Privacy Engineering: An Intro To The Field

Privacy engineering has emerged as a vital function for data-driven businesses, bridging the gap between legal and engineering teams to deliver trustworthy processes and products. Here's what to know.

Overview

- What is Privacy Engineering?
- Privacy Engineering In Practice
- Cultivating Privacy Engineering

What is Privacy Engineering?

Privacy engineering is the practice of building tools and processes that apply privacy protections to personal data. This emerging field includes a variety of activities, all focused on embedding privacy into systems. For instance, privacy engineers inspect code before deployment to assess privacy risk. They also determine best methods for anonymization. On the user-facing side, they design clear privacy controls.

Objectives of Privacy Engineering

Contemporary privacy engineering is about baking privacy into product development.

When an engineering team builds a product, such as an app for online payment processing to support financial transactions, privacy engineers design and build systems to protect the data flowing through that product. Privacy is not the systems' primary purpose, but it is a key component of the design process, now more than ever. For compliant, trustworthy, and efficient systems, privacy engineering is becoming a core focus for businesses worldwide.

Privacy engineers ask questions like: *Are the technical systems enforcing the authorized parties' responsibilities to respect users' consent choices regarding targeted advertising? How can the company implement anonymization practices on personal data while not disrupting essential business functions?*

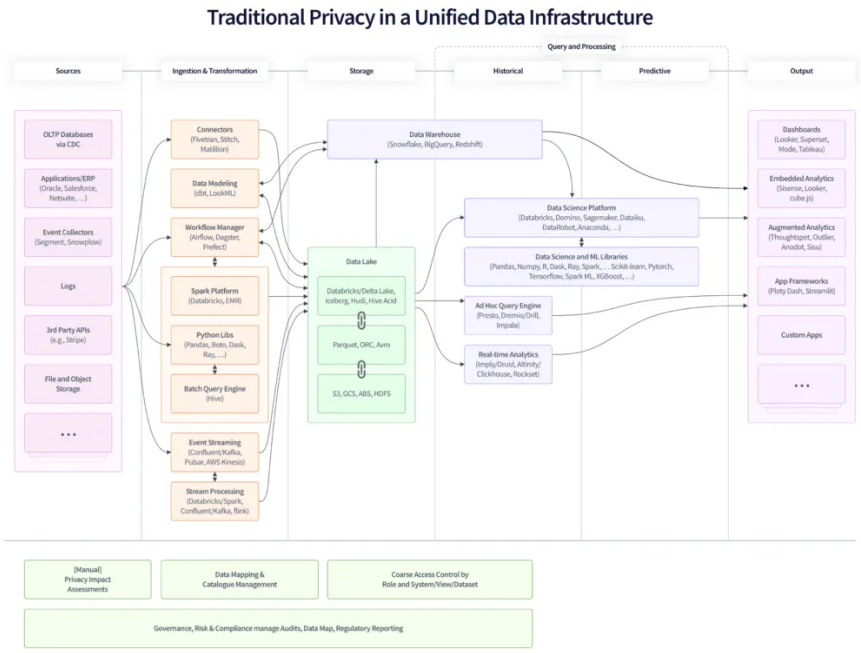
Questions of privacy engineering are inherently cross-functional, touching on legal topics like compliant consent processes, as well as engineering challenges such as data masking and retrieval. As a result, **privacy engineers bring in-demand skills** to the nexus of legal, policy, engineering, and product.

This article will introduce the activities and responsibilities of the emerging field of privacy engineering. But first, a review of today's privacy problems provides compelling evidence for increased attention on privacy engineering.

Understanding the Problem Space

For much of the twentieth century, a company processing personal data would only have that data stored in physical form, such as a health record in a file cabinet. At the time, a Governance, Risk, & Compliance (GRC) team alone could sufficiently protect this information. While domain knowledge was needed to properly manage the information, specialized training was not necessary to simply access or rearrange the folders in the file cabinet. The folder could move from place to place, but its movements were relatively slow and manual.

Today, it's a much different picture. Some data retains a physical form, but much of it exists digitally and resides in specialized data infrastructures. Entire professions and bodies of technical knowledge are dedicated to rearranging this information to meet a company's needs. The volume and complexity of data continues to increase, as data travels rapidly across companies and jurisdictions, often without a physical paper trail to record movements. Up until recently, the rapid evolution of data infrastructure had left privacy outside of core engineering workflows – a manual, reactive task.



Traditional privacy operations happen outside of product development, creating manual and reactive workflows

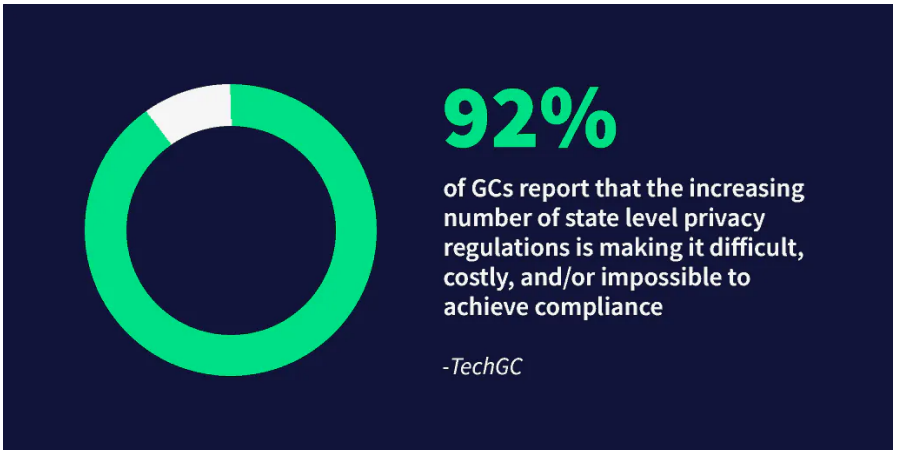
On top of the technological transformations, regulations and consumer attitudes have significantly elevated expectations for businesses' data practices. Regulations like the European Union's **General Data Protection Regulation** (GDPR) have set new precedents for how businesses must account for personal data at every stage of processing, from

initial collection to final deletion.

Finally, the act of building privacy into base tech infrastructure has become a requirement, written into regulations themselves. For example, **GDPR's Article 25** emphasizes that businesses build privacy into their technical workflows, not relying solely on policy to protect users' privacy. Emerging regulations generally require businesses to keep an inventory of the kinds of personal data collected and the methods by which they are processed. Manually completing this task is costly, if not impossible, pointing to the **privacy engineer's clear role** in modern privacy compliance.

Consumer expectations for companies' data practices are similarly increasing. According to a 2021 research report, **22%** of surveyed individuals would spend more money with brands that they trust. Trustworthiness can be difficult to quantify, but strong privacy practices are a clear indication of trustworthiness.

A comprehensive but jargon-laden privacy policy is one thing. Intuitive, compliant, and private by default user controls are another thing altogether.



With this context in mind, a need arises for a technical discipline that can mediate GRC and engineering needs. Privacy engineering fulfills that need.

Privacy Engineering In Practice

Translating Legal Requirements Into Practical Technical Protections

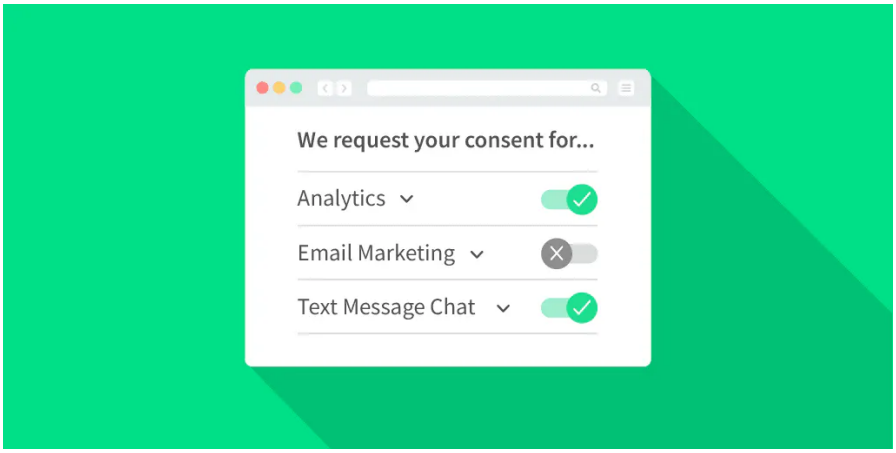
GRC teams appreciate the importance of clearly defined standards. Only with clear standards can organizations measure compliance and business success. Privacy engineers translate this mindset into a company's technical systems. Given regulations, privacy policies, and other organizational priorities, they ensure that the product or service actually delivers on those requirements, at a technical level.

This article is not an exhaustive review of all privacy engineering tasks. Instead, it will look at several widespread privacy issues to illustrate the field's overall role: processing data subject requests (DSRs) and embedding privacy standards into product development.

Privacy Engineering and Data Subject Requests (DSRs)

DSRs are a hallmark of comprehensive consumer privacy legislation, granting rights such as data access and data erasure. Completing an erasure request often involves collaboration between legal and engineering teams. The former ensures that the latter's fulfillment of the request meets regulatory requirements.

For most companies today, this process is cumbersome and manual. Among other tasks, fulfilling an erasure request demands identifying all relevant personal data across distributed systems and adhering to the appropriate erasure strategy. A privacy engineer may be tasked with ensuring that no data is overlooked in processing an erasure request. Further, the engineer designs a process that does not over-apply the erasure method, which could wipe out necessary data or break essential referential integrity between databases in the business backend.



Privacy engineers collaborate with designers to ensure that data controls are useful and usable, presenting clear options to end-users.

To develop efficient workflows for processing DSRs, privacy engineers work at each step of the DSR journey to minimize friction for end-users, engineers, and GRC teams. On the front-end, they advise product designers on effective designs and controls through which end-users can submit DSRs. While privacy engineers are not always designers, they are often conversant in effective privacy UX—which design patterns to emulate and which ones to avoid.

For the latter, privacy engineers are familiar with dark patterns: misleading UX that can trick users into acting against their own best interests, such as when the lower-privacy consent option is displayed much more prominently than the higher-privacy option. Privacy engineers can offer valuable feedback earlier in the design process to root out dark patterns before they mislead end-users and cause legal or reputational damage.

To build the infrastructure to fulfill DSRs, **privacy engineers write SQL queries that take a key, such as an email address from the requester, to retrieve all copies of the requester's personal information.** Beyond this, privacy engineers operate on the full breadth of modern tech stacks, which includes in-house NoSQL platforms like MongoDB as well as integrations with third-party SaaS apps.

For third-party privacy management, **privacy engineers might use off-the-shelf APIs or write custom wrappers around those APIs to ensure that third-party data is properly included in the scope of a DSR.** Because third-party applications process end-users' data, their relevant systems should be queried for DSR. Further, your company is held responsible under GDPR for involved third parties' compliance. In this final respect, privacy engineers can provide a technical supplement to GRC audits of

third parties to mitigate risk.

Privacy Engineering and Data Minimization

Two key steps toward proactive privacy engineering involve data minimization and retention. Data minimization is the processing of the minimum amount of data needed for the activity. For example, while a free email newsletter requires subscribers' email addresses, subscribers' credit card information is not needed and thus should not be collected in the first place. Early in the design phase for new products, privacy engineers can **prevent downstream privacy issues** by evaluating code for unnecessary data collection. Doing so avoids time-consuming corrections later in development, and it also simplifies compliance once the product launches.

Data minimization is straightforward to describe, but implementing it requires a strong grasp of a variety of data architectures and common programming languages. To ensure that code is only collecting necessary data, privacy engineers bring a working understanding of languages most used by the company's software engineers, such as Python and JavaScript, depending on the project and team. They are closely familiar with data serialization formats like JSON, XML, and YAML to hierarchically organize categories of personal data.

example_taxonomy.yaml

```yaml
#Example taxonomy – privacy data (yaml file)
person #person's information
- name
    - first_name
    - last_name
- contact #person's contact information
    - mailing-address #person's mailing address details
        - street
        - city
        - state
        - zip
    - phone #person's contact phone numbers
        - mobile
        - home
        - work
    - email
- identification #person's identification details
    - drivers-license
        - number
        - state
    - passport
        - number
        - issuing-country
```

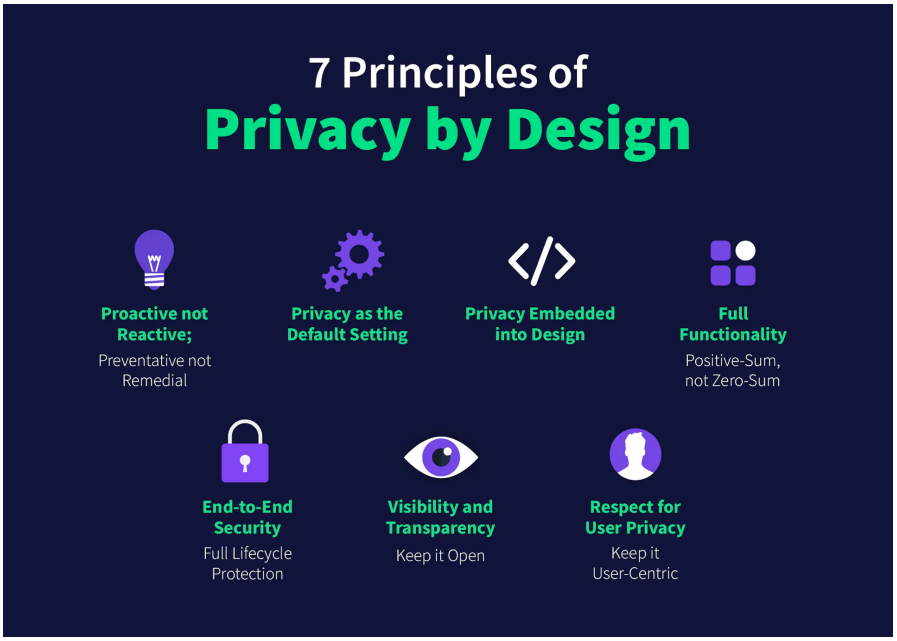
Like data collection, data minimization applies throughout the data lifecycle, including processing and storage. Because of this, privacy engineers review pipelines to see the big picture: where and how data flows through an organization. Aided by their familiarity with data infrastructure such as Databricks or Snowflake, privacy engineers tackle everything from ETL projects to centralizing data from legacy systems. Centralization of existing data into one location ensures that this data can be internally supervised and annotated. Finally, they may also develop access control lists in cloud service providers like Amazon S3 or Azure Data Lake Storage to limit access of data.

Privacy Engineering and Data Retention

While data minimization addresses what personal data is processed, data retention considers how long the personal data resides in the data infrastructure. Some data must persist for a set period of time, such as purchase data for tax purposes. Other data has a shorter lifespan. At the end of its lifespan, data should be responsibly and promptly disposed of, reducing the opportunities for unauthorized parties to access and abuse personal data. This is one example of how, by enforcing appropriate retention standards on databases containing personal information, privacy engineers also mitigate security risks.

Privacy engineers ensure adherence to data retention policies by designing and implementing automated processes for timely data erasure. A privacy engineer might be tasked with managing users' session data in a Redis store that is a cache with expiry dates. To responsibly dispose of session data once it is no longer of business value, a self-deletion process based on expiry times can create a scalable, no-friction process for the Data team while providing peace of mind to the GRC team.

Implementing Privacy by Design



A north star for privacy engineering is the Privacy by Design framework. The framework, first developed by Dr. Ann Cavoukian, integrates privacy into the design process rather than only considering privacy retroactively. The seven principles of Privacy by Design are summarized as follows:

1. **Proactive not Reactive.** Instead of assuming a product respects users' data until a regulator finds otherwise, tackle privacy concerns early by consulting a privacy engineer in the initial phases of product development.

2. **Privacy as the Default Setting.** Identify the ways in which a product processes personal data, from collecting it to analyzing it to destroying it. In each of those events, **the framework calls for more privacy-respecting settings to be the default,** across issues of data minimization, purpose specification, collection limitation, and more. For instance, if a user does not need to provide their Social Security Number in order to receive a service, the service should not collect it in the first place.

3. **Privacy Embedded into Design.** The design process should **regularly assess privacy impacts and risks** before products go out into the wild.

4. **Full Functionality—Positive-Sum, not Zero-Sum.** A fallacy in privacy debates is that in order to respect someone's privacy, some party must be put at a disadvantage. Privacy engineers work to **create products and services that are not impaired by privacy protections.**

5. **End-to-End Security.** Security and privacy are tightly related,and poor security undermines privacy. If unauthorized parties can access personal data, individuals' data rights are directly jeopardized. **Activities like access control and encryption must be secure from a technical standpoint.**

6. **Visibility and Transparency.** At first glance, it might seem bizarre that Privacy by Design calls for visibility, but visibility is integral to trustworthy systems. **Privacy-related policies and procedures, when appropriate, should be clearly accessible to users and internal stakeholders.**

7. **Respect for User Privacy.** Users' privacy controls should be usable, from straightforward consent toggles to timely fulfillment of DSRs. **User-facing visuals, copy, and workflows should prioritize accuracy and accessibility.**

It is the task of privacy engineering to implement these principles into data systems.

Cultivating Privacy Engineering

The Privacy Engineer's Evolving Role

Privacy engineering can be housed under a variety of teams: Design, GRC, Engineering, Security, and more. The very titles for practitioners of privacy engineering can vary. The explicit title of "Privacy Engineer" is gaining traction, but the executors of many of today's core privacy engineering functions go by other titles like "Software Engineer," "Technical Program Manager," and "Data Engineer," with the privacy engineering projects embedded into their specific work.

Just as privacy engineering may fall under a variety of position titles, the professional trajectories to a privacy engineering role are diverse. Individuals might train specifically for this program, pursuing a degree such as the Privacy Engineering master's from Carnegie Mellon University. Alternatively, practitioners from both the privacy and software realms may expand their skills to meet in the middle at privacy engineering. Since it is an interdisciplinary field, a variety of other career trajectories may also lead to privacy engineering. As the field becomes more prominent and structured, specific training programs will likely increase.

Broad Privacy Engagement

Privacy engineering does retain some aspects of a software engineering role, in writing and reviewing code to ensure that processes are respectfully processing users' data. The day-to-day work, however, involves a great deal of communication outside of strictly engineering circles. Privacy engineers interface with different stakeholders with diverse training and priorities, from lawyers to product designers. Those in the field report a mix of technical and non-technical responsibilities. Beyond the technical responsibilities detailed earlier in this article, they consult with legal counsel on risk mitigation when it

comes to consent controls.

Another non-technical responsibility that may land in the privacy engineer's domain is that of privacy educator for colleagues in the company. As with information security, strong privacy practices depend on teams applying a working knowledge of privacy into their day-to-day work. Beyond consultations with developers and legal counsel, the privacy engineer can further mitigate privacy risks by training their team in best practices to implement across the organization. This role as educator is directly in line with the Privacy by Design principle of Proactive, not Reactive. Privacy engineers are responsible for building processes that work smarter, not harder, to make usable privacy a reality.

Learn More

- The International Association of Privacy Professionals drafted a **sample job description** for a privacy engineer.
- The National Institute of Standards and Technology provides a **handful of guides** to privacy management, which can inform the creation of enterprise-level privacy engineering initiatives.
- Learn about today's most pressing issues in engineering privacy and respect in this **recorded conversation** with Twitter's Head of Privacy Engineering Lea Kissner, The Rise of Privacy Tech's founder Lourdes Turrecha, and Ethyca's CEO Cillian Kieran.