the ‘how’ by specifying which concrete technologies may be best suited to implement identified design patterns. They are the result of operational decisions taken to ensure that, at the ground level, the information system integrates the appropriate technical features. Design patterns and PETs are grouped together, as not only do some PETs implicitly define design patterns, but also because PETs complement design patterns. They do so by providing granular guidance on the exact nature of the technology required, during the development (implementation) phase.

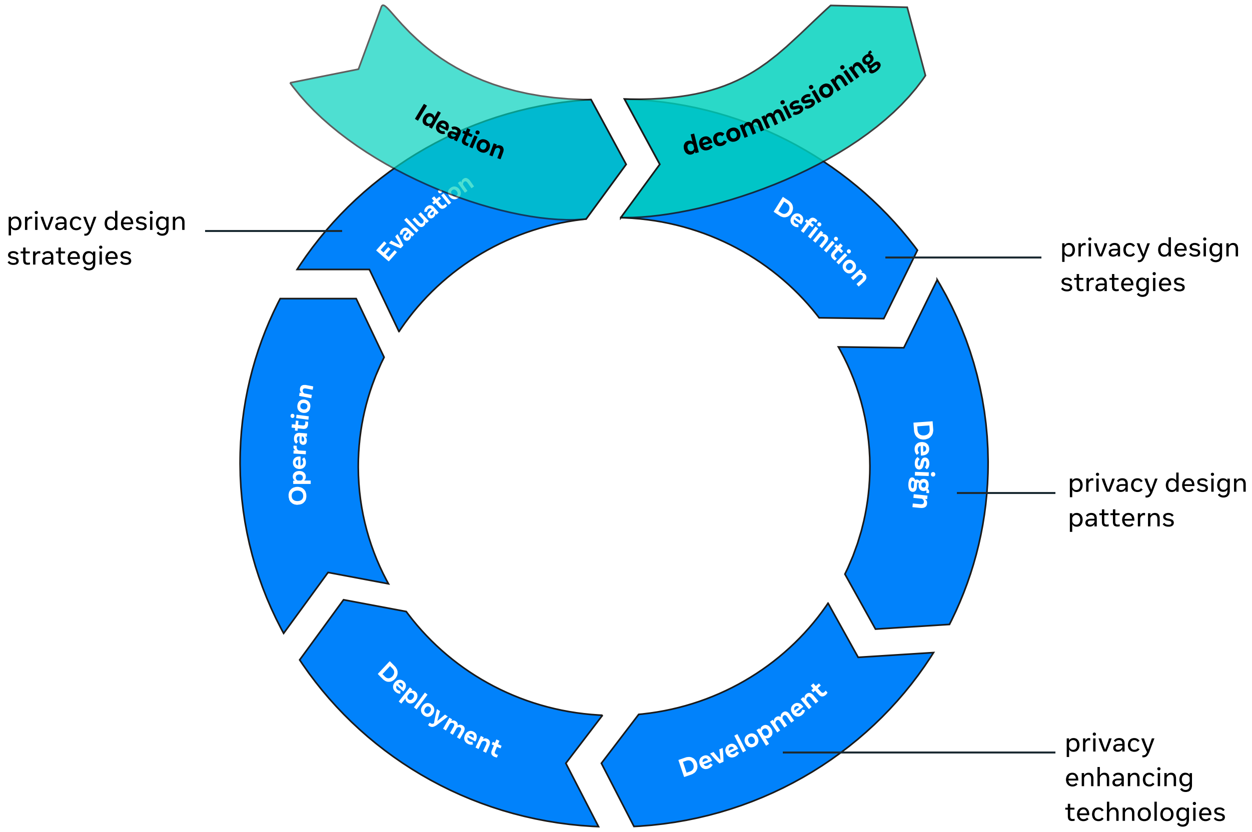


Figure 6: The Information System Development Cycle and its relationship with strategies, patterns, and PETs 16

3.2.1 Privacy by design strategies

There are eight privacy by design strategies: 4 data-oriented strategies (minimize, separate, aggregate, hide) and 4 organization/process-oriented strategies (inform, control, demonstrate, enforce).[17]

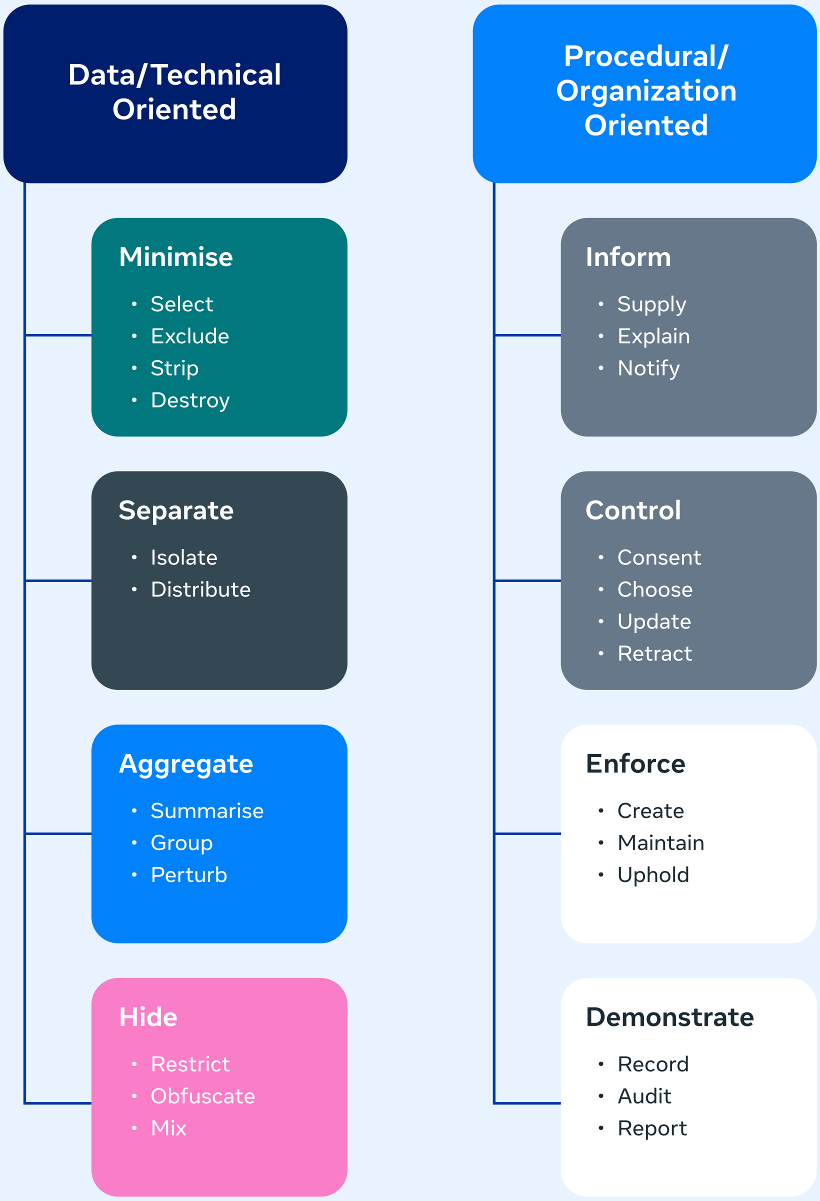


Figure 7: Privacy by Design strategies[18]

Each of the design stratigies can be further broken down into design tactics. For instance, the MINIMIZE strategy can be implemented in several ways: you can ensure that certain data are not collected ('select before you collect'), you can scrub data after ingestion and only keep what is relevant ('strip and destroy'). You can remove all direct and indirect identifiers of a person, or you can pseudonymise the data after ingestion.

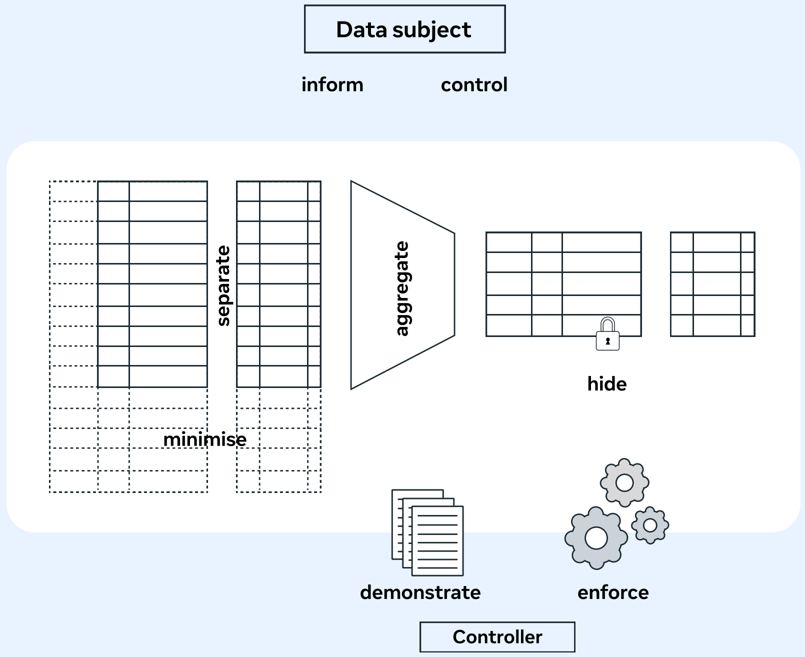


Figure 8: Illustration of Privacy by Design Strategies [19]

The figure above is a representation of how data is stored in an information system- in a tabular format. In this database, a fixed set of attributes are stored in columns, while rows indicate the addition of individuals whose data is to be stored. Since data collection ought to be proportionate to the envisaged purpose, the dotted cells in the table are representative of data that must not be collected, as they are not required for the purpose. The demarcation between the columns indicate that attributes of an individual collected for a particular purpose must be stored separately from data collected for another purpose. When there is no longer a requirement to store multiple data points of an individual, the information in the table is aggregated, denoted by a quadrangle, and stored. As a matter of best practice, all data, aggregated or otherwise, should be kept hidden from unauthorized parties.

These are the data oriented strategies, and they are placed at the core of the illustration, as they technically implement the privacy requirements of an information system. Since the development of PETs has primarily been focused on strengthening these technical features, they are more closely linked to data oriented strategies.

Organizational measures, or process oriented strategies, are meant to supplement the technical measures, with privacy friendly conduct by the parties who interact with the system, and the data contained therein. It is also important that these strategies are depicted as surrounding the data oriented strategies, as it is a continuous process. Failing to implement any of them could potentially vitiate the privacy gains within the information system, when implementing data oriented strategies.

Privacy is depicted as being a relationship between a data subject and a controller. Therefore, a data subject should generally be informed about how their data is being processed. Privacy by design can only be attained if there is a privacy policy that accurately describes how the collection and processing of data takes place. Finally, the controller ought to be able to demonstrate, whenever required, compliance with the requirements stipulated under the pertinent data protection regimes.

3.2.2 Data Oriented Strategies

3.2.2.1 Minimize

Limit as much as possible the processing of personal data to achieve a particular purpose. Minimization of personal data can be achieved by collecting data of fewer people or collecting less data of people. By ensuring that no unnecessary data is collected, the possible privacy impact of a system is limited. Anonymization and Pseudonymization techniques are instances of design patterns that can help achieve this strategy. Additionally, four tactics can be used to achieve this design strategy:

- Select only relevant people and relevant attributes. Determine beforehand which people and which attributes are relevant, and process only that data.
- Exclude people or attributes in advance, by determining beforehand which people or attributes are irrelevant.
- Strip (partial) data as soon as it is no longer necessary, by removing it. Determine beforehand the time you need a particular data item, and ensure it gets automatically deleted as soon as this time expires.
- Destroy personal data, by completely removing them as soon as they are no longer relevant. Ensure that the data cannot be recovered, even in unforeseen ways.

3.2.2.2 Separate

Separate the processing of personal data as much as possible, both logically and physically. By separating the processing or storage of several sources of personal data that belong to the same person, it can also help achieve purpose limitation. By processing personal data from different contexts separately, the risk that data from one context becomes known in another context is reduced. Hence separation implements contextual integrity. Decentralized, peer-to-peer networks, and distributed algorithms, are examples of design patterns that can be used to implement separation. Additionally, two tactics can be used to achieve this design strategy:

- Isolate, by collecting and processing personal data in different databases or applications. These databases or applications are either logically separated or run on different (yet still centrally located or controlled) hardware.
- Distribute the collection and processing of personal data over different physical locations using databases and systems that are not under the control of a single entity. Use the equipment (PC, laptop, smartphone) of the data subject himself as much as possible and use central components as little as possible.21

3.2.2.3 Aggregate/Abstract

Limit as much as possible the detail in which personal data is processed, by processing data at the highest level of aggregation where possible. The privacy of individuals is protected when data items are general enough that the information stored is valid for many individuals, with little information being able to be attributed to a single person. Anonymization techniques and differential privacy are design patterns that can be used to successfully achieve this strategy (See Appendix III). Additionally, three tactics can be used to achieve this design strategy:

- Summarize detailed attributes into more coarse-grained, general attributes.
- Group information about a group of people instead of processing personal information for each person in the group separately, through aggregation.
- Perturb the values of data items to be processed, by either using an approximation or by including random noise to adjust the value. [22]

3.2.2.4 Hide

Protect personal data, or make it un-linkable or unobservable, by making sure it does not become public or known. The rationale behind this strategy is that by hiding personal data from plain view, it cannot easily be abused, and can therefore aid in ensuring confidentiality and integrity of personal data. De-identification and cryptographic techniques are design patterns that can aid in implementing this strategy. Additionally, four tactics can be used to achieve this design strategy:

- Restrict access to personal data. Ensure personal data is properly protected, and set up a strict access control policy.
- Obfuscate the understandability of personal data to those without the ability to decipher it. Encryption of data can help ensure that the personal data is unintelligible without the key.
- Dissociate the correlation between events, persons, and data, by breaking the link between them. Remove directly identifying data
- Mix personal data to hide the source or their interrelationships, through de-identification techniques[23]

3.2.3 Process Oriented Strategies

In this section we describe a selection of the process-oriented strategies. Note that the focus of this playbook will be on the data-oriented strategies as most PETs are not only data oriented but are also technical in nature.

3.2.3.1 Inform

Inform data subjects about the processing of their personal data in a timely and adequate manner where possible. Transparency about which personal data is being processed, how they are processed and for which purpose, is an essential prerequisite for better privacy protection. It allows users to make informed decisions about using a system and the processing of their personal data. Platform Privacy Preferences, and Dynamic visualization of privacy policy are examples of design patterns to achieve this strategy. Tactics that can be used to achieve this design strategy, include:

- Supply information about which personal data is processed, how they are processed, and why. The information should also include the categories of third parties, the duration of personal retention, etc.
- Explain which personal data you process, and why.[24]

3.2.3.2 Demonstrate

Demonstrate that the processing of personal data is being done in a privacy friendly way. This requires the data controller to be able to show how the privacy policy is effectively implemented within the IT system. This strategy helps in complying with the accountability principle and goes one step further than the 'Enforce' strategy in that it requires the data controller to prove that it is in control.. Privacy management systems, and the use of logging and auditing are good examples of design patterns that can be used to implement this strategy. Additionally, three tactics can be used to achieve this design strategy:

- Record all (important) steps taken. Document decisions and motivate them.
- Audit not just the logs regularly, but also the organizational processes in general, and the way personal data is processed within the organization.
- Keep the results of such audits for later reference. Consult the DPA regularly.[25]

3.2.3.3 Enforce

Commit to processing personal data in a privacy-friendly way, and adequately enforce this, through a privacy policy that is compatible with legal requirements, and appropriate governance structures to enforce the said policy. This strategy fulfills both the accountability principle, and purpose limitation. Access control, sticky policies and privacy rights management are design patterns that can be used to successfully implement this design strategy. Additionally, three tactics can be used to achieve this design strategy:

- Creation of a privacy policy for an organization is crucial to show its commitment to privacy and is evidence that it is willing to take responsibility.
- Maintain the policy with all the necessary technical and organizational controls.
- Uphold, consistently, the implementation of the privacy policy and adjust it for it to align with the overall mission and business plan of the organization. Therefore, verify the privacy policy regularly.

3.3 Step 3: Select relevant PETs

A specific approach to practically implement privacy by design strategies and patterns is the use of Privacy Enhancing Technologies (PETs). Privacy Enhancing Technologies can be defined as:

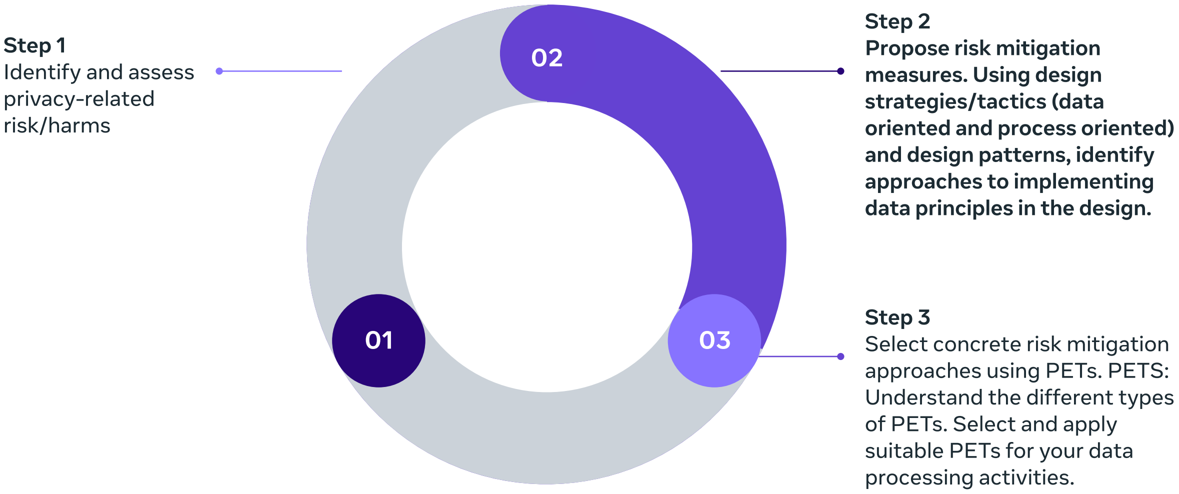
> "a system of ICT measures protecting informational privacy by eliminating or minimizing personal data thereby preventing unnecessary or unwanted processing of personal data, without the loss of the functionality of the information system."[27]

Privacy-enhancing technologies (PETs, privacy-enhancing, privacy-preserving, and privacy-protecting are used as equivalent terms) involve advanced techniques drawn from the fields of cryptography and statistics. These techniques help minimize the data that is processed by or made visible to an organization while preserving critical functionality.

For more details about the different types of PETs, see Appendix II of this Playbook.

Below, this playbook proposes different design strategies and relevant PETs. Note that PETs are mainly focused at eliminating and minimizing personal data. So, they may not provide a solution for all identified privacy risks. As such, most of the PETs will be relevant in the minimize, separate, hide and aggregate strategies.

PETs are approaches and concrete applications to enhance privacy and data protection. They can be used to operationalize a selected privacy by design strategy (as described in Step 2). By identifying the relevant privacy by design strategies and its associated tactics you can select the proper PETs to implement. This logic can be summarized as follows:



Step 1
Identify and assess privacy-related risk/harms

Step 2
Propose risk mitigation measures. Using design strategies/tactics (data oriented and process oriented) and design patterns, identify approaches to implementing data principles in the design.

Step 3
Select concrete risk mitigation approaches using PETs. PETS: Understand the different types of PETs. Select and apply suitable PETs for your data processing activities.

Table 7: Mapping PETs, and their examples, to different Privacy by Design Strategies

Privacy by Design Strategy	PETs	Examples of PETs
MINIMIZE	• De-identification techniques • Differential Privacy (data altering) • Pseudonymization techniques (data altering) • Synthetic Data (data altering)	**Differential Privacy** https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance.html The US Census Bureau applied Differential Privacy as a technique to meet its legal obligations for individual-level disclosure avoidance prior to releasing Census data to the public. This use of differential privacy prevents individual level identification from occurring and allowed the Bureau to release its data to public researchers. **Differential Privacy** https://leapyear.io/ The company LeapYear claims to use differential privacy on its platform, which serves a variety of industries, including healthcare, retail banking, and capital markets. The company claims to use differential privacy to offer solutions that allow for insights and analytics into data while preserving the privacy of their client's data. **Pseudonymization techniques** https://royalsociety.org/-/media/policy/projects/privacy-enhancing-technologies/privacy-enhancing-technologies-report.pdf The NHS has built a system for linking patient data held across different NHS domains. "To protect patient confidentiality, identifiers (such as a patient's NHS number) are pseudonymized through tokenization." For additional security, the tokenization differs between different NHS domains. Linking data about a patient held in two domains first requires removing the tokenization which would expose personal information. **Synthetic Data** https://www.gcffc.org/wp-content/uploads/2020/06/FFIS-Innovation-and-discussion-paper-Case-studies-of-the-use-of-privacy-preserving-analysis.pdf Enveil has designed and developed an approach for financial institutions to identify matching customer information in external datasets without disclosing information about that customer. This allows financial institutions to investigate suspicious activity without revealing personal information, especially in the case that the customer being investigated is ultimately innocent. This proof of concept has been executed using synthetic data and also showed that information can still be made visible for audit, traceability and trust building where required. **Synthetic Data** https://www.gcffc.org/wp-content/uploads/2020/06/FFIS-Innovation-and-discussion-paper-Case-studies-of-the-use-of-privacy-preserving-analysis.pdf Duality Technologies was tasked with a pilot initiative to support banking institutions in investigating financial crime and compliance. Specifically, the AML and Fraud pilots involved banks from Canada and the UK, respectively. Duality relied on synthetic data to carry out the pilots. The synthetic data involved in the project consisted of several tables including account information, telephone numbers, names, transactions, transaction amounts, card information and other fields. Used along with Homomorphic Encryption, it enabled participants to make queries of data without the query being disclosed to the requested party (data owner).

SEPARATE	- Federated Learning/ Analytics (FL) (computation altering) - Trusted Execution environments (data shielding) - Secure Multi-Party Computation (SMPC)	**Federated Learning** https://venturebeat.com/2020/09/17/major-pharma-companies-including-novartis-and-merck-build-federated-learning-platform-for-drug-discovery/ https://owkin.com/de-DE/publications-and-news/blogs/federated-learning-in-healthcare-the-future-of-collaborative-clinical-and-biomedical-research https://owkin.com/publications-and-news/press-releases/owkin-shares-new-ai-severity-score-for-covid-19-integrating-ct-images-published-to-nature-communications https://tech.eu/2022/07/13/eu-backed-healthcare-project-rolls-out-platform-for-federated-learning-in-drug-discovery/ In partnership with Nvidia, Owkin and other pharmaceutical companies leveraged techniques including FL to collectively train AI on datasets without having to share any proprietary data. This work allowed the companies to build a foundation for accelerating medical research on clinical trials for drugs. In 2021, Owkin previously used its FL software Owkin Connect to facilitate a partnership across two French hospitals to build a model that could analyze multimodal health data (CT images of the lung, radiology reports, and a variety of clinical and biological data points). This model was used to create a ‘COVID-19 AI severity score’ for recently diagnosed patients. https://www.rhinohealth.com/ https://www.globenewswire.com/en/news-release/2022/05/05/2436894/0/en/Rhino-Health-Platform-Powers-Hospital-Based-Federated-Learning-Consortium.html Rhino Health is a startup that leverages federated learning to enable healthcare AI development on distributed patient data without transferring data from the original healthcare institution. It is attempting to improve an AI algorithm that is focused on brain aneurysm detection. https://www.sherpa.ai/technology/ Sherpa.ai, an AI company based in Spain, claims to offer an FL solution to create AI/ML models trained on decentralized data, such as data located in users’ smartphones, hospitals, or banks. **Federated Learning** https://ai.googleblog.com/2017/04/federated-learning-collaborative.html GBoard is a keyboard app for Android and iOS devices. It features next-word prediction, driven by a machine learning model. GBoard utilises federated learning where each mobile device downloads an initial model from a central server, which is further trained on the device using user data local to the device. The weights of the resulting model are periodically communicated back to the central server using a secure aggregation protocol (a form of multi-party computation), which aggregates the weights received from all mobile devices into a new common model. Devices download this new model, and the cycle repeats, such that the model is continuously trained without collecting user data centrally.

SEPARATE

- Federated Learning/ Analytics (FL) (computation altering)
- Trusted Execution environments (data shielding)
- Secure Multi-Party Computation (SMPC)

Federated Analytics
https://www.nature.com/articles/s41586-020-2521-4

OpenSAFELY is a secure analytics platform developed in response to the COVID-19 pandemic, which enables researchers to conduct analysis across millions of patients' electronic health records (EHR). The platform works by leveraging federated analysis, where researchers' analytic code is uploaded to the datacenter where EHR data is kept. The code is executed in the datacenter, with the data kept in situ - data is never moved from where it was originally kept. Researchers are thus unable to download data, mitigating a key risk. The platform provides researchers with dummy data (NOT synthetic data) to develop their code. Once developed, the code must pass a series of automated sanity checks before it is packaged and deployed to the EHR provider's datacenter to execute the analysis. OpenSAFELY has enabled risk factors associated with COVID-19 to be identified, without exposing the personal information of individuals.

Trusted Execution Environments
https://signal.org/blog/private-contact-discovery/

Signal uses the Intel SGX TEE- to allow contact information from a user's phone to be used to find their contacts who are also on Signal. "A server-side contact discovery service runs inside the TEE, to which a user uploads their contact information, the service looks for matches in Signal's database of registered users, and information of these matches is returned to the user." Contact information is only decrypted inside the isolated TEE, meaning Signal has no visibility of it. Additionally, SGX supports remote attestation, meaning the client can verify that it is the expected contact discovery service code running inside the TEE before using it.

Trusted Execution Environments
https://github.com/microsoft/CCF/blob/main/CCF-TECHNICAL-REPORT.pdf

The Confidential Consortium Blockchain Framework (CCBF) is a system using trusted execution environments that facilitates confidentiality within a blockchain network. Within CCBF, confidentiality is provided by trusted execution environments (TEEs) that can process transactions that have been encrypted using keys accessible only to a CCBF node of a specific CCBF service. Besides confidentiality, TEEs also provide publicly verifiable artefacts, called quotes, that certify that the TEE is running a specific code. Hence, integrity of transaction evaluation in CCBF can be verified via quotes and not be replicated across mutually untrusted nodes as it is done in public blockchains. In addition, Microsoft's test showed that the CCBF could process 50,000+ transactions per second, demonstrating the scalability of the technology. The framework is not a standalone blockchain protocol, but rather it provides trusted foundations that can support any existing blockchain protocol.

Secure Multi-Party Computation (SMPC)
https://ai.facebook.com/blog/assessing-fairness-of-our-products-while-protecting-peoples-privacy/

In partnership with the privacy-focused technology company Oasis Labs, Meta developed an approach that draws on SMPC to allow analysis of encrypted data in aggregate to measure fairness. Data is securely distributed among multiple facilitators, who together can perform computations over the combined encrypted information without exposing their individual contributions of data to the other facilitators. ("Assessing fairness of our products while protecting people ... - Facebook") This allowed Meta to conduct a survey on Instagram in which they can voluntarily share their race or ethnicity.

SEPARATE	• Federated Learning/ Analytics (FL) (computation altering) • Trusted Execution environments (data shielding) • Secure Multi-Party Computation (SMPC)	**Secure Multi-Party Computation (SMPC)** https://academic.oup.com/comjnl/article/61/12/1749/5095655?login=false In 2011, the Estonian Association of Information Technology and Telecommunications (ITL) proposed collecting key financial metrics from its member companies to better understand the state of the telecoms sector. Members expressed concern over the confidentiality of the metrics as they would be sharing them with competitors. ITL worked with Cyberneticac on their Sharemind platform, to enable the analysis to be done whilst protecting confidentiality. 17 companies participated, uploading their metrics to the Sharemind platform, which distributed the data across three “computing parties” (CPs). These CPs performed the desired analysis using a multi-party computation protocol to ensure confidentiality. The results of the analysis were shared with the ITL who disseminated accordingly. The distributed nature of the computation meant no party, including the ITL, ever had direct access to another party’s metrics. **Secure Multi-Party Computation (SMPC)** https://thefintechtimes.com/vodafone-and-aws-trail-keyless-software-as-a-service-biometrics-to-improve-payment-experience/ Keyless is a cybersecurity platform that provides privacy-first passwordless authentication and personal identity management solutions for enterprises. They combine biometrics with PETs and a distributed cloud network. Their technology means that enterprises no longer need to centrally store and manage passwords, biometric data, and other personally identifiable information. The underlying technology that they use is secure multiparty computation which enables multiple cloud servers to jointly process identity authentication requests without disclosing any data between them. As such, companies can comply with authentication requirements and incur minimal data protection risk.
AGGREGATE/ ABSTRACT	• De-identification techniques (data altering)	**De-Identification techniques** https://www.bio-itworld.com/news/2019/04/24/de-identifying-genomic-data-with-hashing-technology E360 Genomics uses a form of secure computation (tokenization of variants, multi-party is desired, and cell-size rules on statistical outputs). This is being leveraged by Genomics England. Researchers looking at non-identified, patient-level data can do analysis against the tokens to see, e.g., that token ABC123 correlates with type 2 diabetes at a certain P value. But they won’t know the identity of the individuals with that token, or even what genetic variant the token represents. Only aggregated results get detokenized. No information about the variants gets lost in translation because metadata is tagged to the tokens.
HIDE	• Encryption techniques such as zero knowledge proofs (data shielding) • Homomorphic Encryption (data shielding)	**Encryption Techniques** https://www.ingwb.com/en/insights/distributed-ledger-technology/ing-launches-major-addition-to-blockchain-technology ING uses Zero Knowledge Proofs that allow customers to prove that their secret number lies in a known range. For example, a mortgage applicant can prove that their income is in the admissible range without revealing their exact salary. **Homomorphic encryption** https://www.pnas.org/doi/10.1073/pnas.1918257117 Duality Technologies was contracted by the Defense Advanced Research Projects Agency (DARPA) to explore how homomorphic encryption (HE) could be applied to machine learning analysis that investigates potential genetic susceptibilities to severe COVID-19 symptoms. Duality also demonstrated how HE can facilitate genomic research on encrypted genetic data, presenting a privacy-preserving framework based on several advances in homomorphic encryption to demonstrate that it can perform an accurate Genome-Wide Association Studies (GWAS) analysis for a dataset of more than 25,000 individuals.

When you have identified possible privacy related risks, mapped to privacy requirements (See Section 3.1) across the data lifecycle and selected relevant privacy by design strategies and patterns (See Section 3.2), you will be able to select relevant Privacy Enhancing Technologies (PETs).

In this section we will:

1. **describe different applications and goals of PETs,**
2. **provide examples to associate privacy by design strategies to concrete PETs, and**
3. **highlight some points to take into consideration when selecting and applying PETs.**

3.3.1 Description and goals of PETs

In addition to the broad aims, (see Section 1), that PETs help achieve, different PETs can further be used to achieve distinct technical aims, such as:

- Securely providing access to private datasets
- Enabling joint analysis on private data held by several organizations
- Securely out-sourcing to the cloud computations on private data
- De-centralizing services or systems that rely on user data.
- Processing data without having visibility into a single user or data subject

To achieve these ends, the following (non-exhaustive) list of PETs can be used:

- Synthetic data
- Differential privacy
- Homomorphic encryption
- Secure Multi-Party Computation
- Federated analytics/federated learning
- Trusted execution environments
- Anonymization techniques
- Cryptographic techniques

3.3.2 Associating PETs to Privacy by Design strategies

Below we provide a table which associates the privacy by design strategy with relevant PETs described in this guidance. We will focus on the data-oriented privacy strategies (minimize, separate, aggregate, hide), for which PETs are particularly useful. Where possible we will also link the system component where the PET will most likely be applied (see Figure 9 below). Also note that for sake of brevity we have linked PETs directly to design strategies. To find the appropriate PET, it might be necessary to first select the proper privacy by design tactic and/or design pattern (see Section 3.2) before you are able to select a relevant PET.

3.3.2.1 Table 8: Minimize

PET	System component
De-identification techniques	Data source, data storage, data analysis, data access
Differential privacy	Data source, data transfer, data storage, data analysis
Pseudonymisation techniques	Data source, data storage, data analysis, data access
Synthetic data	Data source, data analysis, data access

3.3.2.2 Table 9: Aggregrate

PET	System component
De-identification techniques	Data storage, data analysis, data access

3.3.2.1 Table 10: Separate

PET	System component
Federated learning / analytics	Data analysis
Trusted Execution Environments	Data storage, data analysis
Secure Multi Party Communication	Data transfer, data analysis

3.3.2.4 Table 11: Hide

PET	System component
Encryption techniques	Data transfer
Homomorphic encryption	Data transfer, data storage, data analysis

Appendix I: Glossary

Accuracy- The data protection principle of processing personal data in a manner that ensures that the personal data is accurate, up to date, and that inaccurate data is erased, or rectified without delay.

Anonymization- the reduction of risk of identifiability to a sufficiently low level, taking into account both technical and non-technical measures that have been applied to the data.

Autonomy Harm- Harm whereby the use of personal data leads to restricting, undermining, inhibiting, or unduly influencing people's choices. It can be done through coercion, manipulation, thwarting expectations, because of chilling effects, or by withholding information or control.

Cryptographic Techniques- They are a type of PET that relies on techniques, such as encryption, to enable authentication (verifying the identity of a user or computer), confidentiality (keeping the contents of the data secret), and integrity (ensuring that data doesn't change between the time it leaves the source and the time it reaches its destination).

Data Controller- A natural or legal person, public authority, agency, or other body which, alone or jointly with others, determines the purposes and means of the processing of personal data.

Data Minimization- The data protection principle of processing personal data in a manner that is adequate, limited and to what is necessary in relation to the purpose for which the data is to be processed.

Data Processing- Any operation or set of operations which is performed on personal data or on sets of personal data, whether by automated means or not, such as collection, recording, organization, structuring, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, restriction, erasure, or destruction.

Data Processor- A natural or legal person, public authority, agency, or other body which processes personal data on behalf of another entity, typically the data controller.

Data Protection Authority (DPA)- The independent, public authority of each EU Member State, that supervises, through its investigative and corrective powers, the application of the data protection law.

Data Protection by Design and by Default- Appropriate technical and organizational measures, as mandated by the GDPR and LGDP, to be put in place to implement the data protection principles effectively, and to safeguard individual rights.

Data Protection Impact Assessment (DPIA)- It is a process undertaken by controllers, which is designed to identify risks arising out of the processing of personal data and to minimize these risks as far and as early as possible.

Data-Oriented Strategies- Design strategies that focus on the privacy-friendly processing of the data themselves. They are more technical in nature.

Data Sources- The playbook refers to data sources as all sources from which data is gathered, through the process of extraction, such as from third party databases, entered by a user in a webform, observed on a website, gathered by sensors etc. They can also be defined as specific data sets, metadata sets, database, or metadata repositories from where data or metadata are available.

Data Subject- A natural person, who is the subject of personal data processing, and consequently, can be identified, or is identifiable. The playbook also uses the terms users or individuals to refer to data subjects.

De-Identification Techniques- They are a form of PET that transforms or reduces the amount of information about the data subject, and/or the risk of an individual or entity being re-identified. They involve the direct manipulation of raw data, including techniques such as redaction, tokenization, hashing, generalization, and k-anonymity. (ENISA and Centre for Data Ethics and Innovation)

Design Patterns- They are common ways to approach a design problem. In software engineering a design pattern is defined as: “a scheme for refining the subsystems or components of a software system, or the relationships between them. It describes a commonly recurring structure of communicating components that solves a general design problem within a particular context.” Therefore, privacy patterns are design solutions to common privacy problems — a way to translate "privacy-by-design" into practical advice for software engineering.

Design Strategies- They describe a fundamental approach to achieving a certain design goal. They favor certain structural organizations or schemes over others. It has certain properties that allow it to be distinguished from other (fundamental) approaches that achieve the same goal and are also applicable during the development and analysis phase of the ICT system. A privacy design strategy is therefore a design strategy that achieves (some level of) privacy protection as its goal. (European Union Agency for Cybersecurity (ENISA))

Design Tactics- Tactics that can be deployed by a controller to achieve process oriented, or data-oriented design strategies. Process oriented tactics include informing the data subject about the processing activity, providing them control over it, enforcing the privacy policy in a privacy friendly manner, and demonstrating that it is being done in such a manner. Data oriented tactics include minimizing data processing, separating said processing, abstracting the amount of detail with which personal data is processed, and hiding personal data to protect them from being observed, or linked.

Differential Privacy- It is a type of a PET that, when applied to an algorithm or mechanism, guarantees that, on analyzing a dataset of several data subjects, the outcome of the analysis will not be affected and will remain the same, even if any data subject was not included in the dataset. It acts as a privacy standard, by preventing the output of any statistical analysis from revealing any personal data specific to a data subject, within a particular data set. An algorithm is typically made differentially private by adding noise to either the input data (local differential privacy) or to the output it produces (global differential privacy).

Economic Harm- Harm whereby the use of personal data leads to monetary losses or results in the loss in the value of something.

Encryption at rest- It is a form of encryption that secures data stored on a disk, by using an algorithm to convert legible data into ciphertext, to prevent unauthorized users from reading or attempting to access said data.

Encryption in transit- It is a form of encryption that secures data as it flows between two connected computers, by using an algorithm to convert legible data into ciphertext, to prevent bad actors from eavesdropping on the connection to learn about the contents of the transmission.

End-to-End Encryption (E2EE)- It is a method of encrypting data and keeping it always encrypted between two or more communicating parties. Only the parties involved in the communication have access to the decryption keys.

Federated Analytics- It is a type of PET that involves an edge processing method whereby a party executes a program on a device or server where the data is situated; and the processing of that data happens on that device or server, and the insights gleaned from it is communicated back to the party. In this way, no data is directly revealed to the party.

Federated Learning (FL)- A subset of Federated Analytics, FL is a distributed machine learning method which enables training on a large body of decentralized data residing on edge devices like mobile phones. A central server coordinates a network of nodes, each of which has training data. The nodes each train a local model, and it is that model which is shared with the central server. As a result of this, data inputs are protected at the device level.

GDPR- The General Data Protection Regulation is a regulation on data protection and privacy in the European Union (EU) and the European Economic Area (EEA).

Homomorphic Encryption- It is a type of PET that enables data processing to be outsourced to an untrusted third party. It is a form of encryption that allows certain computations on encrypted data to generate an encrypted result. This encrypted result, when decrypted, matches the result of the same operations performed on the data before encryption.

Information System- It can be seen conceptually as a system that ingests data, stores data, transforms / uses that data and makes it accessible to other systems or people.

Integrity and Confidentiality- The data protection principle of processing of personal data that ensures appropriate security of the personal data, including protection against unauthorized or unlawful processing and against accidental loss, destruction, or damage, using appropriate technical or organizational measures.

LGDP- Lei Geral de Proteção de Dados is Brazil’s data protection and privacy law.

Personal Data- Any information relating to a data subject, the processing of which, can identify said data subject, or make the data subject identifiable; an identifiable data subject is one who can be identified, directly or indirectly.

Privacy by Design- Design philosophy that demands that privacy requirements are addressed, through technical measures, from the start, when designing and developing a new system.

Privacy Enhancing Technologies (PETs)- Also referred to as privacy-enhancing, privacy-preserving, and privacy-protecting technologies, PETs involve advanced techniques drawn from the fields of cryptography and statistics. These techniques help minimize the data that's processed by or made visible to an organization while preserving critical functionality. They are software and hardware solutions, i.e., systems encompassing technical processes, methods, or knowledge to achieve specific privacy or data protection functionality or to protect against risks of privacy of an individual or a group of natural persons. Simply put, they are a "system of ICT measures protecting informational privacy by eliminating or minimizing personal data thereby preventing unnecessary or unwanted processing of personal data, without the loss of the functionality of the information system."

Privacy Preserving Machine Learning (PPML) - PETs that are applied to AI/ML systems.

Process Oriented Design Strategies- Design strategies that focus on the processes surrounding the responsible handling of personal data. They deal with the organizational aspects and the procedures that need to be in place.

Pseudonymization- The processing of personal data in such a manner that data subjects are no longer capable of being identified, or are identifiable from such data, without the use of additional information.

Psychological Harm- Harm whereby the use of personal data leads to negative mental responses by the victim, namely emotional distress, or disturbance. Emotional distress involves painful or unpleasant feelings. Disturbance involves disruption to tranquility and peace of mind.

Purpose Limitation- The data protection principle of processing personal data for purposes that are specific, explicit, and legitimate, and not further processed in a manner that is incompatible with those purposes.

Re-identification- It is any process that re-establishes the relationship between data and the subject to which the data refer.

Relationship Harm- Harm whereby the use of personal data results in damage to relationships that individuals have with one another. Breach of confidentiality in relationships, especially those that are fiduciary in nature, can lead to a loss of trust.

Reputational Harm- Harm whereby the use of personal data causes damage to an individual's reputation and standing in the community.

Risk- The likelihood of something bad happening.

Secure Multi-Party Computation (SMPC)- It is a type of PET that relies on cryptography to enable private distributed computations. SMPC protocols allow computation of an agreed-upon function among a set of parties, while keeping any participant from learning anything about the raw inputs provided by other parties.

Storage- The playbook views storage as the component of the information system that (temporarily) stores, or 'loads', the ingested and transformed data, to be used for analysis, and access and further use.

Storage Limitation- The data protection principle of processing of personal data in a manner whereby personal data is kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed.

Synthetic Data- It is a type of a PET that involves the generation of artificial data by the processing of original data, and by relying on a model. This process is called synthesis. It realistically mimics both personal and non-personal data but does not actually refer to any specific identified or identifiable individual, in the case of personal data, or to the real measure of an observable parameter, in the case of non-personal data.

Third Parties- They refer to natural or legal persons, public authorities, agencies, or bodies other than the data subject, controller, processor, and persons who, under the direct authority of the controller or processor, are authorized to process personal data.

Transfer- The playbook refers to transfer as the component of the information system whereby the data that is ingested from various sources undergoes processing and is transformed and consolidated for its intended analytical use case.

Trusted Execution Environment (TEE)- It is a type of a PET where secure computation is guaranteed through a combination of special-purpose hardware and software, thereby enabling a tamper-resistant, processing environment that is isolated from a computer's main processor and memory. Code or data held within this environment cannot be accessed from the main processor, and communications between the main processor and the TEE are encrypted.

Zero-Knowledge Proof- It is a type of a cryptographic technique which can be used to enforce the confidentiality and the data minimization principles of the GDPR. The core idea is to allow a data subject to prove to a server (data controller) that they know a secret information without revealing anything on this secret.

Appendix II: Commonly used PETs

1.1 Differential privacy (DP)

DP is a privacy standard, rather than a single tool or algorithm. For an algorithm or "mechanism" to be considered differentially private, outputs of the mechanism must be approximately indistinguishable were one to change any person's data in the database—e.g., add or remove it, or change a record's data values. DP was designed to avoid pitfalls that previous attempts to define privacy suffered, especially in the context of multiple releases and when adversaries have access to side knowledge. Unlike software and hardware encryption-based PETs which address privacy during computation, differential privacy addresses privacy in disclosure.

1.1.1 How it works

Differential privacy can be achieved or amplified through a range of mechanisms, such as removing or randomizing data points and injection of carefully calibrated noise.

Global Differential Privacy

The goal under global DP is to release a statistical summary of information when a trusted party has access to the full collection of sensitive data. In these cases, the data releasing party can inject noise that is proportionate to the characteristics of the full sample. Global DP is useful in settings where the goal is the privatization of a sample statistic (e.g., an average, count, sample variance) or model estimate (e.g., 'lift', regression coefficients), and the aim is to avoid the leakage of a particular data subject.

Local Differential Privacy (LDP)

LDP relates to privacy guarantees over individual records in a database. Whereas global mechanisms produce a statistical summary of sensitive data, LDP gives protections over record-level data releases. In many cases, LDP can be used in absence of any central trusted aggregator, such as third parties. Due to privatizing individual records instead of statistical summaries, however, the noise distributions used for LDP tend to differ from central models and require unique statistical frameworks to utilize their output.

1.1.2 Concerns or limitations

It is important to assess whether the desired level of epsilon (a measure of privacy level attained by DP) is attainable with acceptable signal-to-noise ratio. For example, privacy advocates and regulators might demand very low epsilon values (high privacy guarantees) for certain use cases, but that might significantly impact utility of downstream data use in analyses and ML models. In contrast, lower privacy guarantees or bad configuration may result in the possible leakage of personal data or re-identification of an individual. There is no agreed upon standard for what is an acceptable level of differential privacy, or where to set the "privacy budget." Parties must meet legal obligations as a minimum standard.

1.1.3 Tools for engineers

Statistical DP

- OpenDP
- Google’s Differential Privacy

DP for Machine Learning

- Pytorch: Opacus
- Tensorflow: TensorFlow Privacy
- JAX: JAX-Privacy

1.1.4 Educational resources

Blogposts and videos

- minutephysics: quick primer on differential privacy
- OpenMined blog series: Privacy-preserving data science explained, Use cases of differential privacy, full series
- PyTorch Developer Day: Opacus. 2022, 2021

Quick tutorials

- OpenMined tutorials: opacus, PyDP, PipelineDP,
- Tutorials by tool: opacus, TensorFlowPrivacy

In-depth classes

- Secure and Private AI: A gentle introduction into Differential Privacy (along with Federated Learning and Encrypted Computation)
- Privacy in Statistics and Machine Learning, taught by Adam Smith and Jonathan Ullman
- Gautam Kamath - A Course in Differential Privacy

1.1.4 Educational resources

- Awesome Differential Privacy by menisadi
- Awesome Differential Privacy by Billy1900
- Recommended by DifferentialPrivacy.org

Date
+
Noise introduced
=
PRIVATE RELEASE

Figure 11: Simple visualization of differential privacy

1.2 Secure Multi-Party Computation

Secure multi-party computation (SMPC) is a subfield of cryptography that enables private distributed computations. SMPC protocols allow computation of an agreed-upon function among a set of parties, while keeping any participant from learning anything about the raw inputs provided by other parties.

1.2.1 How it works

SMPC relies on dividing each data input item into two or more shares and distributing these to compute parties. To perform the shared computation required for MPC, all participating compute parties follow a protocol: a set of instructions and intercommunications that when followed by those parties implements a distributed computer program. MPC protocols often use secret sharing based methods. The computations can be simple summary statistics or extend to a complex sequence of calculations like performing ML training.

1.2.2 Concerns or limitations

Most existing MPC protocols take significantly more time than centralized systems to compute a given function (slowdown factors of 100x to 100,000x), due in part to delays incurred in communicating encrypted data across the network (latency). Implementation complexity is also a limiting factor since MPC requires all participating entities to play arole in execution of the computation. Computational complexity grows with the number of parties due to high communication cost, which likely limits applications to use cases with just 2 parties

Furthermore, the reconstruction of input data may be possible or incorrect calculation can result due to corruption of an SMPC protocol.

1.2.3 Tools for engineers

- For PyTorch: CrypTen
- For TensorFlow: TF Encrypted

1.2.4 Educational resources

- Online course by OpenMined
- Simple tutorial with TinySMPC

1.2.5 Resource collections

- Awesome MPC by rdragos
- Awesome Secure Computation by Jamie-Cui

1.3 Federated analytics and federated learning (FA/FL)

Federated analytics is an edge processing method which involves a party executing a program on a device or server where data is situated, processing that data on that device or server, and communicating insights back to the party.

Federated Learning (FL) is a subset of Federated Analytics. It is a distributed machine learning method which enables training on a large body of decentralized data residing on edge devices like mobile phones. A central server coordinates a network of nodes, each of which has training data. The nodes each train a local model, and it is that model which is shared with the central server. As a result of this, data inputs are protected at the device level.

1.3.1 How it works

The idea is to rely on on-device processing to train models across many devices without sending the raw training data to a central server. This can be combined with differential privacy for end-to-end privacy protection. It is also feasible to generalize its application to data servers besides user mobile devices.

1.3.2 Concerns or limitations

Although federated analytics provides a central party with no direct access to the data, it is possible that information about individual data subjects can be inferred from the output of the analysis. Specifically, it is possible that data leakage may arise from the preservation of characteristics and correlations of personal data from data and training samples that can be extracted or inferred by bad actors. Applying local or central differential privacy can be a way to mitigate this risk. Furthermore, not having direct access to data can make it more difficult to develop, test, and troubleshoot code.

As to FL, training in heterogeneous and potentially massive networks introduces novel challenges that require a fundamental departure from standard approaches for large-scale machine learning, distributed optimization, and privacy-preserving data analysis. For instance, communication is a critical bottleneck in federated networks. To fit a model to data generated by the devices in the federated network, it is therefore necessary to develop communication-efficient methods that iteratively send small messages or model updates as part of the training process, as opposed to sending the entire dataset over the network. Additionally, the storage, computational, and communication capabilities of each device in federated networks may differ due to variability in hardware (CPU, memory), network connectivity (3G, 4G, 5G, WIFI), and power (battery level).

1.3.3 Tools for engineers

1. **Research oriented**

    1.1 PyTorch: PySyft, FLSim
    1.2 Tensorflow: TensorFlow Federated
    1.3 JAX: FedJAX

2. **Enterprise grade**

    2.1 OpenFL by Intel
    2.2 NVFlare by NVIDIA
    2.3 IBM Federated Learning

1.3.4 Educational resources

Blogposts and videos

- OpenMined blog series: What is Federated Learning?, Privacy-preserving data science explained, full series
- Google AI blogpost
- What is Federated Learning by NVIDIA

Quick tutorials

- OpenMined tutorials: short intro to PySyft, longer intro to PySyft
- Federated Learning for Image Classification with TensorFlow

In-depth classes

- Secure and Private AI: A gentle introduction into Federated Learning (along with Differential Privacy and Encrypted Computation)
- Online course by OpenMined

1.3.5 Resource collections

- Awesome Federated Learning by chaoyanghe
- Awesome Federated Learning by innovation-cat

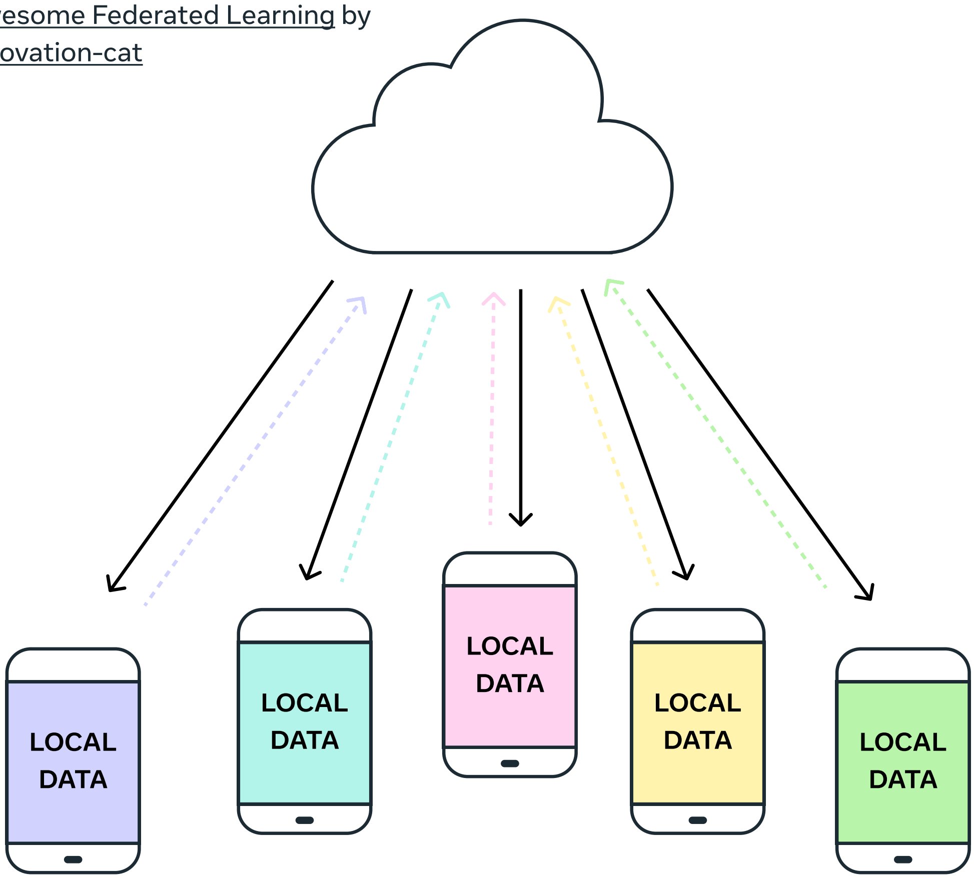


Figure 12: Simple diagram

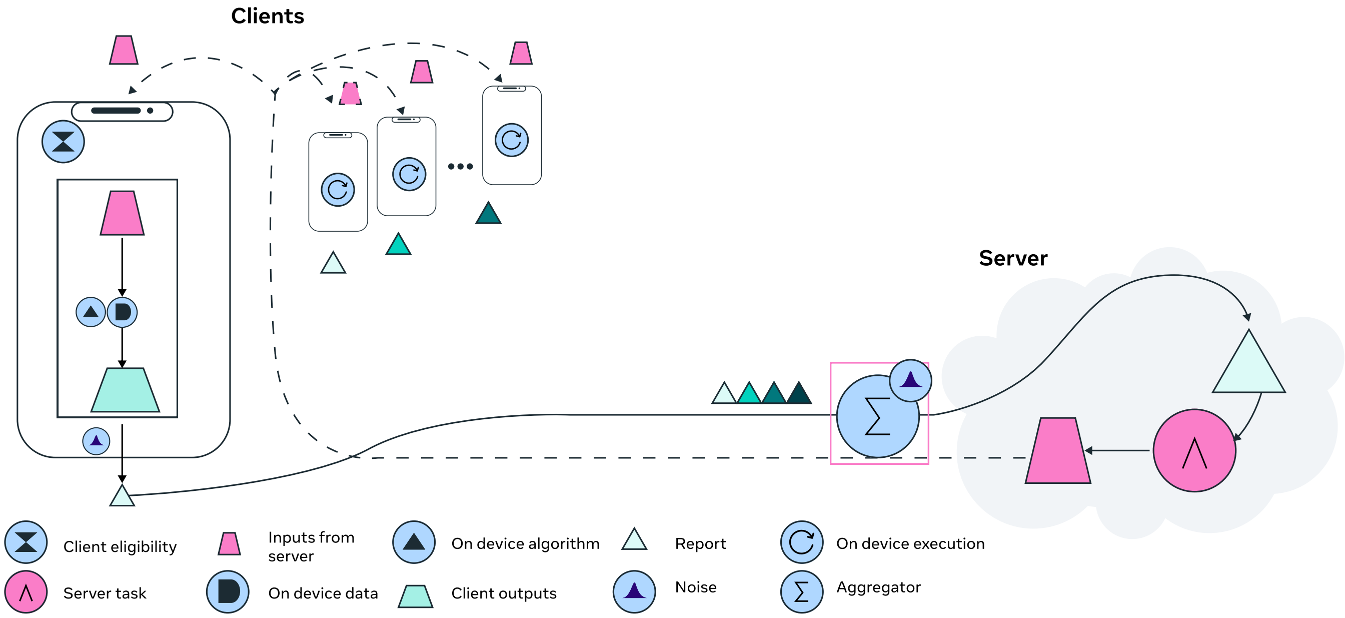


Figure 13: Complex diagram with deeper visibility into data processing

1.4 Trusted execution environments (TEEs)

Trusted Execution Environments (TEEs) provide secure computation through a combination of special-purpose hardware in modern processors and software built to use those hardware features. This special-purpose hardware provides a mechanism by which a process can run on a processor without its memory or execution state being visible to any other process on the processor, including the operating system or other privileged code. Like HE, TEEs can be used to securely outsource computations on sensitive data to the cloud. Instead of a cryptographic solution, TEEs offer a hardware-based way to ensure data and code cannot be learnt by a server to which computation is outsourced.

1.4.1 How it works

Because TEE is built using special purpose hardware, some companies have their own TEE solutions. Intel Software Guard Extension (SGX) is the most commonly available TEE that is built into CPUs. SGX involves encryption by the CPU of a portion of memory. The enclave is decrypted only within the CPU itself, and even then, only for code and data running from within the enclave itself. The actual computation is performed on the unencrypted data within the enclave. ARM's Trustzone and AMD's Platform Security Processor also offer TEE capability.

1.4.2 Concerns or limitations

TEE-based solutions require trusting hardware manufacturers and other service providers, but a number of recently exposed side-channel attacks on Intel SGX has reduced confidence in using TEEs for preserving input privacy. Furthermore, in terms of compute, enclave computation is usually comparable in speed to computing "in the clear". However, the secure memory size limitation in TEEs is also a challenge, so that only limited data can be processed at any one time.

Risks around TEE also include:

1. Side channel attacks - an attack based on extra information that can be gathered from the way the TEE communicates with other parts of a computer.

2. Timing attacks - attacks that can leak cryptographic keys or infer information about the underlying operation of the TEE.

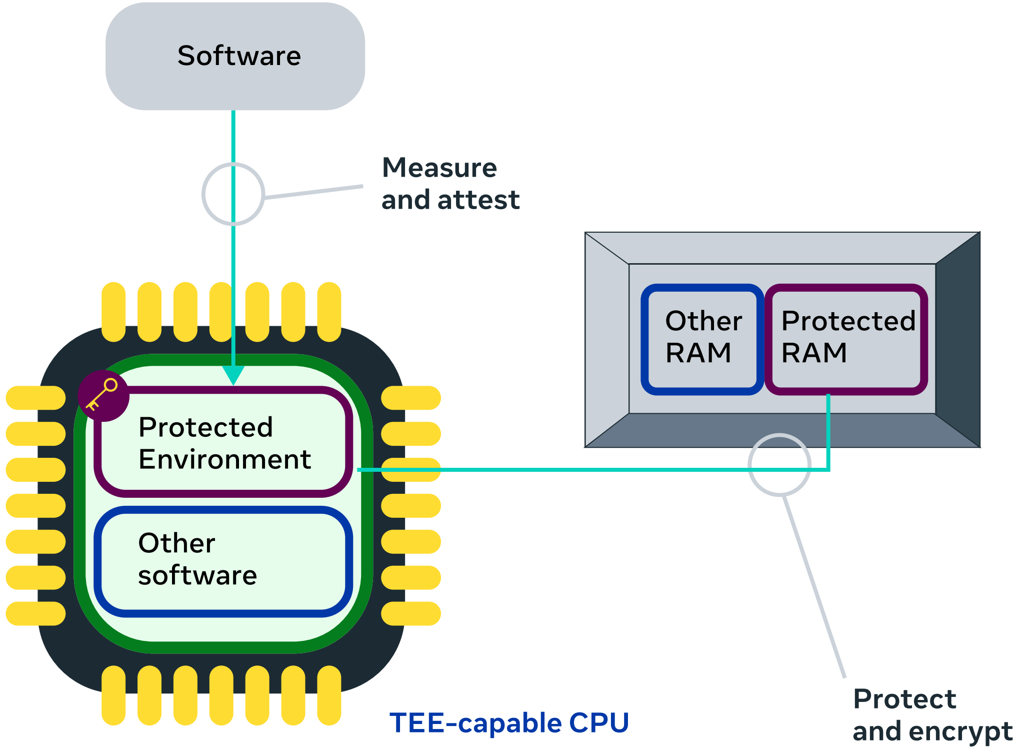


Figure 14: Simple diagram of trusted execution environments

1.5 De-identification techniques[76]

A de-identification technique is any data transformation or alteration that reduces the amount of information about the data subject, and/or reduces the risk that an individual or entity can be re-identified. De-identification techniques involve direct manipulation of raw data.

1.5.1 How it works

By way of example, here are some De-identification techniques:

Redaction

Deleting an entire record or field, or obfuscating part of a record or field (e.g., deleting all but the last 4 digits of a credit card number or government identifier.)

Tokenization

Replacing a real value with a randomly generated value

Hashing

Using an algorithm (such as SHA or MD5) to applying a function to a value to produce a fixed-length value (or hash)

Generalization

Transforming a value to a less precise value, e.g., replacing a height of 156 cm with a range 150-160 cm

K-anonymity

A method of releasing person-specific data while, at the same time, safeguarding the privacy of the individuals to whom the data refer by applying a combination of de-identification techniques so that any record in the dataset becomes indistinguishable from (k-1) records.

1.5.2 Concerns or limitations

Data cannot be absolutely anonymized—reducing the risk of identifying an individual to zero. Many legal obligations are pinned to this binary. In practice, though, achieving absolute anonymization is often difficult, technically complex, and resource intensive.

1.6 Cryptographic techniques

Generally speaking, a cryptographic technique concerns three basic purposes:

- **Authentication:** Verifying the identity of a user or computer
- **Confidentiality:** Keeping the contents of the data secret
- **Integrity:** Ensuring that data doesn't change between the time it leaves the source and the time it reaches its destination.

1.6.1 How it works

A core cryptographic technique is encryption, one of the principal security technologies used to protect information. Encryption converts legible data into ciphertext - a representation of the data that is not readable by a human or a computer. For the data to be read, it must be first decrypted, which requires access to an appropriate decryption key. Thus, data is kept secret from entities that do not have access to this key.

Confidentiality

Encryption in transit

Deleting an entire record or field, or Encryption in transit secures data as it flows between two connected computers. For example, when you login to a website your username and password are first encrypted before being sent over the internet to the website in question, ensuring that a bad actor eavesdropping on your connection is not able to learn your credentials. Common examples of encryption in transit are SSL and TLS.

Encryption at rest

At rest encryption secures data for storage on disk. For example, you may well be using a computer that stores data on an encrypted hard drive. This ensures that data stored on your computer cannot be read by an unauthorized user attempting to access your computer.81

End to end encryption

End to end encryption, or e2ee, means that data is encrypted while in transit from an original sender to the intended final recipient.. For example, many messaging services only rely on encryption in transit and encryption at rest. In contrast, e2ee prevents unintended users, including third parties, reading, or modifying data when only the intended readers should have this access and ability.

In transit and at rest encryption are mature technologies and commonplace. They should be considered standard practice when sending information over the internet, and when storing sensitive information on disk. E2ee is less common.

1.6.2 Concerns or limitations

As a legal and policy matter, many governmental entities object to using e2ee, as it can make it difficult to respond to government legal requests. Companies should examine their use case and consult with their counsel before applying e2ee to particular products and services.

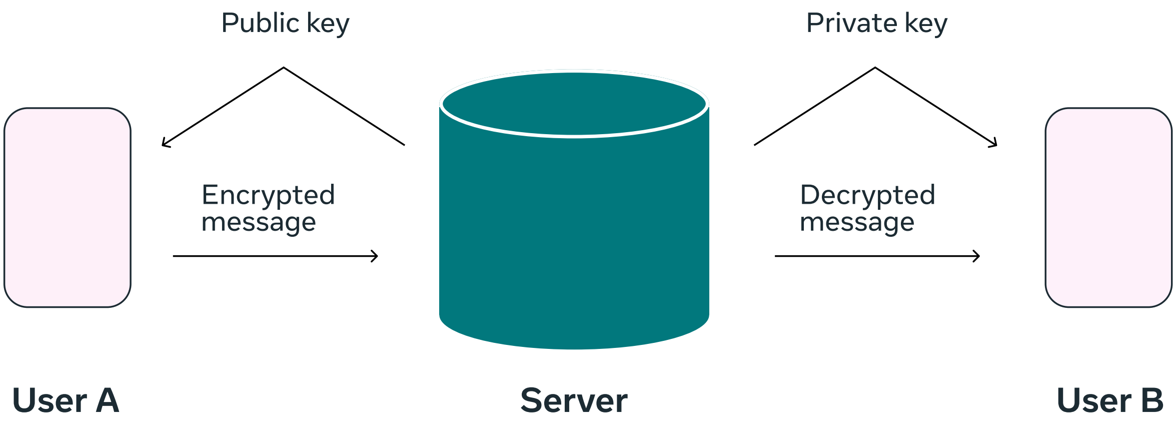


Figure 15: e2ee encryption visualized

1.7 Homomorphic encryption (HE)

Homomorphic Encryption (HE) is a form of encryption that allows certain computations on encrypted data, generating an encrypted result. This encrypted result, when decrypted, matches the result of the same operations performed on the data before encryption. HE doesn’t require distributed computing and can be used in circumstances where the client wants to maintain their input data privacy but does not have the resources to handle the secure computation and wants to “outsource” the computation to a server.

1.7.1 How it works

Homomorphic encryption, depicted in the context of a client-server model. The client sends encrypted data to a server, where a specific analysis is performed on the encrypted data, without decrypting that data. The encrypted result is then sent to the client, who can decrypt it to obtain the result of the analysis they wished to outsource.

1.7.2 Concerns or limitations

Compared with computing on unencrypted data, homomorphic encryption is computationally expensive and has lower throughput. Encryption can entail a substantial increase in data size, which can cause a major bandwidth problem. Also, computations need to be represented as polynomials, which can be a limitation in practice. In the case of Fully Homomorphic Encryption (FHE), the running time increases dramatically with the number of operations (additions or multiplications). These concerns are the subject of ongoing research.

1.7.3 Tools for engineers

- Microsoft SEAL - C++ FHE library
- Fully Homomorphic Encryption by Google

1.7.4 Educational resources

- Online course by OpenMined

1.7.5 Resource collections

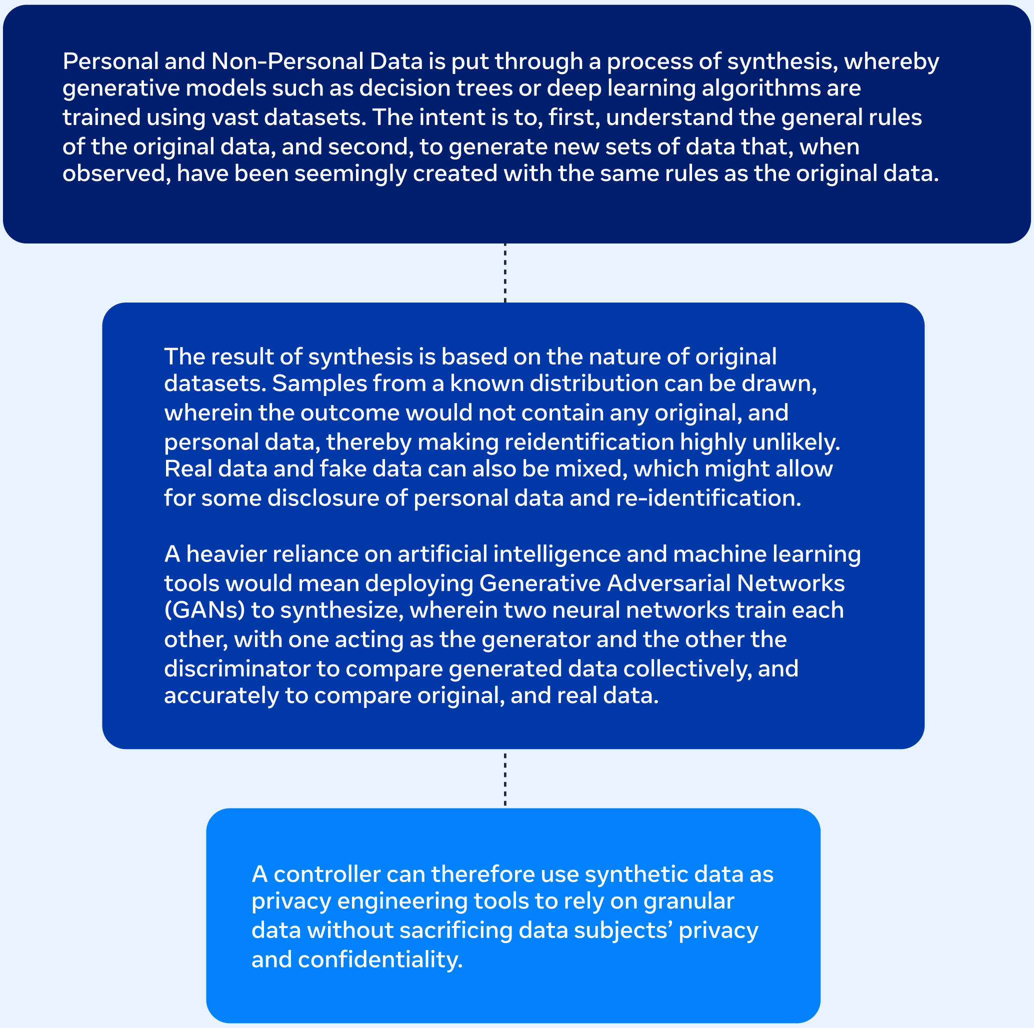
- Awesome HE by jonaschn

1.8 Synthetic Data

Synthetic data is artificial data that is generated by the processing of original data, and by using a model. It realistically mimics both personal and non-personal data but does not actually refer to any specific identified or identifiable individual, in the case of personal data, or to the real measure of an observable parameter, in the case of non-personal data. However, the generated data exhibits similar statistical properties, which means that when both the original and the synthetic data is put through statistical analysis, they should deliver very similar results. The degree of accuracy of the generated proxy, is dependent on the extent of simulation desired by the actors and is a measure of the utility of the method and the model deployed. For instance, synthetic data can be entirely simulated to test software applications, or simply manipulated to restrict re-identification of data subjects. Synthetic data can also be a cross-pollination of data from different data sources.

1.8.1 How it works

Homomorphic encryption, depicted in the context of a client-server model. The client sends encrypted data to a server, where a specific analysis is performed on the encrypted data, without decrypting that data. The encrypted result is then sent to the client, who can decrypt it to obtain the result of the analysis they wished to outsource.



Personal and Non-Personal Data is put through a process of synthesis, whereby generative models such as decision trees or deep learning algorithms are trained using vast datasets. The intent is to, first, understand the general rules of the original data, and second, to generate new sets of data that, when observed, have been seemingly created with the same rules as the original data.

The result of synthesis is based on the nature of original datasets. Samples from a known distribution can be drawn, wherein the outcome would not contain any original, and personal data, thereby making reidentification highly unlikely. Real data and fake data can also be mixed, which might allow for some disclosure of personal data and re-identification.

A heavier reliance on artificial intelligence and machine learning tools would mean deploying Generative Adversarial Networks (GANs) to synthesize, wherein two neural networks train each other, with one acting as the generator and the other the discriminator to compare generated data collectively, and accurately to compare original, and real data.

A controller can therefore use synthetic data as privacy engineering tools to rely on granular data without sacrificing data subjects' privacy and confidentiality.

1.8.2 Concerns or limitations

- The quality of the synthetic data is highly correlated with that of the original datasets, and therefore would be susceptible to biases if the same have infiltrated the original datasets.
- Ensuring that the synthetic output is accurate is difficult, especially when the datasets and the processing operations are complex. Any manipulation of the original datasets to create fair synthetic datasets would also make the resulting data inaccurate.
- Since synthetic datasets only mimic real data, they cannot be considered true measures as they only replicate certain properties of a phenomenon. Therefore, synthetic data may not cover some outliers that original data has, which might be more important than original data points.

RISK OF RE-IDENTIFICATION

The more the synthetic data imitate the real ones, the more useful they are, increasing the risks of revealing the personal data of individuals.

1.8.3 Tools for engineers

- MostlyAI
- MDClone

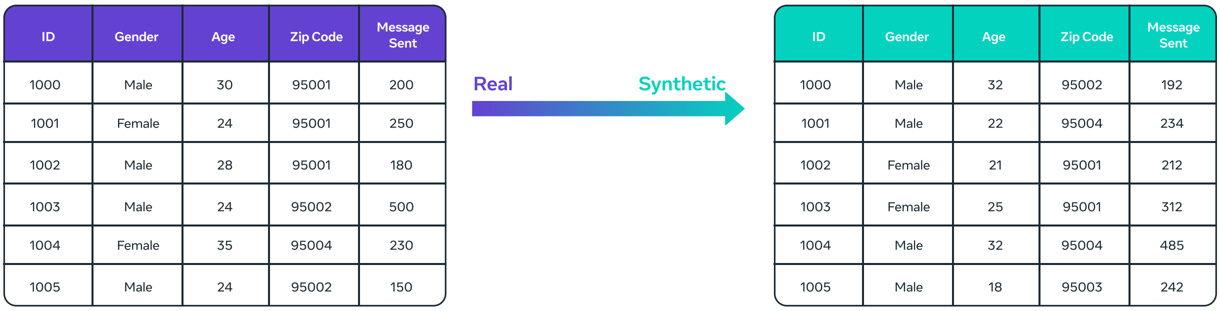
1.8.4 Educational resources

- MostlyAI's Guide to Synthetic Data
- Synthetic Data: The Complete Guide by Datagen

1.8.5 Resource collections

- UK Data Service Open- https://github.com/UKDataServiceOpen/Synthetic-Data
- GitHub-GretelAI

Below, note that synthetic data looks similar to randomly generated test data but looks and feels like real data with realistic values. It is generated by sample-based method to mimic and carry through all the statistical properties, patterns, and correlations in the source data (i.e., statistically equivalent)



ID	Gender	Age	Zip Code	Message Sent
1000	Male	30	95001	200
1001	Female	24	95001	250
1002	Male	28	95001	180
1003	Male	24	95002	500
1004	Female	35	95004	230
1005	Male	24	95002	150

ID	Gender	Age	Zip Code	Message Sent
1000	Male	32	95002	192
1001	Male	22	95004	234
1002	Female	21	95001	212
1003	Female	25	95001	312
1004	Male	32	95004	485
1005	Male	18	95003	242

Figure 16: Synthetic data, visualized

Appendix III: PETs’ technical considerations

1. Technical considerations when selecting and applying PETs

This section describes some considerations that organizations may consider when selecting and applying PETs to their data processing activities, such as for which type of dataset these are more suitable, what are their level of maturity, and trade-offs between utility of data and the relative privacy of data, among others.

Depending on your organization and your use case, the optimal PET should be decided on a case-by-case basis. This is especially true in the context of applying a PET in the development or deployment of an AI system.

2. Key conditions for applying PETs

As a company considers privacy harms, it should also consider the necessary setup conditions for applying different PETs. We set those out below.

🟢 **Generic,** applicable to the most client/ server applications	🔴 **Niche,** only applicable for a specific setup, which are not that common	🟣 **Unique,** see comments

- 🟢 **Cryptographic techniques -** Varies based on the technology but may work for most client -> server data sharing setup, where appropriate. As a general matter, encryption is better applicable for protecting data that is stored on the server side.
- 🟢 **Deidentification techniques -** may work for any dataset where determined appropriate
- 🟢 **Differential privacy:**
  - **Local differential privacy -** may work for any dataset where determined appropriate
  - **Central differential privacy -** may work for any data aggregation (including ML training) where determined appropriate
- 🟢 **Federated Learning / Federated Analytics -** works for any client -> server data sharing setup, where determined appropriate
- 🟢 **Synthetic data-** works for any dataset
- 🟢 **Federated Learning / Federated Analytics -** works for any client -> server data sharing setup, where determined appropriate
- 🔴 **Homomorphic encryption -** may work if computing resources are owned by an untrusted party
- 🔴 **Secure Multi-Party Computation -** works if private data is split between multiple parties. Could be either client+server or server+3rd party.
- 🟣 **Trusted Execution Environments -** Does not affect privacy by itself. Removes the need to trust the data operator (company) for any of the technologies above.

3. Consider PETs maturity

PETs can also be categorized based on their maturity level. Maturity can generally be described as the phase of development and the level of research that is invested in a particular PET.

PET	Category	Maturity
Cryptographic technologies (e.g., encrypting data at rest or in transit using standard techniques, e2ee, zero-knowledge proofs)	Data-shielding PETs	Mature/improving, depending on the PET (e2ee is less mature than other techniques)
Trusted execution environments	Data-shielding PETs	New
Homomorphic encryption	Data-shielding PETs	Improving
Deidentification techniques (e.g., k-anonymity, tokenization, generalization)	Data-altering PETs	Mature
Differential privacy	Data-altering PETs	Mature
Secure multiparty computation	Computation-altering PETs	Improving
Federated learning/ federated analytics	Computation-altering PETs	Improving
Synthetic data	Data-altering PETs	Improving

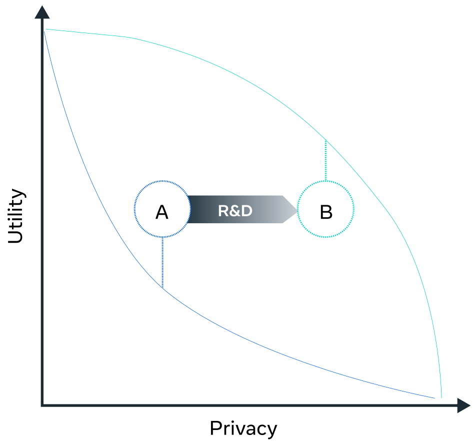
Table 12: PET summary with relevant maturity level

Relative maturity of a PET is a key factor for decision makers to consider as they weigh PET applicability. Some PETs depend on certain client-server setups, specialized hardware, specialized software, and a company's ability to scale the technology to provide a tangible benefit to stakeholders. Other PETs may help mitigate specific threat models and privacy and consumer harms. Companies that are interested in experimenting with PETs may wish to consider more mature PETs where many resources exist about implementation that do not require unique technical setups.

4. Consider privacy trade-offs

A core consideration for implementing risk reducing measures is considering the tradeoff between the utility of data and the relative privacy impact to an individual. Data that has been protected or altered by multiple PETs may have little utility, and maximal utility of data may have greater impact on a data subject.

The application of any PET to address a core privacy harm will implicate some privacy vs. utility tradeoff. It is key to acknowledge this type of tradeoff when considering a company's overall privacy risk and when choosing a particular PET.



Navigating this tradeoff requires PETs adopters to have a clear idea of what information or value they are trying to protect, and they need to determine the potential benefits and costs of different PETs. In some cases, as illustrated in the figure above, further research and development (R&D) could make this tradeoff less severe; technical advances could permit marginal increases in privacy resulting from the use of a PET to carry lower marginal costs to utility.

Technical tradeoffs may also exist between applying PETs and making progress toward equity or fairness. PETs may help enable measurements along sensitive characteristics to inform fairness work, but applying PETs can also make analyses of data less accurate for smaller, historically marginalized communities represented in it. Applying PETs may also be in tension with developing certain forms of fairness-aware machine learning that require having access to sensitive information about individuals to more actively correct for certain biases. That said, there may be instances in which PETs can enable privacy-protective insights into fairness and equity.

The exact contours of limitations and tradeoffs will be highly dependent on the specific use cases and datasets to which PETs are applied.

The explanations below are not intended to be exhaustive or complete and are provided for informational purposes only. We have also shortlisted some of the resources we believe to be a good starting point for engineers. Above in Appendix II, we set out details on a number of different PETs and describe how they function and key concerns and limitations. We also set out resources for engineers and other personnel to learn more about each type of PETs and provide diagrams. Privacy practitioners may consider the details set out here as they weigh the adoption of a particular PET for their business use cases alongside their legal and compliance strategies.

Appendix IV: Core technical use cases for engineers to consider with PETs mitigations

When engineers are deciding which PETs are best for which product use case, they may want to consider particular consumer & data subject concerns, which include, but are not limited to, the following:

“I, as a consumer/data subject, don’t want a company or bad actor to...

- “Access my data without my authorization”
- “Identify me in a dataset”
- “Make a sensitive inference about me”
- “Use my data for a different or incompatible purpose I’m not aware of”
- “Have real-time visibility into my actions”
- “Collect data in a centralized place about me”

As a data processing matter for engineers, many of these above arise from one core concern:
- whether a party the consumer may not be aware gains access to the relevant data.

Accessing data without a data subject’s authorization, whether resulting from an internal actor or external bad actor, could lead to identifying that subject in a dataset.

Collecting data in a centralized place could be concerning where someone (rogue employee or a hacker) could gain unauthorized access to the data and identify a person in the dataset.

Identifying a person in the dataset could be concerning where an entity can make a sensitive inference about a person based on their presence in the dataset and take an action based on that.

Similarly, having visibility into a data subject’s real time actions could potentially yield sensitive inferences, which sometimes may be concerning to data subjects

Using data for a different or incompatible purpose can lead to an entity accessing data without a person’s permission, identifying a person in a dataset, or making a sensitive inference.

In sum, with consideration to the above privacy concerns, to protect privacy, engineers should strive to reduce the risk that data is used improperly or exposed to entities that the people do not expect to have access"

Therefore, the core entities that data may be exposed to that could result in a privacy concern to consumers
are the following:

- The company itself
- 3rd parties
    - Commercial partners
    - Vendors (service providers/processors)
    - External entities
- Adversaries in a data leak scenario
- Adversaries’ communication intercept scenario)

Furthermore, in the context of the development of AI models, engineers should consider two privacy concerns:

ⓘ Exfiltration of data from an AI/ML model (whether that data is about an individual user, or data that might yield sensitive insights about a population)

ⓘ Exposing data to a central server (how much should data be visible to or accessed from a central server in the process of creating or training an AI system)

In light of the entities and the guidelines we described above, we outline core common technical use cases with associated threat models and how some PETs can mitigate for those threat models. Specifically, we focus on simple analytics (non-ML), cloud data analysis, simple ML, and joint data analysis.

1. Simple analytics

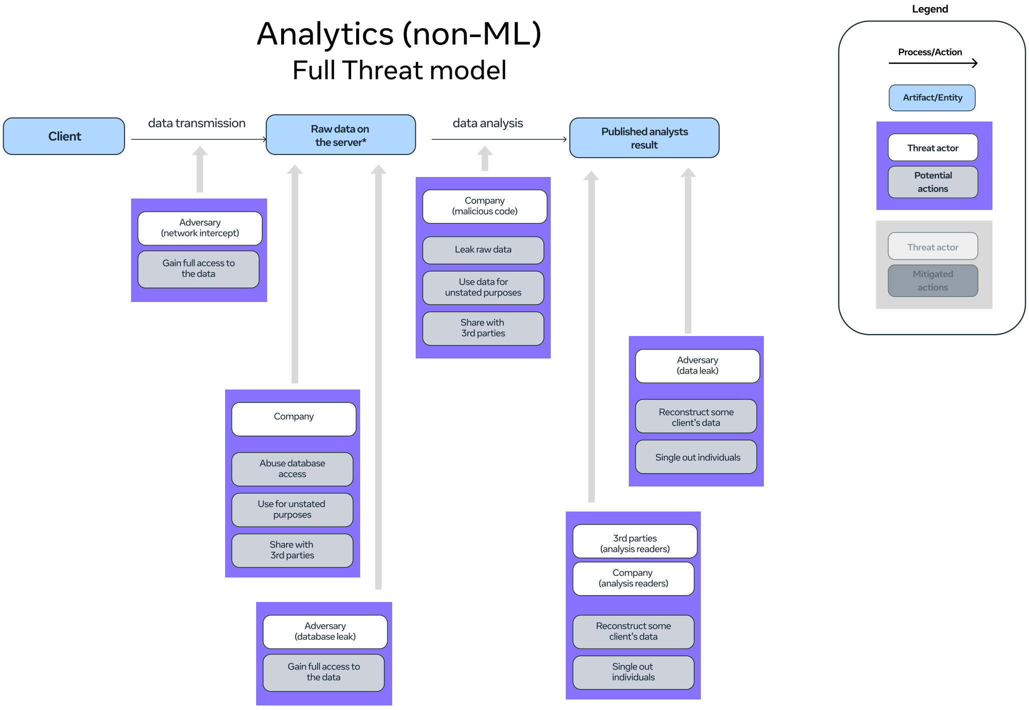


Figure 11: Non-ML analytics full threat model

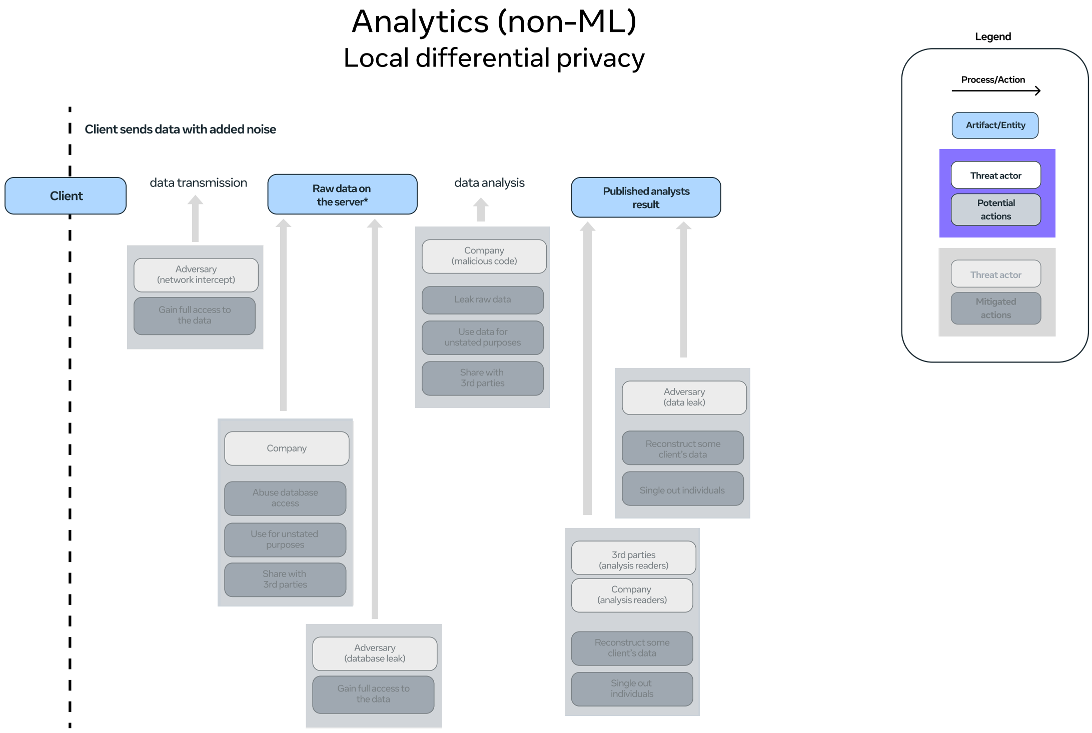
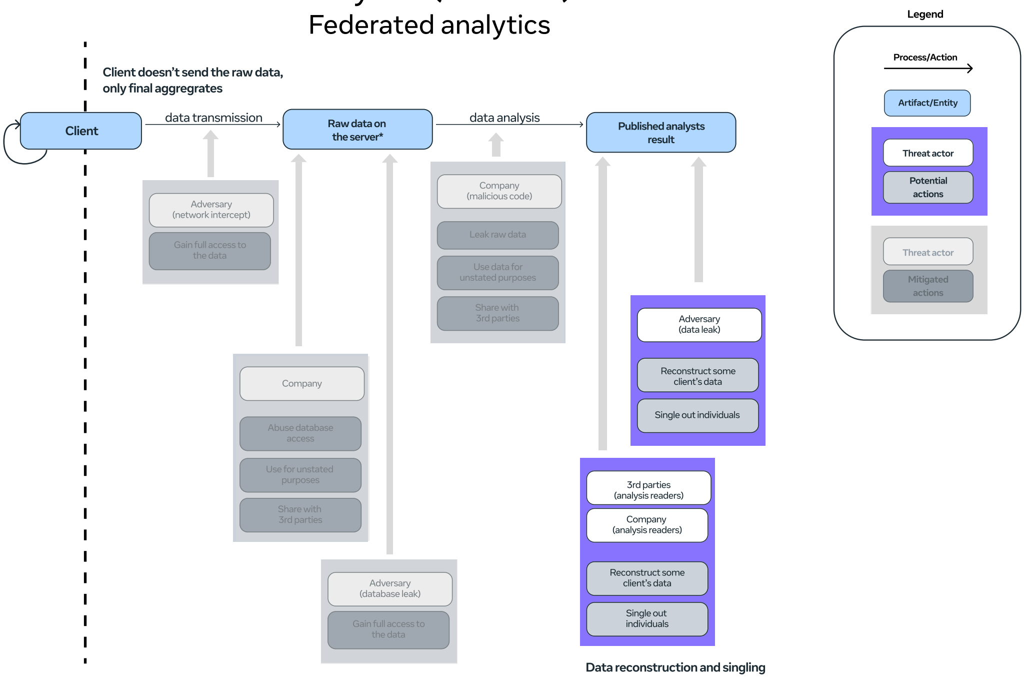

Analytics (non-ML)
Local differential privacy
Client sends data with added noise
Client
data transmission
Raw data on the server*
data analysis
Published analysts result
Adversary (network intercept)
Gain full access to the data
Company (malicious code)
Leak raw data
Use data for unstated purposes
Share with 3rd parties
Adversary (data leak)
Reconstruct some client's data
Single out individuals
Company
Abuse database access
Use for unstated purposes
Share with 3rd parties
3rd parties (analysis readers)
Company (analysis readers)
Reconstruct some client's data
Single out individuals
Adversary (database leak)
Gain full access to the data
Legend
Process/Action
Artifact/Entity
Threat actor
Potential actions
Threat actor
Mitigated actions

Figure 12: Non-ML analytics local differential privacy

Analytics (non-ML)

Federated analytics



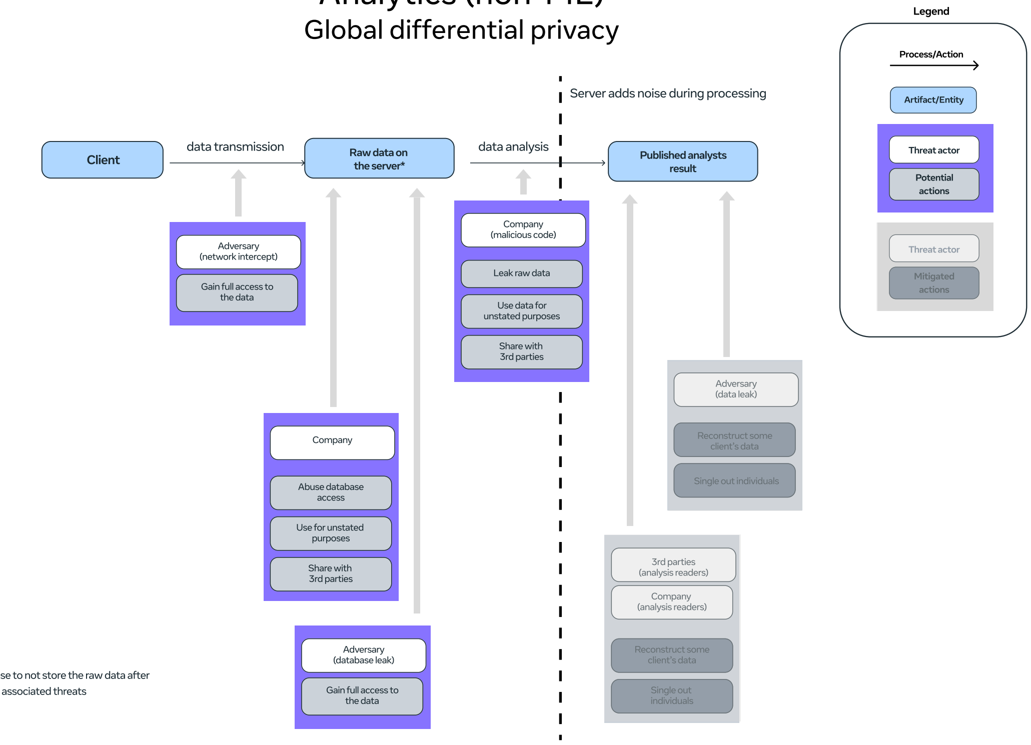
Client doesn't send the raw data, only final aggregrates

Data reconstruction and singling out is still possible

Figure 13: Non-ML analytics federated analytics

Analytics (non-ML)

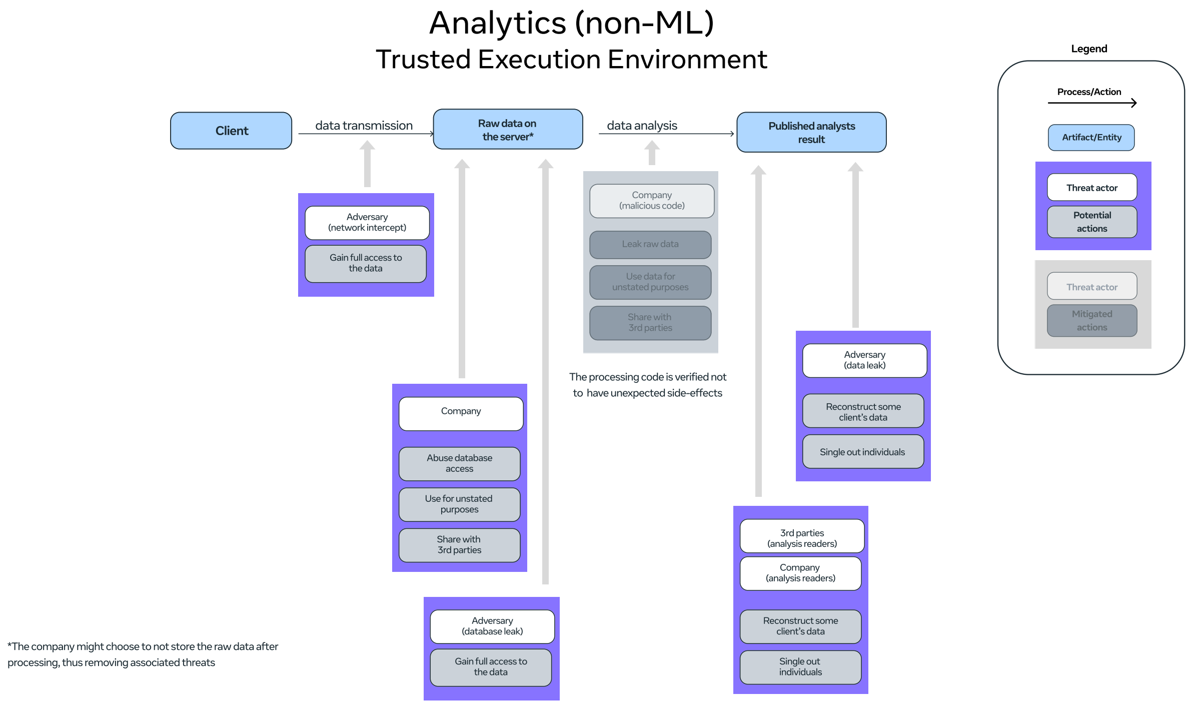
Global differential privacy



Server adds noise during processing

*The company might choose to not store the raw data after processing, thus removing associated threats

Figure 14: Non-ML analytics global differential privacy



The processing code is verified not to have unexpected side-effects

*The company might choose to not store the raw data after processing, thus removing associated threats

Figure 15: Non-ML analytics trusted execution environment

2. Cloud data analysis

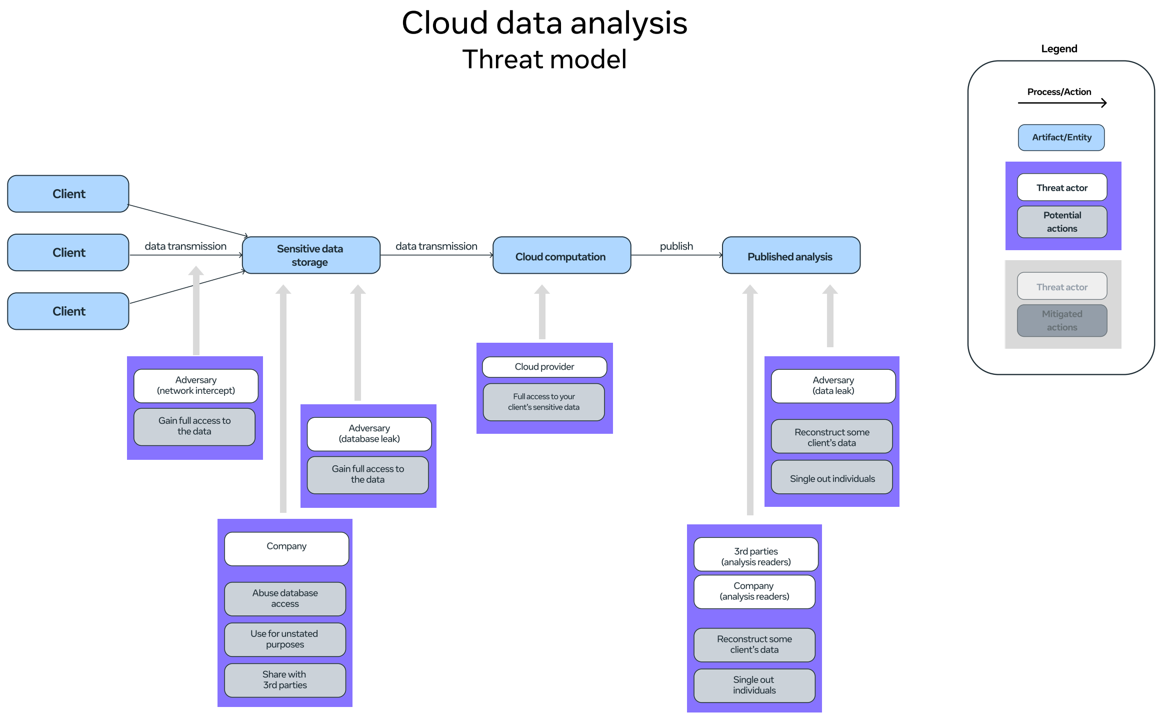


Figure 16: Cloud data analysis full threat model

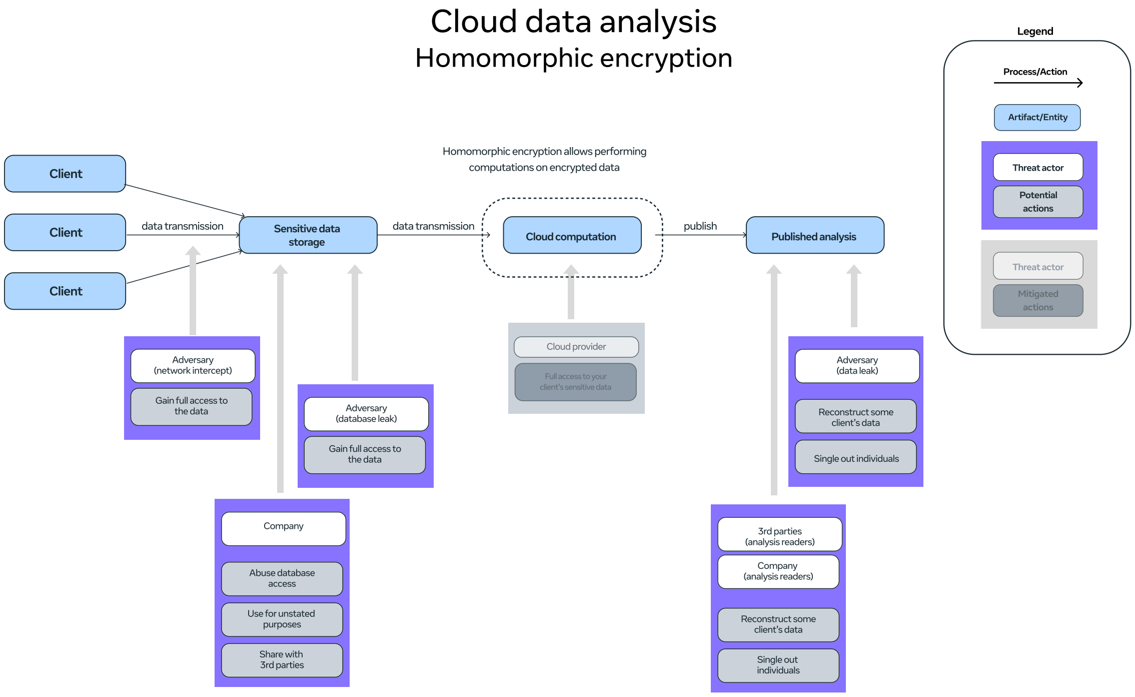


Figure 17: Cloud data analysis homomorphic encryption

3. Simple ML

ML training
Full threat model

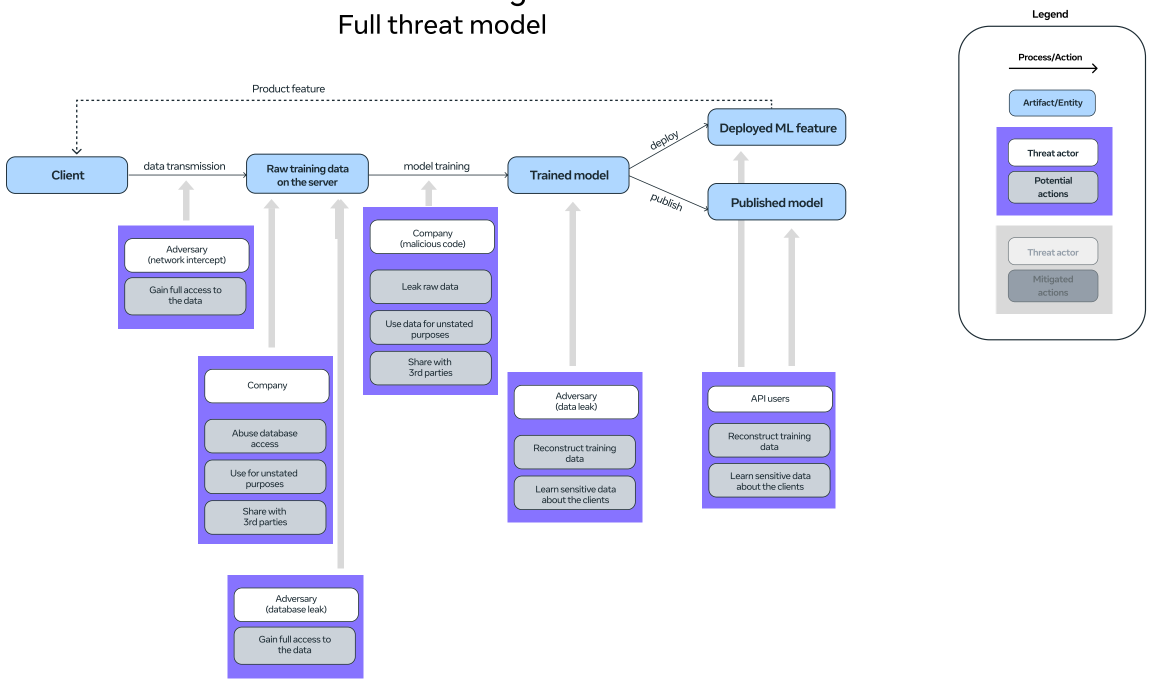
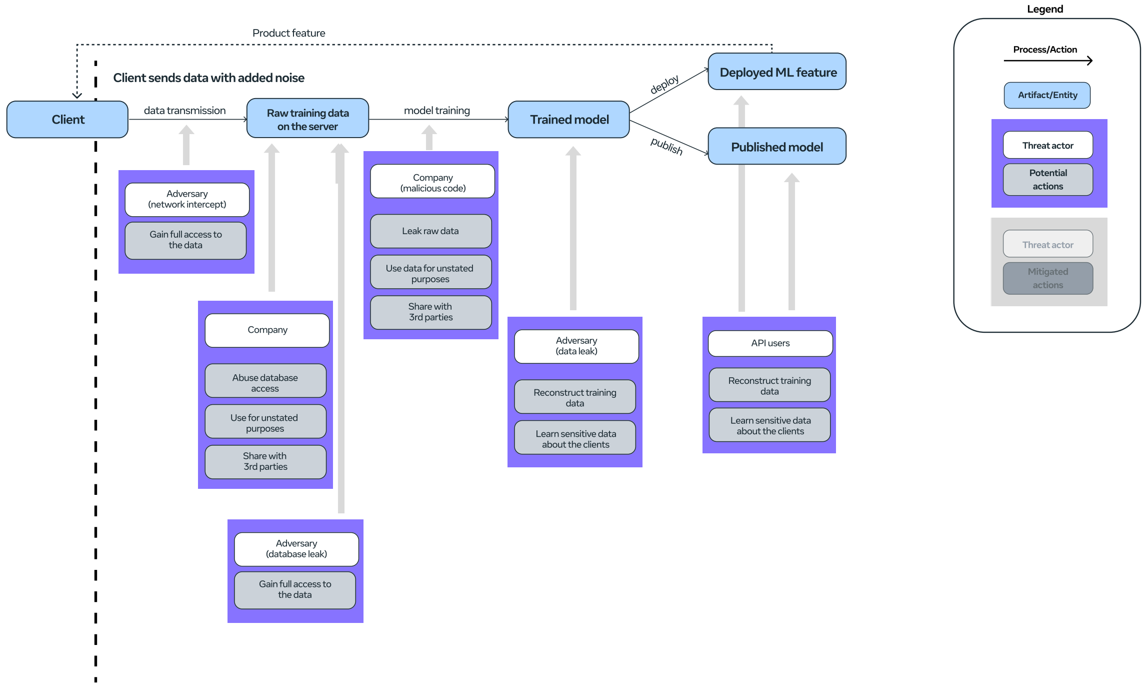


Figure 18: ML training full threat model

ML training
Local Differential Privacy



Legend
Process/Action
Artifact/Entity
Threat actor
Potential actions
Threat actor
Mitigated actions
Product feature
Client sends data with added noise
Client
data transmission
Raw training data on the server
model training
Trained model
deploy
publish
Deployed ML feature
Published model
Adversary (network intercept)
Gain full access to the data
Company (malicious code)
Leak raw data
Use data for unstated purposes
Share with 3rd parties
Company
Abuse database access
Use for unstated purposes
Share with 3rd parties
Adversary (data leak)
Reconstruct training data
Learn sensitive data about the clients
API users
Reconstruct training data
Learn sensitive data about the clients
Adversary (database leak)
Gain full access to the data

Figure 19: ML training local differential privacy

ML training
Full threat model

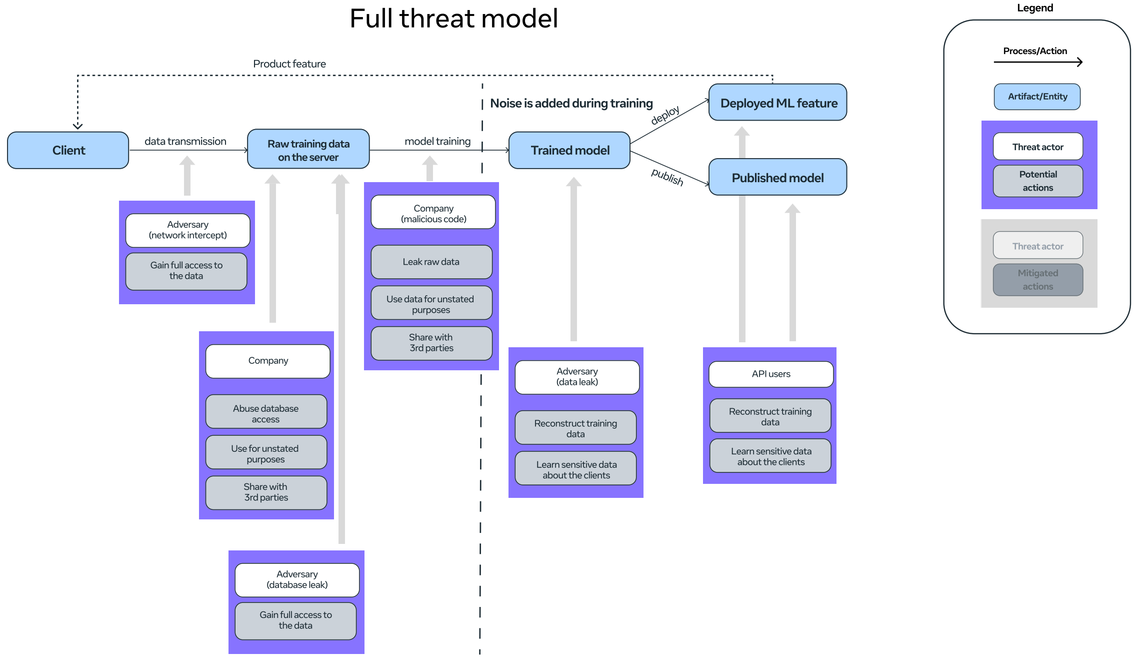
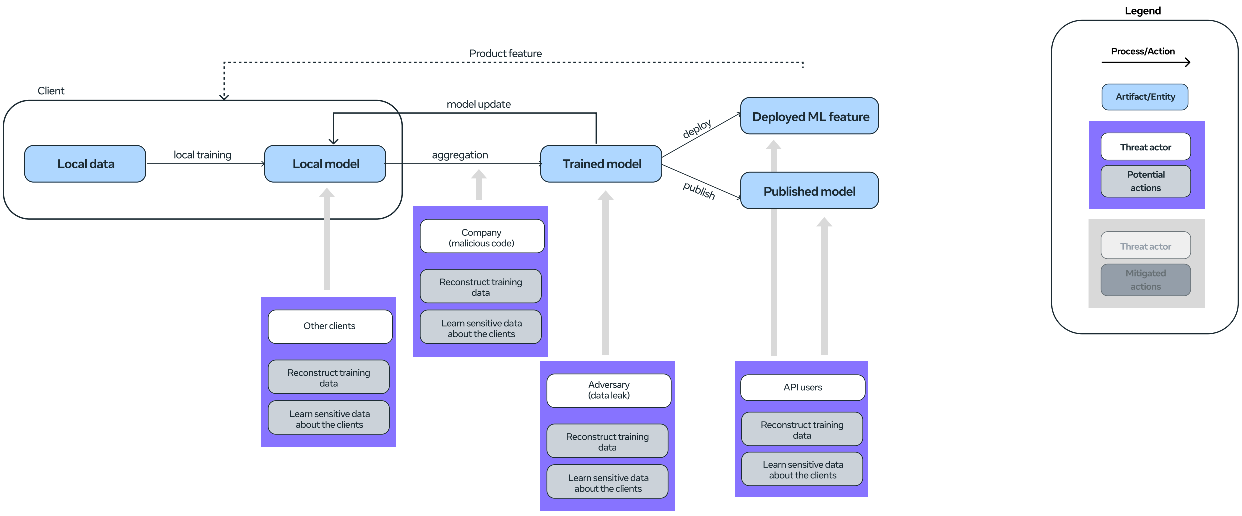

Legend
Process/Action
Artifact/Entity
Threat actor
Potential actions
Threat actor
Mitigated actions
Product feature
Client
data transmission
Raw training data on the server
model training
Noise is added during training
Trained model
deploy
publish
Deployed ML feature
Published model
Adversary (network intercept)
Gain full access to the data
Company
Abuse database access
Use for unstated purposes
Share with 3rd parties
Adversary (database leak)
Gain full access to the data
Company (malicious code)
Leak raw data
Use data for unstated purposes
Share with 3rd parties
Adversary (data leak)
Reconstruct training data
Learn sensitive data about the clients
API users
Reconstruct training data
Learn sensitive data about the clients

Figure 20: ML training central differential privacy

ML training
Federated learning



*Note: None of the risks are fully mitigated, but the severity of potential privacy violations is reduced

Figure 21: ML training federated learning

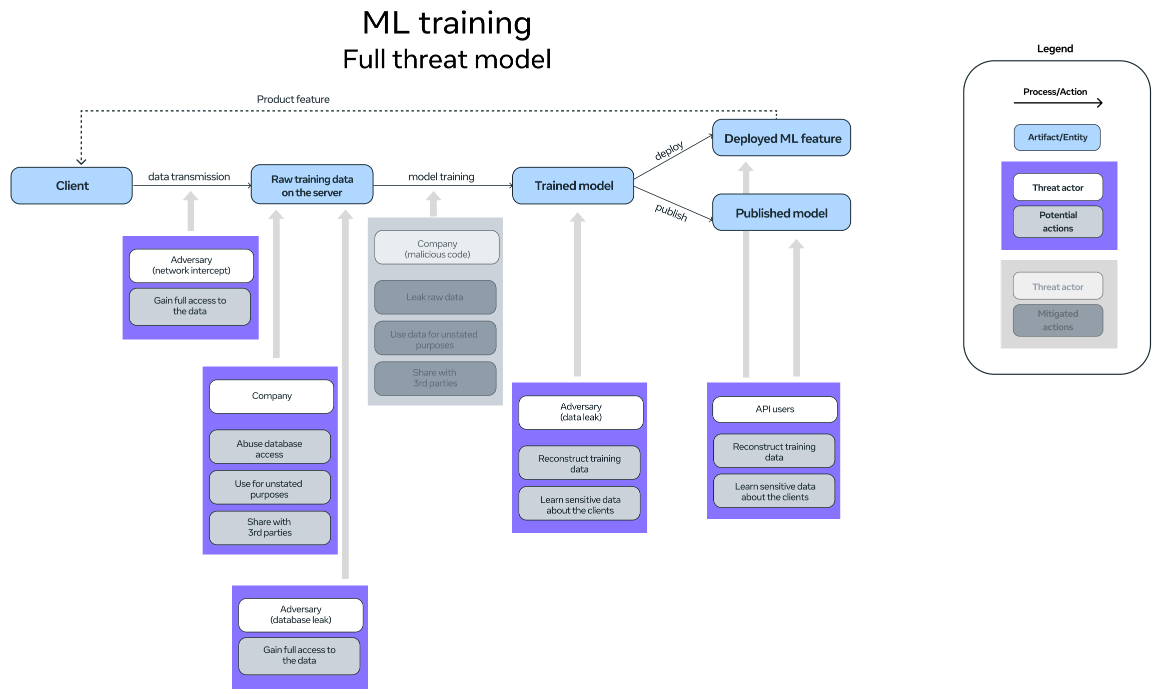


Figure 22: ML training trusted execution environment

4. Joint data analysis

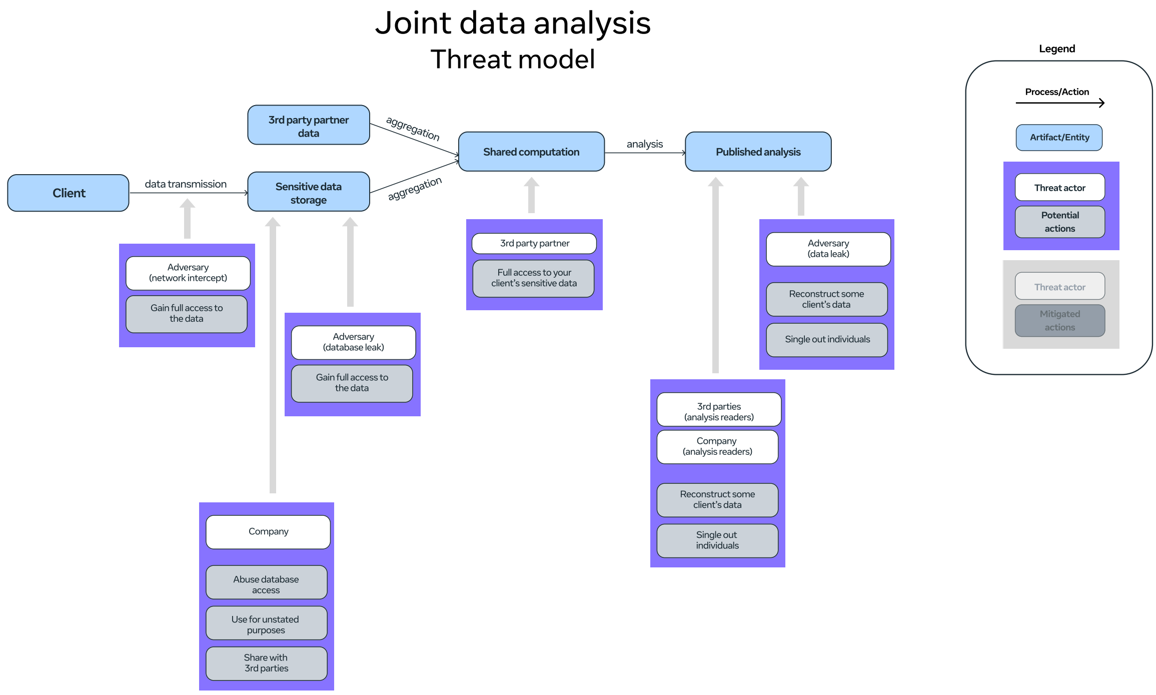


Figure 23: Joint data analysis full threat model

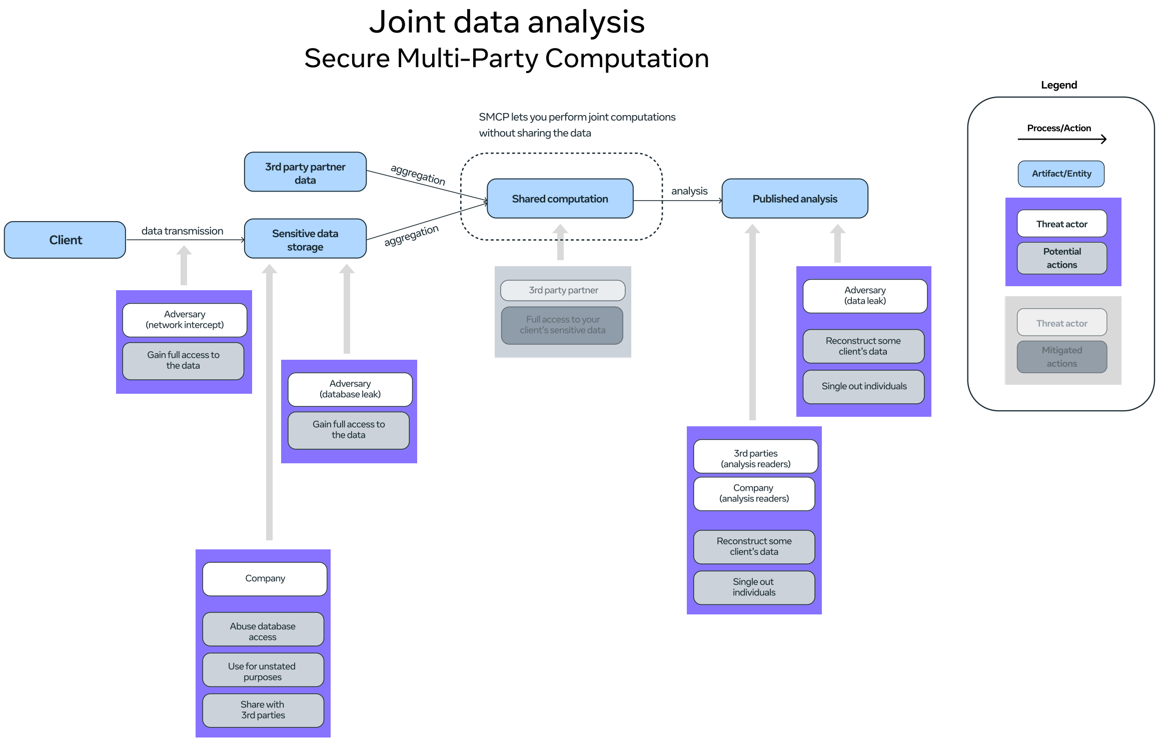

Joint data analysis
Secure Multi-Party Computation
SMCP lets you perform joint computations without sharing the data
Client
data transmission
3rd party partner data
Sensitive data storage
aggregation
aggregation
Shared computation
analysis
Published analysis
Adversary (network intercept)
Gain full access to the data
Adversary (database leak)
Gain full access to the data
Company
Abuse database access
Use for unstated purposes
Share with 3rd parties
3rd party partner
Full access to your client's sensitive data
Adversary (data leak)
Reconstruct some client's data
Single out individuals
3rd parties (analysis readers)
Company (analysis readers)
Reconstruct some client's data
Single out individuals
Legend
Process/Action
Artifact/Entity
Threat actor
Potential actions
Threat actor
Mitigated actions

Figure 24: Joint data analysis SMPC

Appendix V: Applying PETs in an AI context: AI specific tips

1. Considering PETs and privacy concerns with the use of AI/ML

As with any collection and use of personal data, processing using AI systems should be carried out with due regard for data protection.

This Appendix explores some of the key questions prompted by AI in relation to the use of PETs. This is not an exhaustive exploration of all issues; rather, it is designed to provide guidance regarding some privacy considerations that arise out of AI/ML which may be mitigated by PETs.

However, in realm of training and deploying AI/ML models, deciding on the right technology to address these concerns is not straightforward, for a few reasons:

- Not all technologies are implementable given a particular use case (for example, if you don't have data you can process on device, you can't do federated learning)
- As noted in the section above, discussing consumer and privacy concerns, PETs could be considered as solutions for different user concerns(for example, central DP addresses a membership inference, which yields the identification of a particular data subject, but not exposing data to a central server) In sum, there is no perfect individual PET to solve all your privacy problems.
- As discussed above (see Figure 10), how to implement a given PETand choose parameters is not straightforward to answer, because you have to consider tradeoffs between data utility and data privacy.
- As discussed above in our section on applying PETs for engineers, given your threat model and stages you are considering in the AI/ML model lifecycle, different solutions might apply.

To address these risks, we outlined the following tips and considerations for policy professionals, product managers, and engineers to consider as they choose their appropriate PET during the design and deployment processes of their ML system. The considerations outlined below do not provide absolute answers for the best PET in every use case and do not serve as guidance for legal compliance. However, they could be considered as a heuristic for decision makers to consider when deciding which PETs, they should use and how they should calibrate that PET.

2. Risk dimensions in an AI context that may help guide your use of PETs

When considering the type of PET that should be applied to the development or deployment of an AI system, the following considerations are useful.

- Origin and location of data (“Where did that data that you are using to train your model originate from? Public sources available on the web? Government sources? A company’s platform, an acquisition? Where is that data currently stored?”)
- Exposure of AI system (“Does your AI system in cold storage, internally accessible, is it live in product, is it open-sourced? Considering this exposure, how much access could an adversary have to data that underlies a model?”)
- Historic use of data or AI system (“How have you used your underlying data or AI system in the past and what were the rules, legal obligations, and internal policies you followed?”)
- Downstream impact of AI system (“What kind of impact will your AI system have on your users or external stakeholders that are not users?”)

Consideration	**Examples** (can vary based on regulation, this is a non-exhaustive list made to help guide companies).
Location and origin of training data	**Location:** • On user device • On server/on prem • On server but assets maintained by third-party vendor **Origin:** • Public data scraped from the web • Public or private data purchased from a vendor • Data obtained from a government source **Origin:** • Data generated on your platform that is privacy protected (e.g., protected by privacy settings) • Data generated on your platform that is accessed by/open to users on your platform • Data generated on your platform that is not visible or accessible to users
Exposure of AI system’s models	• Models are in cold storage • Models are black box • Models are glass box • Model is accessible through API • Model is open-sourced • Other access control considerations
Storage limitation	**Internal rules based on:** • Policies about the use of certain types of data (confidentiality policies, ethics policies) • Legal contracts with commercial partners • Legal obligations across jurisdictions • Compliance with a cybersecurity program • Public commitments to stakeholders • Intellectual property concerns • Use of consent, notice, opt out, or other privacy controls
Downstream impact of AI system	**AI impact risk** While not necessarily a privacy risk (for instance, an AI system built with privacy protections in mind may still be harmful to people) consider the downstream impact of your AI system on your users and stakeholders (who does it impact, what are the sensitive use cases, what type of product do you have).

Table 13: Considerations: A deeper dive

3. Discussing how PETs can be applied in certain AI contexts based on the above considerations

Dimension	Guideline about PETs that are applicable to the risk dimension (not applicable in all situations, consider these as examples)	Possible trade-offs (these examples are not exhaustive, and be sure to coordinate with your appropriate legal and product personnel to consider other examples)
Training data sensitivity	High sensitivity of training data may be mitigated by the use of differential privacy Cryptographic techniques or homomorphic encryption may allow ML training on encrypted training data that is sensitive Using synthetic data may mitigate concerns around using sensitive data and also help address fairness issues De-identification techniques can also be used to mitigate risks around using sensitive data	• Data quality loss can happen depending on differential privacy epsilon, the type of deidentification used, (e.g. the use of differential privacy to enhance privacy may reduce fairness of the dataset) • Some data is so sensitive, the use of PETs may not fully mitigate the risks of using for ML training • Cryptographic techniques and homomorphic encryption may not be available for your model training setup
Origin and location of data	Data that is originating from user devices or from non-affiliate entities that you want to use to help train an AI model may be subject to processing through Federated analytics/ federated learning (FA/FL) or Secure Multi-Party Computation to mitigate end user privacy concerns	• Technical setups for federated analytics/ federated learning and SMPC may be computationally costly or require users with certain types of devices or operating systems to provide the privacy benefit, resulting in possible fairness issues
Storage limitation	Data that is originating from user devices or from non-affiliate entities that you want to use to help train an AI model may be subject to processing through Federated analytics/ federated learning (FA/FL) or Secure Multi-Party Computation to mitigate end user privacy concerns	• Technical setups for federated analytics/ federated learning and SMPC may be computationally costly or require users with certain types of devices or operating systems to provide the privacy benefit, resulting in possible fairness issues
Exposure of data/AI system, model	Consider conducting your model training or storing your model in a Trusted Execution Environment SMPC may allow model training on data that would be otherwise inaccessible If a model is intended to be open sourced, using differential privacy may mitigate the risk of an adversarial privacy attack	• Trusted execution environments may be costly to create or access • Using differential privacy to enhance the privacy of model you intend to open source may yield unknown accuracy risks
Historic use of data	Data that is subject to certain requirements might be subjected to certain types of deidentification techniques to allow for compliance prior to being used to train the model	• Some legal regimes, contracts, or internal policies may restrict the processing of data in ways that PETs cannot mitigate for • Deidentification is a spectrum, you may want to consider the techniques you are using and their relative strength before applying them
Downstream impact	Depending on your model's impact (e.g. human impact, finance, healthcare, manufacturing, public service), you may want to consider a combination of privacy enhancing technologies alongside your AI risk strategy	• The downstream impact of your AI on stakeholders and users is something that is not solely a privacy issue, consider what you are training your model to do (e.g. does it cause any kind of harm) before introducing it into a production environment/ "in the field"

Note, that many technologies can be combined with one another depending on the combination of risk dimensions you are looking at. For example, Federated Learning can be enhanced with differential privacy or deidentification measurement techniques to ensure no data can be reproduced from individual gradients of an AI model.

4. Considering PETs in the AI development lifecycle and AI model development lifecycle

As product managers and engineers are considering where and when they should apply a particular PET, they should consider the general lifecycle of an AI/ML model. The OECD defines a model as “a computational representation of all or part of the external environment of an AI system – encompassing, for example, processes, objects, ideas, people and/or interactions that take place in that environment. Core characteristics include technical type, how the model is built (using expert knowledge, machine learning or both) and how the model is used (for what objectives and using what performance measures).”

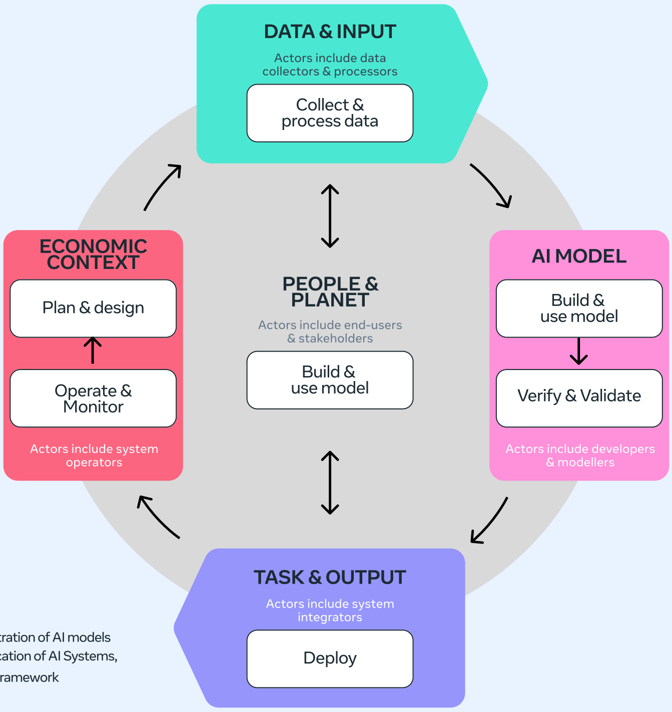

DATA & INPUT
Actors include data collectors & processors
Collect & process data
ECONOMIC CONTEXT
Plan & design
Operate & Monitor
Actors include system operators
PEOPLE & PLANET
Actors include end-users & stakeholders
Build & use model
AI MODEL
Build & use model
Verify & Validate
Actors include developers & modellers
TASK & OUTPUT
Actors include system integrators
Deploy

Figure 31: Illustration of AI models for the classification of AI Systems, under OECD Framework

As product managers and engineers are considering where and when they should apply a particular PET, they should consider the general lifecycle of an AI/ML model. The OECD defines a model as “a computational representation of all or part of the external environment of an AI system – encompassing, for example, processes, objects, ideas, people and/or interactions that take place in that environment. Core characteristics include technical type, how the model is built (using expert knowledge, machine learning or both) and how the model is used (for what objectives and using what performance measures).”

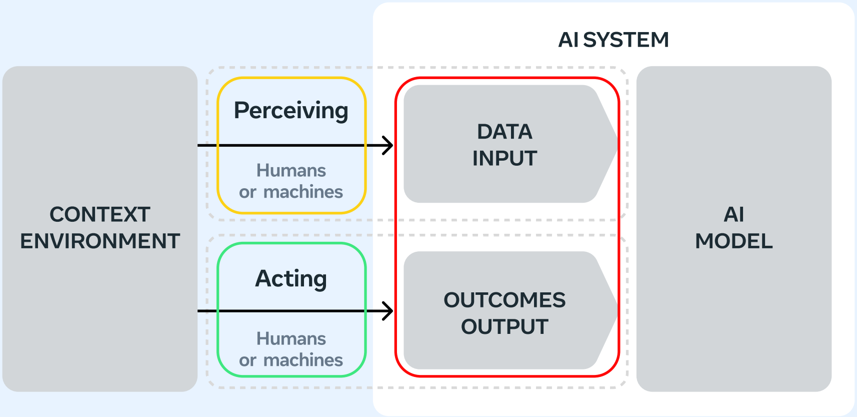


Figure 32: Impact of Models, as components of AI Systems

Below, we illustrate in the AI model lifecycle key stages that are tied to model inputs and model outputs.

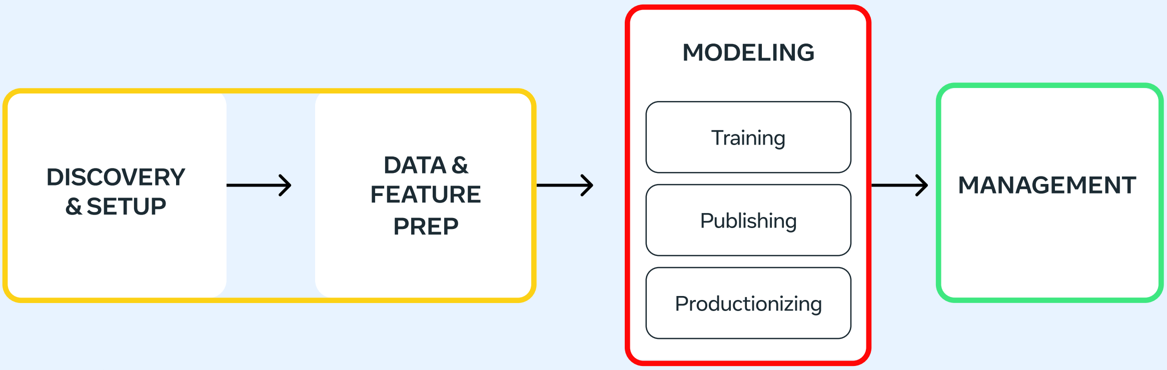


Figure 33: AI model lifecycle stages

Below, we describe the points in the timeline in an AI system's lifecycle where PETs can be applied.

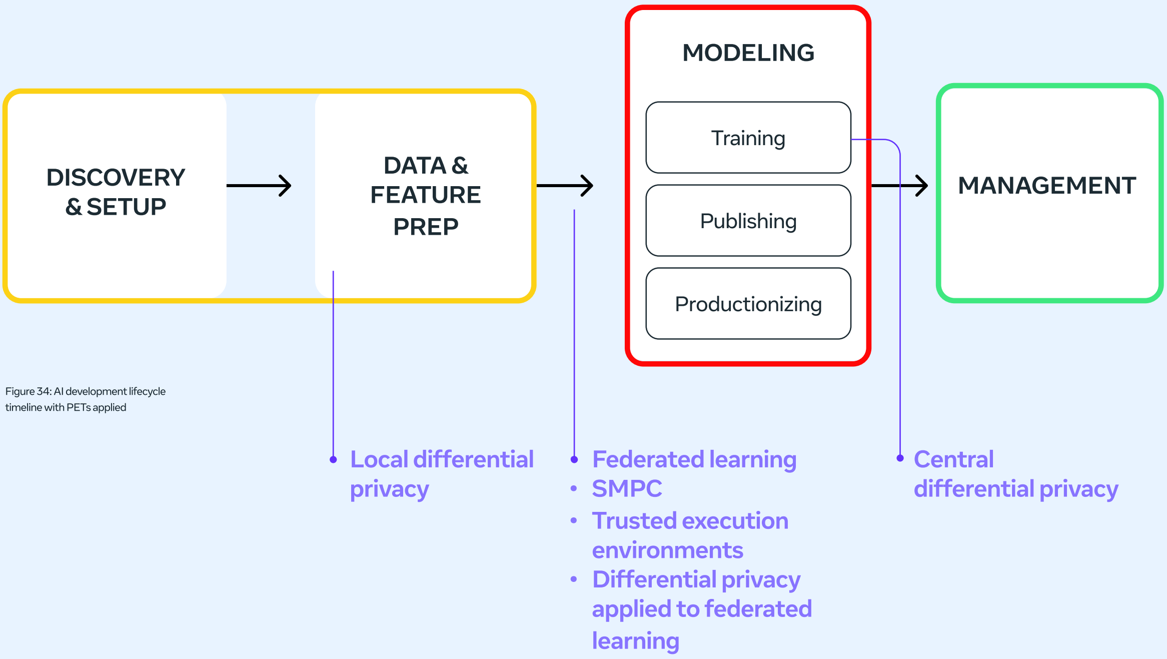

DISCOVERY & SETUP
DATA & FEATURE PREP
MODELING
Training
Publishing
Productionizing
MANAGEMENT
Local differential privacy
Federated learning
SMPC
Trusted execution environments
Differential privacy applied to federated learning
Central differential privacy

Figure 34: AI development lifecycle timeline with PETs applied

References

[1] See: the EU General Data Protection Regulation (GDPR). Brazilian Data Protection Law (LGPD), Uruguay's Data Protection Law Data Protection Act Law No. 18.331 (11 August 2008); Decree No. 414/009 (31 August 2009) and Decree 64/2020 (17 February 2020).

[2] Este manual fará referência aos princípios de proteção de dados (consulte a seção 2) e aos "requisitos de privacidade" de forma intercambiável.

[3] Autoridad Española de Protección de Datos. (2018). Guía de privacidad desde el diseño, https://www.aepd.es/es/documento/guia-privacidad-desde-diseno_en.pdf

[4] Digital Principles. Address privacy and security, https://digitalprinciples.org/principle/address-privacy-security/.

[5] See Annex to the Recommendation of the Council of 23rd September 1980: Guidelines governing the protection of privacy and transborder flows of personal data. See also: https://digitalprinciples.org/principle/address-privacy-security/, https://globalprivacyassembly.org/wp-content/uploads/2022/11/2.2.b.-Global-Frameworks-and-Standards-Workin-Group-English.pdf

[6] Risk can thus be 'quantified' using the following formula: Risk = chance x impact. If we look at the risk formula from a privacy risk perspective, we get the following: Privacy / data protection risk = Chance of unintended and/or illegal processing of personal data x privacy harm done to the individual. See also https://www.gub.uy/unidad-reguladora-control-datos-personales/comunicacion/publicaciones/guia-evaluacion-impacto-proteccion-datos

[7] The playbook looks at an information system, conceptually, as a system that ingests data, stores data, transforms / uses that data and makes it accessible to other systems or people. This definition can also be found in the glossary.

[8] A common approach in describing this process the ETL process (extract, transform, load), whereby data is extracted from an external source cleaned up and then loaded into the target system.

[9] Data sources are all the sources from which data is gathered.

[10] ACID stands for Atomicity, Consistency, Isolation and Durability.

[11] CRUD stands for Create, Read, Update and Delete.

[12] Oosterhout, E. (2012, September 10). Eight privacy design strategies [Blog post]. Retrieved March 9, 2023, from https://blog.xot.nl/2012/09/10/eight-privacy-design-strategies/.

[13] Hoepman, J. H. (2014). Privacy by design: The little blue book (Version 2.0). TILT (Tilburg Institute for Law, Technology, and Society). https://www.tilburguniversity.edu/sites/default/files/tilt/downloads/privacy-by-design.pdf

[14] Buschmann, F., Meunier, R., Rohnert, H., & Sommerlad, P. (1996). Pattern-oriented software architecture, volume 1: A system of patterns. John Wiley & Sons. https://dl.acm.org/doi/book/10.5555/539766

[15] Note that the development of practical privacy patterns is still in its infancy. You can find a list of privacy patterns here: Privacy Patterns. (n.d.). Retrieved March 9, 2023, from https://privacypatterns.org

[16] Hoepman, 2014, p. 2.

[17] Hoepman, 2014, p. 23.

References

[18] Hoepman, 2014.

[19] Hoepman, 2014, p. 7.

[20] Hoepman, 2014.

[21] Hoepman, 2014.

[22] Ibid.

[23] Ibid.

[24] Hoepman, 2014.

[25] Ibid.

[26] Ibid.

[27] George Danezis, Josep Domingo-Ferrer, Marit Hansen, Jaap-Henk Hoepman, Daniel Le Métayer, Rodica Tirtea, Stefan Schiffner, (2014) Privacy and Data Protection by Design- from Policy to Engineering. European Union Agency for Network and Information Security. They relied on Communication COM (2007)228 from the Commission to the European Parliament and the Council. On Promoting Data Protection by Privacy Enhancing Technologies (PETs). (Not published in the OJC), 2007 which practically restated J. Borking, 'Der Identity Protector', Datenschutz und Datensicherheit, 11, 1996, pp. 654-658

[28] Relevant resources include: Centre for Data Ethics and Innovation 2021 Privacy Enhancing Technologies Adoption Guide, see https://cdeiuk.github.io/pets-adoption-guide/; the UN Handbook on Privacy-Preserving Computation Techniques (2022), see https://unstats.un.org/bigdata/task-teams/privacy/UN%20Handbook%20for%20Privacy-Preserving%20Techniques.pdf; the U.K Royal Society report on “Protecting privacy in practice: The current use, development and limits of Privacy Enhancing Technologies in data analysis” (2019) https://royalsociety.org/-/media/policy/projects/privacy-enhancing-technologies/Protecting-privacy-in-practice.pdf and the 2023 report “https://royalsociety.org/-/media/policy/projects/privacy-enhancing-technologies/From-Privacy-to-Partnership.pdf”.

[29] Although PETs could possibly play a role in process oriented design strategies (such as inform and control strategies, see Figure 6.) These interactions will be further explored at a later stage in the testing phase of this Open Loop program.

[30] Article 5(1)(d) of Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC OJ L 119, 4.5.2016 (“GDPR”), and it can be mapped to the OECD Principle of Data Quality, which can be found here- https://www.oecd.org/digital/ieconomy/oecdguidelinesontheprotectionofprivacyandtransborderflowsofpersonaldata.htm#guidelines

[31] Recital 26 of GDPR. Note: "anonymization" can have two different meanings depending on whether it is used in a technical sense or a legal sense. If being used in a technical sense, "anonymization" refers to a class of techniques (that includes de-identification techniques) that directly reduce the risk of identifiability in a dataset. "Deidentification" is used to encompass that class of techniques—though some PETs documents don’t reflect this in practice. If being used in a legal sense, "anonymization" means the reduction of risk of identifiability to a sufficiently low level, taking into account both technical and non-technical means.

References

[32] Based on: Citron, Danielle Keats and Solove, Daniel J., Privacy Harms (February 9, 2021). GWU Legal Studies Research Paper No. 2021-11, GWU Law School Public Law Research Paper No. 2021-11, 102 Boston University Law Review 793 (2022) ("Keats and Solove")

[33] https://www.sciencedirect.com/topics/computer-science/cryptographic-technique

[34] Article 4(7) of GDPR

[35] Article 5(1)(c) of GDPR and it can also be mapped to the OECD Principle of Collection Limitation Principle.

[36] Article 4(2) of GDPR

[37] Article 4(8) of GDPR

[38] Article 4(21) of GDPR

[39] Article 25 and Recital 78 of GDPR

[40] Article 35 of GDPR and Irish DPC Guidance found here- https://www.dataprotection.ie/en/dpc-guidance/guide-data-protection-impact-assessments

[41] Hoepman, Jaap-Henk (2013). "Privacy Design Strategies." arXiv:1210.6621v2. p. 3

[42] For further clarification, see https://www.ibm.com/cloud/learn/etl

[43] https://stats.oecd.org/glossary/detail.asp?ID=7045

[44] Article 4(1) of GDPR

[45] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets,

[46] Frank Buschmann, Regine Meunier, Hans Rohnert, and Peter Sommerlad. Pattern-Oriented Software Architecture, Volume 1: A System of Patterns. John Wiley & Sons, 1996; ENISA, "Privacy and Data Protection by Design – from policy to engineering", www.enisa.europa.eu, December 2014, p.17

[47] ENISA, "Privacy and Data Protection by Design – from policy to engineering", www.enisa.europa.eu, December 2014, p.17-18

[48] Hoepman, J. H., Privacy by design: (The Little Blue Book), p.3

[49] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets; European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.12

[50] Keats and Solove

[51] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[52] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[53] European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.19

[54] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[55] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[56] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets; European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.14

References

[57] Article 5(1)(f) of GDPR and it can be mapped to OECD Principle of Security Safeguards.

[58] Article 4(1) of GDPR

[59] Article 25 and Recital 78 of GDPR

[60] European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.9

[61] Hoepman, J. H., Privacy by design: The Little Blue Book) p3

[62] Article 4(5) of GDPR

[63] Keats and Solove

[64] Article 5(1)(b) of GDPR, and it can be mapped to both the OECD Principles of Data Quality, Use Limitation and the Purpose Specification

[65] https://ec.europa.eu/eurostat/cros/content/re-identification_en, https://csrc.nist.gov/glossary/term/re_identification

[66] Keats and Solove

[67] Keats and Solove

[68] https://dictionary.cambridge.org/dictionary/english/risk

[69] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets; European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.14

[70] For further clarification, see https://www.ibm.com/cloud/learn/etl

[71] Article 5(1)(e) of GDPR

[72] https://edps.europa.eu/press-publications/publications/techsonar/synthetic-data_en; European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.17

[73] Article 4(10) of GDPR

[74] For further clarification, see https://www.ibm.com/cloud/learn/etl

[75] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets; European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.15

[76] European Union Agency of Cybersecurity & ENISA, "Data Protection Engineering: From Theory to Practice", January 2022, p.22

[77] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[78] https://cdeiuk.github.io/pets-adoption-guide/what-are-pets

[79] https://dataprivacylab.org/dataprivacy/projects/kanonymity/paper3.pdf

[80] https://www.sciencedirect.com/topics/computer-science/cryptographic-technique#:~:text=The%20process%20of%20confidentiality%2C%20integrity,the%20key%20to%20decrypt%20it.

[81] https://developer.okta.com/books/api-security/tls/exposed-data/, https://en.wikipedia.org/wiki/Transport_Layer_Security

References

[82] https://www.ibm.com/topics/end-to-end-encryption, see also https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

[83] See generally, https://www.frbsf.org/economic-research/wp-content/uploads/sites/4/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf "Another challenge is that PETs are in variable stages of maturity. While promising, some techniques and systems are still in early phases of development, and there is limited investment in ongoing research. Technology-driven firms with robust research and development funds are focused in this space,20 along with pockets of academic work, but PETs as a category are not yet widely studied. This variability in maturity and research adds to the complexity of PET adoption and makes it harder for firms to determine which PETs are appropriate and what resources they need to deploy them."

[84] See generally, https://www.frbsf.org/economic-research/events/2021/august/bard-harstad-climate-economics-seminar/files/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf. Different organizations categorize PETs in many different ways. To name just a few approaches, some draw lines based on maturity, some distinguish between cryptographic methods and those that affect data processing, some look to what capabilities they provide users, some distinguish between those that affect the security of computation and those that affect data fidelity, and some do not attempt to categorize PETs at all. We believe our approach—grouping PETs based on what they do technically and then mapping them to user-centric use cases—captures the myriad goals of the disparate approaches at categorization.

[85] See generally, https://www.frbsf.org/economic-research/wp-content/uploads/sites/4/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf "Privacy enhancing technologies that shield data do not alter the underlying information; instead, they make data unintelligible or unusable at certain times to prevent unauthorized parties from accessing it."

[86] See generally, https://www.frbsf.org/economic-research/wp-content/uploads/sites/4/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf "Privacy enhancing technologies that shield data do not alter the underlying information; instead, they make data unintelligible or unusable at certain times to prevent unauthorized parties from accessing it."

[87] See generally, https://www.frbsf.org/economic-research/wp-content/uploads/sites/4/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf Altering data generally refers to use a variety of methods and characteristics to "obscure identifying characteristics within datasets…when data are altered they can remain in that state indefinitely and are still intelligible, or readable, for processing."

[88] Similar to trusted execution environments, secure multiparty computation techniques are not a monolith; they are treated here at a higher level of generality simply to facilitate our analysis.

[89] https://www.frbsf.org/economic-research/wp-content/uploads/sites/4/Privacy-Enhancing-Technologies-Categories-Use-Cases-and-Considerations.pdf Rather than altering or shielding the data itself, computation altering PETs change the way the data is computed to process data in a more secure or privacy preserving way.

[90] https://ico.org.uk/media/about-the-ico/consultations/4021464/chapter-5-anonymisation-pets.pdf

References

[91] It is important to note that the considerations presented in this playbook are intended to provide a roadmap for the potential use of PETs in privacy risk management, rather than a prescription for immediate implementation. As such, they may not reflect current industry practices or be realistically achievable at scale in the present moment. Instead, we hope that this playbook will inspire organizations to explore the possibilities of PETs and to consider how they might adapt their privacy risk management strategies in the future. We acknowledge that there are many complex factors involved in the deployment of PETs and that each organization will need to make decisions based on their unique circumstances. Therefore, the information and materials presented in this playbook should be used as a starting point for discussion and exploration rather than as a definitive guide to action.

[93] We provide this list as a guidance only and do not endorse its accuracy or conformity with any law, self-regulatory framework or regulation.

[94] https://www.amazon.science/blog/machine-learning-models-that-act-on-encrypted-data

95 https://www.globalsecuritymag.com/Machine-Learning-in-Trusted,20210610,112686