Limitations and Pitfalls of Integrating PETs in Online Age Verification

Sylvain Chatel[1], Christian Knabenhans[2], Wouter Lueks[1], Mathilde Raynal[2], Carmela Troncoso[2,3], and Ádám Vécsi[3]

[1]CISPA Helmholtz Center for Information Security
[2]EPFL
[3]Max Planck Institute for Security and Privacy (MPI-SP)

Requiring age verification to access online services is a controversial topic. The main motivation for introducing age controls is the protection of children. Requiring a certain age to access particular online services impedes children's access to content inappropriate to their age. Yet, this benefit is often contrasted with the dangers to privacy, freedom of information, and power balance that such checks can result in. Privacy-enhancing technologies (PETs), specifically PETs based on zero-knowledge proofs, are often portrayed as a means to bring balance to this complex problem.

In this paper, we present our reflections on whether PETs can actually bring that balance. Concretely, we reflect on the fundamental limitations of PETs when it comes to addressing the problems associated with age verification regardless of architectural decisions; and on issues stemming from the architectural choices in which PETs that support online age verification are integrated "as a service" or in the form of libraries.

1 Limits of PETs protection in age verification pipelines

The integration of PETs to implement age restrictions is commonly seen as a positive development, and is often portrayed as the step needed to make age-related controls societally acceptable. Yet, PETs only solve a part of the problems arising from the introduction of age-based restrictions for content access.

PETs cannot prevent privacy leaks beyond the dataflow where they are embedded. PETs do reduce dataflows, but only the dataflows in which they are embedded. Proving statements about a user's age in zero knowledge does not prevent browsers and servers from learning other personal data about the user via further collection and tracking (e.g., metadata collection: IP, browser fingerprints, etc.). Thus, while introducing PETs can reduce data collection, it is not guaranteed that adding PETs would result in an overall enhancement of users' privacy. We note that in the web environment, tracking based on metadata is still possible even if the user is denied access to the content in a privacy-preserving way.

PETs cannot prevent censorship. Restricting access to data is inherently related to the notion of *censorship*. If age verification is put in place, it opens the door to implementing censorship by re-purposing the existence of the age-verification checkpoint. For example, the access condition can be set to a specifically-chosen value to prevent access to certain groups of users beyond children. We stress that the censorship capabilities enabled by age-based restrictions are induced by *the very existence* of a control spot, regardless of whether the control is implemented using PETs, or which PETs are implemented or their implementation.

PETs can facilitate censorship. In addition to not being able to *prevent* censorship, the use of PETs can actually *foster* censorship. On paper, age-based restrictions are limited to the value of an attribute. In reality, age-based restrictions also apply to users who cannot produce proof, e.g., those with devices that do not have the capability to run the privacy protocols, or those who cannot find a trusted issuer to vouch for them being of age. Because the use of PETs is not always inclusive, age-based restrictions increase the attack surface of censorship adversaries that can have new ways of restricting access to information, in a way that is not possible to eliminate via technology (i.e., selecting one or other PET to implement the age-based restriction).

PETs cannot prevent circumvention. In the age verification context, security can be defined as the capability of preventing minors from accessing content deemed inappropriate for them (according to a policy, e.g., a policy imposed by regulators). Thus, while PETs can improve the privacy of the system, using PETs, by themselves, cannot ensure security of the restrictions.

Additionally, several circumvention methods can easily be envisioned to bypass age verification. Among the main circumvention methods we can expect: the use of VPNs to bypass geo-dependent controls,[1] the acquisition of valid credentials (for example at a black market, or simply accessing the phone of an adult) to access the content illegitimately, or the access to content in platforms without age control (e.g., BitTorrent). These circumventions are fully independent of private data and thus cannot be prevented by the use of PETs. In fact, similarly to censorship, circumvention methods are independent of the implementation of the age-based content access restriction, and thus, cannot be prevented by any design decision pertaining to the age verification pipeline.

How prevalent these circumvention methods would be among those minors the technology is trying to protect is hard to say. Adding friction is likely to prevent some access, but will not deter motivated minors who seek to gain access to restricted content.

Takeaway: A lose-lose situation. PETs are not all-solving. They can solve privacy problems but it is important to be aware that they cannot prevent circumvention of restrictions nor solve issues related to misuse.

Given these circumvention risks, it is difficult to measure the potential benefits of age-based content access control. Even more so because some dangers (like censorship) are inherent to the existence of such access control and cannot be mitigated from a technological perspective. This could lead to several drawbacks. First, under the cover of privacy enhancement, PETs can introduce means to enable censorship and discrimination to large-scale systems. Counterintuitively, the use of zero-knowledge-based PETs might worsen this situation: by legitimizing the use of age verification despite the unproven benefits, they might cast a negative image on other privacy technologies—such as VPNs, which could receive negative press (as Tor often does)—preventing their use in many legitimate and socially important scenarios. Finally, PETs themselves might be criticized for not fully solving the problem, leading to unfounded skepticism in the usage of PETs for purposes other than age verification.

With regards to misuse, we argue that, when discussing how access restrictions can avoid being re-purposed for censorship, the discussion should acknowledge that, for most if not all cases, this is just *not* possible. Censorship can be implemented by exploiting the very existence of a check, regardless of its implementation. It is thus outside of the technological realm to mitigate censorship capabilities once the control is in place.

2 Pitfalls of integrating PETs in age verification pipelines

PETs have the potential to greatly increase privacy by minimizing data flows and by minimizing the trust placed in servers and service providers operating them. In this vision, there is an implicit assumption that the integrator of these PETs has full control over how these PETs are configured.

Exercising full control over the PETs and its design parameters requires expertise to be able to understand, implement, and verify PETs implementations. Such expertise, however, might not exist when relying on novel, very complex, or highly customized and optimized PETs. Instead, this knowledge might be concentrated at one or very few points, often at large companies.

This concentration of knowledge leaves integrators of highly complex PETs solutions vulnerable with respect to these groups. Since no good alternative bases of knowledge are available, there will likely exist only a single, third-party implementation. This leads to the integrator having limited say in which privacy-enhanced functionalities the building-blocks provide, and how applications interface with these building blocks.

While these risks are exacerbated when building blocks are closed-source, they also exist when open-source implementations are available. The problem is the complexity of the design. Concretely, when implementing online age verification, if the underlying zero-knowledge technology is highly complex, the following restrictions might arise.

Lack of control on functionality. When relying on third-party building blocks or libraries for highly complex PETs, the integrator has very little control over the functionality provided. Because only few groups can even understand the implementation, such knowledge is highly unlikely to be available to the implementor, thus making changes will be very difficult, even if the implementation is open source. Very specific optimizations (e.g., to ensure good practical performance) might make

[1] See for example: https://www.bbc.com/news/articles/cn72ydj70g5o and https://arxiv.org/pdf/2402.03255

it even more difficult to make changes. As a result, any legitimate desire to make changes (e.g., to add a new age category; or to implement another feature) depends on the willingness of the third party to make these changes.

When the PET is instead used as a service, where a third party has full end-to-end control over the ecosystem (e.g., because it also provides applications and backend infrastructure) legitimate integrators can also not prevent additional functionality from being enabled. This can lead to reductions in the privacy provided.

An example of such a reduction in privacy would occur when the provider enables range proofs of a user's age being within a specific range. Age verification is typically thought of as a binary computation in which the prover proves to a verifier that their age (as certified by a trusted issuer) is greater than a threshold (which depends on the service being accessed). This solution provides maximum privacy for users, because it reveals the minimum amount of information: 1 bit (older or younger than the threshold). This means that the anonymity sets for users is as large as possible given the threshold that is imposed by the content restriction policies. When the PETs service starts to support range proofs, the amount of information leaked increases considerably. With services, however, the integrator has no control over such leakage.

Unintended standardization of data formats. One of the main goals of APIs, as those provided by libraries and building blocks, is to standardize the inputs and outputs of particular functions. When a system integrates a library, the integrator must ensure that dataflows in the system are compatible with this library. While natural, this introduces a new constraint for the system developers: they must adopt the data formats imposed by the library.

For example, a browser integrating a zero-knowledge library that only accepts mdoc-like data structures to represent identity documents and their attributes, would force all websites and services to only use this data format. This means that either the entirety of the Internet moves to converge to the format accepted by this third party library or the library developers accept to broaden their API to support additional formats. As it is not reasonable to assume that all providers have the capability to change their services, nor that regulations can all be made tailored to implementation decisions so that incompatibilities do not arise by design, the decisions of libraries developers have the potential to leave a (potentially large) number of services and service providers incapable of use the zero-knowledge proofs and join the ecosystem.

Take-away: Complex designs concentrate third-party power. Concentration of power is often seen as service providers gaining power over their users to decide and constrain how users can interact in the digital world. The above two examples illustrate that the perils of centralization of power are not necessarily limited to service providers, or even to actors that explicitly appear in a standard age verification architecture (like issuers).

Providers of complex privacy technologies, in particular zero-knowledge proofs, that are integrated by some (or all) of the actors in the age verification ecosystem, can also amass significant power by (implicitly) deciding who can use the technology, for what purpose, and what the privacy guarantees – that anyone in the system can enjoy – are. This decisional power is intrinsically embedded in the functionality implemented by the building block (e.g., through the kind of proofs that are enabled, or the API).

3 Overall Position

In this piece, we highlight issues that we believe are often disregarded in the discussion on age verification. Either because the complex ecosystem of actors involved in the supply chain of the age-verification pipeline implementation is overlooked, or because the positivism about the capabilities offered by technology precludes reflections on issues that stem from the introduction of a new functionality, regardless of its implementation.

We believe that surfacing these issues is of utmost importance to the discussion. They are necessary to define the boundaries of protection that any age verification technological implementation can provide, from both a security and a privacy perspective. A discussion of technical implementations that misses this perspective runs the risk of overestimating the protection provided by the system, and as a result, support deployments that result in more harms than benefits for the minors age verification aims to protect, and adults alike.

We hope we can join the workshop in October to discuss these aspects in person with the community.