Privacy

Privacy Engineering Superheroes

Privacy engineers are essential to both preventing and responding to organizational privacy problems.

DOES YOUR ORGANIZATION want to offer cookie choices without annoying popups? Do you want to share sensitive data in aggregate form without risking a privacy breach? Do you want to monitor data flows to ensure personal information does not end up in unexpected places? What if personal information does leak out and now you need to clean up the mess? Do you want to do this in the messy, failure-prone world of a large system? Who ya gonna call? How about a privacy engineer!

The privacy profession is dominated by lawyers—who certainly play a critical role—but privacy engineers are often the real superheroes when things go wrong, and essential to preventing privacy disasters. Privacy engineering has emerged as a growing discipline focused on finding practical and often technical solutions to privacy protection. Organizations hire privacy engineers to develop privacy-protective products and services, build tools to promote and monitor privacy compliance throughout their organization, and to detect and remediate privacy problems. Privacy engineers may play a holistic role or focus on specific areas such as front-end, back-end, user experience, product management, or legal compliance.¹

One of us (Lea Kissner) is a privacy engineering practitioner who has led privacy engineering teams at four companies, and one of us (Lorrie Cranor) has spent almost 20 years in

academia and now co-directs an academic program to train privacy engineers.^a Over dinner a few years ago, Kissner complained about the lack of a venue for privacy engineering practitioners to discuss their problems and solutions and learn about research that could be applied to their work. Cranor noted that privacy researchers would also benefit from learning more about actual problems faced by practitioners and from having a forum where they could share their research results directly with practitioners. By the time we finished din-

ner, we had a skeletal plan for a new conference.

Privacy Engineering Practice and Respect

We started the conference on Privacy Engineering Practice and Respect (PEPR) in 2019,^b bringing together privacy engineers from academia, industry, civil society, and government to share their expertise. Privacy engineers from industry have discussed ways data deletion can fail in truly bizarre ways in large-scale systems, while academic researchers have presented user study

^a See <http://privacy.cs.cmu.edu>

^b See <http://pepr.tech>

results providing insights into why users do not seem to understand many privacy-related icons. PEPR brings us all together to discuss privacy-related ideas and how they work (and fail) in practice. Because the traditional academic paper format is focused on conveying research results rather than experiences from practice, and because many of the practitioners we want to hear from are not experts at writing academic papers, PEPR asks prospective speakers to submit talk outlines rather than papers. Since 2020, all PEPR talks have been recorded and made freely available after the conference.^c

PEPR 2021 was fully remote, but was probably the largest (virtual) gathering of privacy engineers ever, with over 500 participants attending talks and engaging in discussions. PEPR focuses on building respectful products and systems instead of breaking them. It is easy to look around and see the ways things are broken. It is easy to succumb to nihilism. But we want to build the world we want to live in and sys-

Privacy engineering has a fundamentally different focus than much of the privacy field as a whole.

tems will not get better unless we build them better. So while we are interested in both breaking and building, we lean toward building.

For the same reason, we look more broadly than privacy and also focus on respect. According to Wikipedia, “Respect is a positive feeling or action shown toward someone or something considered important, or held in high esteem or regard; ... it is also the process of honoring someone by exhibiting care, concern, or consideration for their needs or feelings.”

Building respectful systems requires

considering concepts beyond privacy, including security, trust, safety, algorithmic fairness, and more. We must move out of our disciplinary silos and consider our work in a broader context.

Consider for example the issue of filtering out unwanted email. If one looks at it from the perspective of privacy, the first questions tend to be around whether it should be done by default and whether it is reasonable to build models over the contents of multiple users’ inboxes. But it is also a security issue: some of those unwanted email messages are direct security threats such as phishing or messages that contain malware. And without scanning large amounts of email to understand the quickly changing threat landscape, it is effectively impossible to identify those threats. However, building models based on everyone’s personal email may be problematic. Are these models fair or do they operate unfairly toward people using, say, particular dialects?

Privacy engineering has a fundamentally different focus than much of the privacy field as a whole. Many organizations are interested in privacy because they want to be compliant with

^c Slides and recordings available at <https://bit.ly/3tx6j1l> and <https://bit.ly/3hkUW7R>

ICCCQ

The Second International Conference
on Code Quality (23 Apr, online)

Static/Dynamic Analysis, Program Verification,
Bug Detection, and Software Maintenance

www.icccq.ru

CfP closes on 18 Dec

In cooperation with
ACM SIGPLAN and SIGSOFT
IEEE Computer Society

laws such as GDPR, CCPA, and a whole alphabet soup of regulations coming into effect. While we certainly want systems to be built and operated in accordance with the applicable laws, that should be a side effect of building privacy-respectful products and systems in the first place. Compliance is necessarily reactive. It is responsive to failures of the past. If you are doing new things, then you are likely to hit new failure modes—compliance is not going to be sufficient. For one, when things go really wrong, no one cares about paperwork. But also those laws are moving; if you have built your systems around the checklist from compliance, then it is unlikely to be able to smoothly incorporate changes to the rules along with changes to the threat models your users face. Proactive privacy engineering considers how the changing world impacts your strategic direction to help shape your products and systems to better support your users and the folks affected by your systems.

Privacy Engineering Specialties

Addressing privacy engineering requires many skillsets—here are some of the major specialties and roles that have developed in this young field.

Analysis/consulting. These folks look at your product/system (or better yet, the plans you have to build one), ask questions, find failures before they happen, and help you design in a way to robustly avoid those failures. For example, someone develops a new feature and the analyst asks questions such as “How can the user delete this information and is it really deleted?” and “If we put this nifty crypto in there, we can avoid collecting this information at all. Does that open up abuse vectors?” Privacy analysis/consulting folks have a skillset somewhat akin to security reviewers, but usually with a heavier emphasis on how humans of various stripes interact with each other and your product. They may well audit code.

Privacy products. This is where you are building the privacy technology users can see, such as account deletion pages, interfaces that let users see and control their data, and so forth. It is usually best when privacy affordances are part of the main product rather than a standalone tool.

While we are interested in both breaking and building, we lean toward building.

Math and theory. Need to do anonymization? Need to analyze whether there is some kind of funky joinability risk across all your datasets? Need to figure out what is going to happen when you delete a particular type of data? Is it going to break your abuse models? Math.

Infrastructure. If you have got infrastructure, you probably need privacy infrastructure. For instance, when you want data to be deleted, you need a system to kick that off and then monitor it. You need access control and probably something to deal with cookies. Privacy engineers who build infrastructure have the infrastructure-building software engineering skills as well as knowledge of privacy.

Tooling and dashboards. You might have a data deletion or access system, but how do you help the humans in your organization understand what is going on in there or get access? Tooling! Tooling and dashboards are also extremely useful for things like efficient, accurate analysis and review of systems. Good remediation tooling holds a mirror up to the rest of the organization and tells them both how they are doing, exactly what they can do to get better, and how much better they are expected to be.

User experience (UX). The difference between a privacy-respectful product and a privacy-invasive product can sometimes be a matter of user experience. Is there transparency about what information is being collected and how it will be used and are users able to easily understand and implement privacy choices? A lot of privacy-engineering UX focuses on design and testing of privacy-related affordances (settings, dashboards, dialogues, icons, and so forth). There is also a need for privacy engineer involvement

in writing clear and accurate privacy notices—ideally this is not a job left only to the lawyers! Privacy UX research uses qualitative and quantitative methods to study people and how they interact with a product.

Privacy policy. This is not always done by privacy engineers, but we strongly recommend having someone with privacy engineering skills in the mix. It is very easy to write aspirational policy or policy that does not work with systems as they are. Both of these are bad. Kissner has written and managed privacy policy at both large scale (Google) and small scale (Humu).

Privacy process. Process design is key to getting things done. A deeply engineering-integrated process that aligns the incentives of everyone involved is key to making privacy a “well-lit path,” something smooth and efficient for the rest of the organization. If you make the rest of the organization grumpy, you are not going to get good results.

Incident and vulnerability response. Things will go wrong in a real system. There will be bugs and process failures and new issues you have never even thought of before. Incident responders use a cool head, excellent communication skills, and knowledge of how things go wrong in privacy to help folks work out what has gone wrong, how to fix it, and then how to keep that whole class of failures from happening again.

Of course, privacy engineers do not work in a vacuum, and they must collaborate with lawyers, policy makers, software developers, and many others. With all these roles and specializations, there are many opportunities for all kinds of privacy engineers with diverse skillsets to help organizations build privacy into their products and services and help save the day when things go wrong. □

Reference

1. Brook, B. The disciplines of modern data privacy engineering. IAPP. (Sept. 9, 2020); <https://bit.ly/3DZv774>

Lea Kissner ([lkissner@twitter.com](https://twitter.com/lkissner)) is Head of Privacy Engineering, Twitter, San Francisco, CA, USA.

Lorrie Faith Cranor (lorrie@cmu.edu) is Director and Bosch Distinguished Professor in Security and Privacy Technologies, CyLab Security and Privacy Institute and FORE Systems Professor, Computer Science and Engineering & Public Policy, Carnegie Mellon University, Pittsburgh, PA, USA.

Copyright held by authors.