Global Content Revocation on the Internet: A Case Study in Technology Ecosystem Transformation

Narek Galstyan[1,2] James McCauley[3] Hany Farid[1] Sylvia Ratnasamy[1] Scott Shenker[2,1]

[1] UC Berkeley [2] ICSI [3] Mount Holyoke College

ABSTRACT

Common wisdom holds that once personal content such as photographs have been shared on the Internet, they will stay there forever. This paper explores how we could allow users to reclaim some degree of their privacy by "revoking" previously shared photographs, hindering (but not eliminating) any subsequent viewing or sharing by others. Our goal is not to build a system that can withstand determined efforts to subvert it, but rather to give well-intentioned users the ability to respect the privacy wishes of others. Achieving this goal at scale will eventually require the participation of large content aggregators, and they are unlikely (putting it mildly) to find our proposal compelling. We therefore propose an approach we call technology ecosystem transformation (TET) that begins with a transitional and more easily deployable (but not fully scalable) design that does not require the participation of large incumbents but is designed to change user and societal expectations enough so that these companies would find it in their interest to adopt the approach we propose here. The intellectual challenge in this TET approach is finding transitional designs that (i) have parties willing to deploy it and (ii) once deployed, would change the incentives for the incumbents so that they would be willing to adopt the proposal.

CCS CONCEPTS

• **Security and privacy** → *Social aspects of security and privacy*;

KEYWORDS

Security; Privacy; Social aspects of privacy

ACM Reference Format:
Narek Galstyan, James McCauley, Hany Farid, Sylvia Ratnasamy, Scott Shenker. 2022. Global Content Revocation on the Internet: A Case Study in Technology Ecosystem Transformation. In *The 20th ACM Workshop on Hot Topics in Networks (HotNets '22), November 14–15, 2022, Austin, TX, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3563766.3564099

HotNets '22, November 14–15, 2022, Austin, TX, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9899-2/22/11…$15.00
https://doi.org/10.1145/3563766.3564099

1 INTRODUCTION

Today, with only rare exceptions, pictures that have been spread online remain available regardless of the wishes of the original owner. Sometimes this merely causes embarrassment, such as an old picture from a youthful drunken evening clouding the image of a now-prim adult. More crucially, there are cases where young people are coerced into taking and then sharing compromising pictures of themselves, followed by the pain and, in some cases, suicide associated with the realization that the photos are being widely shared online [11]. Thus, our failure to preserve privacy of photos on the Internet is not just a matter of embarrassment, but can be far more profoundly harmful for young people who can so easily be shamed by their peers.

This situation remains unresolved, but not because of technological barriers; in Section 3, we describe a system composed of known techniques that would significantly mitigate the spread of such photos. Rather, the lack of resolution is due to an *ecosystem failure*, where the set of actors in the content distribution ecosystem have not prioritized addressing the problem, primarily because there are no strong financial incentives to deploy such a system, and there are strong financial incentives not to.

This paper is a call for action on this issue, outlining steps towards a solution; we do not assume that the incumbent content providers would automatically adopt our proposal, but instead propose transitional changes (which we call our bootstrap design) that are easily deployed by non-incumbents that will eventually make such a solution be in the incumbents' own self-interest.

Technology ecosystem transformation (TET) is relevant when a technology ecosystem, by which we mean the set of actors in a particular technology sector, does not adequately meet the needs of society. In some cases regulation can force the actors to change their behavior, but in many cases regulation can be politically impossible, technically infeasible, and/or hard to enact and enforce internationally. Instead of relying on *government* intervention, the TET approach (also see [27]) starts with a *technical* intervention that (i) can be deployed by a set of "first-movers" who are already motivated to act and (ii) once the technical intervention reaches a significant level of adoption, it then changes the incentives for the current incumbents (*e.g.,* by changing the economic or legal landscape) so they, purely out of self-interest, change their behavior in a way that better meets the needs of society with respect to the particular problem being considered.

Note that the technical intervention need not solve the problem perfectly, nor scale to universal adoption; it merely needs to change the behavior of the incumbents in an appropriate manner. In addition, the technical intervention need not involve novel technology; in fact, its chances of adoption are probably higher if it only uses familiar technology. While not about the invention of new technologies, TET is most definitely a challenging design problem, in that it requires identifying a technical intervention that satisfies the two stringent criteria above: it can be deployed by a set of motivated first-movers, and its deployment will likely change the behavior of the incumbents for the better. TET is similar to "mechanism design" in theoretical economics (see chapter 7 in [14]), but here the mechanism must be realizable in deployable technologies, not merely theoretical incentive schemes.

In our design, which we call the Internet Revocation System (IRS), the technical intervention involves ledgers (where photo ownership can be registered), browser extensions (that prevent "revoked" photos from being displayed), and proxies (that provide viewer anonymity and scaling enhancements). While this intervention can only handle a limited scale (perhaps up to 100B images, as we argue in Section 4), by the time the system reaches this limit, content providers will be incentivized to internally adopt similar techniques, and these internal implementations can scale as needed (because the required operations are only a small fractional addition to their current workflow). Our proposed solution merely reuses standard techniques and, as outlined in Section 3, is similar to proposed solutions for certificate revocation and provenance tracking. However, we use these techniques for a different purpose, providing a technical intervention that will change the incentives for, and thus the behavior of, the incumbents.

Many have previously considered how to control the spread of information in networked systems; here we discuss three particularly relevant efforts. The Oblivion system [28] provides technical support for various "right to be forgotten" regulations by allowing people to have pictures of themselves removed, even if they were not the one to have *taken* the picture. Oblivion is more general than IRS (focusing on all those impacted by a photo, not just the owner) but inherently reactive (removing a photo once it is posted, whereas IRS proactively tries to prevent such photos from being posted or viewed). We see these as complementary efforts. The work on Secure Data Capsules [20] binds data to associated policy information that defines acceptable uses. While having greater policy flexibility than IRS, the system does not handle changes in policy for widely replicated data, as is required here. IRS shares goals and techniques with the Online Certificate Status Protocol (OCSP), which was proposed for and is primarily used to check X.509 Internet Certificate validity in HTTPS [25] (but see [4] for another use case). However, IRS operates under more stringent performance (latency and scale) requirements and a weaker threat model, entailing different trade-offs. In addition, IRS must be adopted by larger ecosystem to be useful, which is our main innovation.

2 OVERVIEW

While our introduction laid out our vision in general terms, we now provide more precise descriptions of our assumptions, intended use cases, goals and non-goals, and relevant technologies. Before doing so, we clarify our terminology relating to a particular photo: "owner" refers to the person who took the photo, and "viewer" refers to those who seek to view and/or reshare that photo. In addition, while our treatment focuses on preventing the unwanted sharing of photos, our approach applies more generally to other digital media (such as personal videos) that are discrete (*i.e.,* unlike text), have a clearly identified owner (again, unlike text), and are intensely personal (so the owner cares more about revocation than most viewers care about seeing the content).

Assumptions: When it comes to respecting the privacy of others, we assume most users will be willing to respect the wishes of an individual owner. We do not naively expect users will look away if they see a picture labeled "don't look!"; rather, we think many (but definitely not all) users would be willing to use a system (browser, website, etc.) whose default behavior was to not download, display, or reshare/upload pictures that had been explicitly revoked by their owner. Note that we do not make this same charitable assumption for all types of content. Many viewers willfully and ingeniously disregard copyright protections in order to view or illegally profit from movies and other high-value but impersonal content. In such cases, however, the content has a high value for the viewer (where a private photo would typically not) and it isn't clear that most viewers see content companies as deserving of good will.

Use cases and goals: To motivate the system requirements, we consider two broad use cases. First, there are photos the owner always intended to keep private but through either accident (*e.g.,* mistakenly uploading them to their Facebook page) or malfeasance (*e.g.,* their phone was hacked, and all the photos put online) they become public. Ensuring the revocability of these photos requires that ownership must be established when the photos are created, not when they are shared. Second, there are photos which the owner initially intended to be freely shared, but later changes their mind and decides that they would like to revoke and make private. Addressing this use case requires that the mechanism allow the owner to revoke all copies of a photo at an arbitrary future time. Based on these assumptions and use-cases, and the nature of the ecosystem, we have five main technical goals:

Goal #1: *Give owners control over their content:* The owner of a photo should be able to revoke access to it (i) even after it has been shared and reshared, (ii) without individually tracking down and requesting the removal of every copy, (iii) without the requirement to divulge the content of the image before revoking it (in contrast to [7, 17]), and (iv) without divulging their identity to arbitrary third parties (*i.e.,* the system is informed that the owner has requested that the picture be revoked in a way that the ownership is verifiable but the owner is not identified).

Goal #2: *Preserve privacy for viewers:* Viewing an image should be subject to the owner's wishes, but should not expose the identity of the viewer to any parties beyond those to whom their identity is exposed today (*i.e.,* websites already know what pictures a user is viewing, but we do not want the revocation mechanism to reveal any additional information).

Goal #3: *Empower viewers and systems to behave well:* The ecosystem should let a viewer and/or a system know when they are viewing/displaying or resharing an image against the wishes of the owner. This act should either be prohibited or should require explicit confirmation or action from the user.

Goal #4: *Opting in should be low-overhead:* Viewers choosing to respect owner preferences by using IRS should not experience significant degradation in the performance of their applications.

Goal #5: *The system should be robust against most benign photo and metadata alterations:* When photos are uploaded to sites, metadata is often stripped and various manipulations (such as transcoding) are applied. These should not interfere with an owner's ability to revoke photos.

Nongoals: It is also important to clarify our nongoals, because otherwise our quest would be impossible:

Nongoal #1: *Protection against willful violation:* We do not try to enforce privacy against skilled users who are intent on viewing or sharing prohibited content.

Nongoal #2: *Protection against third-party photos:* We do not try to protect against someone who owns and shares a picture that may be harmful to others. IRS is built around a specific notion of content *ownership*, not content *impact*. Ownership can be cleanly established when the picture is created, while the harmful impact on others cannot be ascertained by an automated mechanism. While preventing such third-party impact is a laudable goal, we leave this problem to Oblivion and other systems.

Nongoal #3: *Automatically handle modified content:* When sharing, some users might modify content, either incidentally or to purposely avoid detection. We do not expect the system to automatically handle all such modifications. We handle some modifications via watermarking and do provide an unambiguous notion of ownership to facilitate easier adjudication when human intervention is required (both of these issues are discussed in Section 3).

Nongoal #4: *Instantaneous revocation:* We believe that IRS provides benefits even if it does not implement revocation instantaneously, and insisting on instantaneous revocation would make both the bootstrap design and the eventual design of IRS unscalable, though we expect the delays to be far smaller once the eventual system is adopted.

Relevant Technologies: There are two recent technologies that are relevant to our design, that we mention here.

C2PA: The Coalition for Content Provenance and Authenticity (C2PA [10]) has defined open technical standards that give publishers, creators, and consumers the ability to trace the origin of different types of media; this involves the entire content supply chain, starting from origin device (camera, recorder, etc.), to design and newsroom edits, all the way to the consumer. To accomplish this, C2PA proposes a new set of media metadata primitives that can be embedded in media files in a backward-compatible manner or be hosted remotely by the content owner. Though still nascent, the initiative has significant industry support (see [19] for a similar proposal from a different consortium). Even though IRS solves a different problem, it shares many technical challenges with C2PA and can benefit from the adoption of the C2PA metadata standard and the infrastructure C2PA industry partners create.

PhotoDNA: This is an image-identification technology used for detecting child sexual abuse material (CSAM) using robust hashing, allowing for the identification of altered forms of a photograph [13]. PhotoDNA is widely used to detect most of the CSAM reported to the National Center for Missing & Exploited Children.

3 APPROACH & EVENTUAL SOLUTION

We first outline the four basic operations supported by IRS, then describe the technical challenges our design faces, and briefly sketch the eventual solution.

3.1 Operations of IRS

The technical components of the current ecosystem for handling photos involves the recording camera (along with associated software), browsers and other applications through which viewers access those photos, and the various websites and services where these photos are found. To this ecosystem, IRS adds another component called *ledgers* that are essentially timestamped databases of photos allowing IRS to support the following four basic operations:

Claiming ownership of a photograph is the act of entering it into a ledger with enough cryptographic information so that the owner can later provide proof of ownership.

Labeling a photo consists of attaching to it a ledger identifier.

Revoking a photograph consists of updating a flag in the ledger, after the owner proves ownership.

Validating is checking that a photo has not been revoked; this is required before a photo can be displayed, saved to disk, or shared (*i.e.,* uploaded), as none of these should be allowed on revoked photos. C2PA, through its industry partners, aims to build cloud infrastructure that would remotely host part of the media authentication chain whenever the chain is too large to fit in image metadata. Though this infrastructure is only meant for a small set of trusted media authenticators, it could be extended to act as a more broadly used ledger. We expect there will be several commercial ledgers, offering different pricing and auxiliary services, and together they constitute a database of all registered photos in IRS.

Recall that among our goals for IRS are that it preserve privacy of both viewers and owners, and does not result in significant degradation of the viewing experience. To that list, we add that the system must not impose undue load on the ledgers. Our bootstrapping design (described in Section 4) must meet these requirements within the constraints of the prevailing ecosystem, while the eventual solution (described below) can make very different assumptions about the actions of the incumbents.

3.2 The Eventual Solution

Our eventual goal is to have widespread ecosystem participation in IRS by websites and applications where content is shared and reshared, particularly social media sites; we will call these *content aggregators*. Once this participation has been achieved, IRS's operation at a high level is fairly simple. Whenever a photo is uploaded to a content aggregator, the aggregator checks with the associated ledger to make sure that the photo is not revoked, and thereafter periodically rechecks the revocation status. We now describe this in slightly more detail (omitting many subtleties for lack of space), but note that we expect the complexity to be hidden by user-facing applications that record and upload photos.

When taking a photo, the camera (or owner-controlled software) generates a unique key pair for the photo, hashes the photo, and then encrypts the hash with the private key. The owner then claims the photo with a ledger: the ledger records the encrypted hash, the public key, an authenticated timestamp (as in [1]), and a Boolean "revoked" flag, and then hands back a unique identifier that refers to both the ledger and the specific photo. The owner safely stores the original photo, the private key, and the identifier, and then labels the photo with two forms of metadata that both encode the identifier: explicit metadata (carried in normal image metadata fields) and a watermark that encodes the metadata into the pixel data itself while causing little or no perceptible distortion. Because the identifier has relatively few bits, the watermark can be made robust to many benign picture manipulations (*e.g.,* compression, cropping, tinting) [2, 6, 18, 24], which is relevant to our Goal #5. We also assume that content aggregators supporting IRS keep IRS-related metadata intact (since it is designed to preserve user-privacy, and privacy is one of the main reasons sites strip metadata now). To revoke a photo, the owner simply requests that the ledger flip the Boolean flag in the ledger record.

When a user uploads a photo to an aggregator, the aggregator inspects the metadata and watermark. If they agree, the site then checks with the ledger (using the identifier); if the image has been revoked, the upload is denied. If the explicit metadata or watermark disagree or one of them is missing (indicating that the photo has been modified in some way that has lost metadata), the upload is also denied. Note that this does not prohibit common (and potentially valid) cases of modifying and reusing photos, such as adding text to create memes; rather, the intention is to encourage those making derivative images to transfer the metadata to the modified version so that it is also revoked if the original is revoked. If a photo has neither a watermark or metadata indicating it has been claimed, the aggregator can either reject the photo or claim it (and watermark it) in a custodial role so that it can later be revoked (as in the attacks described in Section 5). When an aggregator provides a response to an application or browser containing a claimed photo, it includes in metadata cryptographic proof that it has recently verified the non-revoked status of the photo.

The process of uploading or viewing a photo does not violate the privacy of owners (aggregators merely check ledger records, with no ownership identification revealed), the privacy of viewers (since the sites already know what photos users are viewing), nor does it cause high load on ledgers or delays for viewing (pictures are only validated on upload and periodically thereafter). There remains the question of whether claiming a photo violates the privacy of the owner (*i.e.,* reveals an association between the photo and the owner to an arbitrary third party). Nothing about the IRS design itself requires that a ledger entry be linked to anything but the key pair used for the image, so an owner can protect their anonymity by simply not linking the key pair with their own identity. Some ledger implementations, however, might store payment information in a way that allows such an association to be made; a privacy-focused ledger could use a payment system that intentionally makes such an association difficult even if their database is leaked (*e.g.,* a payment system where an owner buys tokens which are exchanged with other users in a mixing market before being used to pay for claims).

At this point we should address one obvious loophole in the system. Even though a photo has been claimed by its owner, another person could claim a copy of the photo themselves and therefore try to override any revocation by allowing that copy to be widely distributed. We thus add to IRS an *appeals* process (similar to the current processes for reporting inappropriate content) whereby the original owner can lodge a complaint against the ledger on which the copy has been claimed (or, if the photo has not been claimed, against the site displaying the photo, but here we discuss the former case). The original owner presents the ledger with the original photo and a signed timestamp of the original claim, along with the copied version of the photo. The ledger then compares the original with the copy, using robust hashing (as in PhotoDNA) and/or human inspection. If they believe that the copy is derived from the original photo, they then mark it as permanently revoked and (if appropriate) take legal action against the person claiming the photo. This step is fairly heavyweight, but it does not rely on vague judgements about whether the picture is harmful, only whether it is derived from the original photo. Aggregators could also keep a database of robust hashes of their current content and check all newly uploaded photos against this database to ensure that they use the original metadata (so that revoking the original will also remove images derived from it).

4 THE BOOTSTRAP PHASE

4.1 The Bootstrap Design

The eventual solution described above only requires reasonably minor adjustments by content aggregators: preserving IRS-related metadata and checking with ledgers when photos are uploaded (and periodically rechecking existing photos). Unfortunately, there is little reason for content aggregators to take this step because there would be no immediate payoff if unilaterally pursued as its effectiveness relies on an entire ecosystem of components. Additionally, some aggregators are geared more towards engagement than privacy and adopting IRS would reduce engagement. If, however, IRS were to reach some threshold level of adoption then the incumbents would find themselves interested for two reasons: for those companies branding themselves as "pro-privacy" this would be seen as a competitive advantage (and adoption by a single aggregator would be effective, because the bootstrap phase has established the other components of the ecosystem), and for all companies not supporting IRS, their lack of support could become a legal liability (*e.g.,* if a claimed and revoked picture were shown by an aggregator, and harm resulted, the aggregator could potentially be sued because the owner's intent was clearly knowable to the aggregator).

The question is, how do we reach this threshold level of adoption. This is where the TET notion of a technical intervention is relevant: we need a temporary and partial solution to achieve this threshold. We believe the right place to make this intervention is within browser software. First, this leaves content aggregator sites unchanged but still covers many users (both for viewers who directly use browsers, and for the many applications, both mobile and desktop, that use browser software as their base). Second, several of the major browsers are already actively working on (and even competing on) privacy protection features (*e.g.,* Mozilla, Brave, and Apple), and they may be willing to be the "first movers" in adopting IRS. Thus, our proposal should be seen as a design that one or more browser vendors could adopt by (i) adding support for IRS to their browser and (ii) running a ledger (with ledger service possibly being integrated into existing paid privacy services such as Mozilla VPN [22] or Brave Talk [8]). Modifying a browser and running a ledger are straightforward: the question is how can this be done at scale, in a way that supports privacy and low-latency for the viewers with reasonable ledger loads. This will require adding proxies and Bloom filters to the design, as we explain below.

4.2 Privacy

A naive IRS-enabled browser would reach out directly to a ledger to check for photo revocations. Unfortunately, this would leak information about a user's browsing activity to the ledgers of viewed photos. Recent and ongoing developments have been combating just this sort of leakage in different contexts (*e.g.,* DNS queries) such as in Mozilla's Trusted Recursive Resolver program [23] (enabled by default in Firefox), Oblivious DNS [26] (currently offered by Cloudflare, PCCW Global, SURF, and Equinix), and Apple's Private Relay [3]. At their most essential, these solutions insert trusted proxies which aggregate the requests from many users. We propose making use of this same approach: browsers will not directly query ledgers, but will make queries through an IRS proxy; indeed, we could imagine some of the same organizations which offer the various existing anonymizing proxy services would extend their offerings to include IRS.

4.3 Viewing Latency

One might worry that a revocation check before displaying every labeled photo would create an intolerable degradation of performance for end users. We investigated this possibility in three ways. We first turned to the HTTP Archive Web Almanac study, which categorizes any website that fully renders in under 1.8s as having "good performance" [5], and notes that over 60% of studied sites take over 2.5s. Any reasonably responsive ledger would produce delays that would be a small fraction of this (say, under 100ms, as in [12, 26]).

Second, we noted that one need not wait for page resources to be fully loaded before issuing revocation checks – one can generally check a photo as soon as its metadata has been downloaded. In many cases, this can hide significant ledger latency. For example, when loading pinterest.com (a typical photo-heavy site), as long as revocation checks complete in less than 250ms, there is *no* delay in page rendering.

Lastly, we built a prototype ledger and browser extension that performed revocation checks. While a much more complete user study is warranted, we did not notice additional delay when scrolling through a variety of web sites containing claimed images.

4.4 Ledger Load

If every labeled photo must be looked up before being displayed, the load on ledgers could easily become enormous. This could make it prohibitively expensive to host a suitably scalable ledger in this bootstrap phase. Fortunately, the proxies described above can ameliorate this issue by caching lookups (which would also further reduce viewing latency).

The load on ledgers will also depend on common usage patterns. We assume that many photos will be automatically registered and revoked (allowing an owner to manually unrevoke ones they want to share); consequently, a high fraction of *total* photos will be revoked. However, we can also reasonably assume that a very high fraction of *viewed* photos are *not* revoked; that is, most photos that are made available on content aggregators for general viewing are not shared against the owner's will. Thus, the vast majority of ledger requests in response to a browser viewing a photo will result in the ledger responding with a not-revoked response.

Given this, proxies can utilize Bloom filters to lighten the load on ledgers. Each ledger would produce a Bloom filter of their claimed photos (it is in a ledger's best interest to provide such Bloom filters as they reduce their load), which the proxies would download and then take the OR of all ledger Bloom filters. We assume these will be updated regularly (perhaps hourly), and transferred with a delta encoding such that the update traffic will be low. Such a filter allows a proxy to make a quick determination of whether a labeled photo *might* be revoked: if the photo does not hit in the filter, it is definitely not revoked and no actual ledger query need be performed. If a photo hits in the filter, it may (or may not) be revoked, and a real query must be performed. Note that during early adoption, when the photo population is small and revenues to the ledger vendors minuscule, one could use the same strategy to reduce the load on the proxies by inserting a Bloom filter in browsers themselves.

Using a standard Bloom filter (see more recent advances in [9, 15, 16]), a 1GB filter would provide a 2% false-hit rate with a population of 1 billion photos, thereby lessening the load on ledgers by a factor of fifty. Similarly, a 100GB Bloom filter would provide a similar error rate for a population of 100 billion photos. We think proxies could easily support Bloom filters of this size, and that once the population of photos in the bootstrap phase of IRS reaches anywhere close to 100 billion photos, the ecosystem incentives will start to kick in and the major content aggregators would support IRS with internal implementations.

Not all sites will adopt IRS after the bootstrap phase, but their decision to not respect owner-privacy will be known because browsers could mark such sites (as they do with TLS icons), third-party rating services could publicize their lack of adoption, and search engines might lower their rankings.

5 DIRECT ATTACKS AND UNINTENDED CONSEQUENCES

All technologies, no matter how high-minded their original purpose, can have unfortunate and unintended consequences; IRS is no exception. Below we discuss some potential attacks that might give rise to bad outcomes.

Direct Attacks: A relatively naive attacker could insert incorrect metadata and/or apply enough cropping and/or distortion to render the watermark unreadable. This would render the picture unsharable, which is self-defeating because the original version if not revoked could still be shared and if revoked then this malformed copy does no harm. To distribute a photo that is currently revoked, a more sophisticated attacker could claim the picture (*i.e.,* register a copy with a ledger), mark it as not revoked, insert new metadata and a matching watermark (erasing the old one), and then start sharing it. IRS cannot prevent or detect this automatically, as it appears to be validly shared picture, but must rely on the aforementioned appeals process: the original owner would have to notice the copied picture, appeal to its ledger or site, and force it to be marked as permanently revoked.

Enabling Censorship? One might worry that government authorities could use their influence on owners or ledgers to force photos to be revoked. IRS cannot stop direct coercion, but nonprofit groups could create ledgers for specific types of photos; *e.g.,* that document human-rights violations or important information about government operations. These ledgers could register photos and not allow their revocation (and would deny the appeals process if it appeared the appeal was done under duress). We also expect the equivalent of the dark web to continue, where such materials could be distributed and reach news outlets and other venues that could report on their existence.

Malicious Ledgers? Ledgers could misbehave in various ways (*e.g.,* answering queries incorrectly, not responding to an owner's request to revoke or unrevoke a photo, etc.). One could force ledgers to provide cryptographic proof of the owner's intent about revocation, but it is very difficult to ensure that such an intent was the most recent interaction with the ledger. We expect that as in many other areas (*e.g.,* banking), it is almost impossible to scalably prevent bad behavior in the short-term but one counts on reputational effects (*i.e.,* users will avoid ledgers that are known to behave badly) to prevent bad behavior in the long term. In addition, the automated software that claims photos on behalf of owners could periodically send probes to ledgers to ensure that they are being answered correctly.

6 DISCUSSION

It seems obviously desirable for the Internet ecosystem to allow owners to express their intent about whether or not photos should be shared in a way that persists even when removed from its original context. This is not currently the case, and the sharing of photos against the wishes of the owner – while only a nuisance for some – can cause great harm for others (particularly young people coerced into doing so). We have proposed the IRS approach to significantly mitigate such sharing, by giving both viewers and sites the ability to respect the wishes of owners. IRS first involves an initial and only partially-scalable bootstrapping phase that could potentially be initiated by pro-privacy browser vendors, and then eventually a fully-scalable phase supported by the incumbent content aggregators who adopt it because it is now in their best interest (to achieve competitive advantage by embracing privacy and to avoid legal liability for showing photos that are clearly marked as having been revoked by their owners). If deployed, IRS would achieve its goals while preserving both privacy and performance, but would not prevent determined and malicious actors from seeing revoked photos and sharing with similarly-inclined actors.

IRS is an example of technology ecosystem transformation (TET). Note that TET is not the same as "disrupting an ecosystem" (as in [21]) where the goal is to unseat the incumbents. IRS, in contrast, is merely trying to change behavior. While TET does not necessarily involve novel technologies, it does involve a significant intellectual challenge: finding a transitional design that can (i) be deployed by a motivated group of first-movers and (ii) eventually change the incentives for the incumbents so that the desired changes are adopted more broadly.

We end by noting that TET in general, and IRS in particular, are not guaranteed to succeed, because the success of such a strategy depends on many factors outside our control and beyond what we can foresee. However, this is true for the deployment of almost all academic proposals. Our hope is that we as a research community embrace the use of TET-like approaches to bring about socially desirable changes.

REFERENCES

[1] Carlisle Adams, Pat Cain, Denis Pinkas, and Robert J. Zuccherato. 2001. Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP). *IETF Request for Comments (RFC)* 3161 (2001), 1–26. https://doi.org/10.17487/RFC3161

[2] Ali Al-Haj. 2007. Combined DWT-DCT digital image watermarking. *Journal of Computer Science* 3, 9 (2007), 740–746.

[3] Apple. 2021. iCloud Private Relay. (2021). https://developer.apple.com/videos/play/wwdc2021/10096/.

[4] Apple. 2021. Safely open apps on your Mac. (2021). https://support.apple.com/en-us/HT202491#view:~:text=Privacy%20protections.

[5] HTTP Archive. 2021. Web Almanac, First Contentful Paint, Figure 10.12. (2021). https://almanac.httparchive.org/en/2021/performance#first-contentful-paint-fcp.

[6] Mauro Barni, Franco Bartolini, and Alessandro Piva. 2002. Multichannel watermarking of color images. *IEEE Transactions on Circuits and Systems for Video Technology* 12, 3 (2002), 142–156.

[7] BBC 2018. Facebook wants your naked photos to stop revenge porn. BBC. (2018). https://www.bbc.com/news/newsbeat-44223809.

[8] Brave. 2021. Brave Talk. (2021). https://talk.brave.com/.

[9] Jehoshua Bruck, Jie Gao, and Anxiao Jiang. 2006. Weighted Bloom filter. In *Proceedings 2006 IEEE International Symposium on Information Theory, ISIT*. IEEE, The Westin Seattle, Seattle, Washington, USA, 2304–2308. https://doi.org/10.1109/ISIT.2006.261978

[10] C2PA. 2022. Coalition for Content Provenance and Authenticity. (2022). https://c2pa.org/.

[11] Josh Campbell and Jason Kravarik. 2022. A 17-year-old boy died by suicide hours after being scammed. The FBI says it's part of a troubling increase in 'sextortion' cases. *CNN* (2022). https://www.cnn.com/2022/05/20/us/ryan-last-suicide-sextortion-california/index.html https://www.cnn.com/2022/05/20/us/ryan-last-suicide-sextortion-california/index.html.

[12] DNSPerf. 2020. DNS Performance Analytics and Comparison. (2020). https://www.dnsperf.com/.

[13] Hany Farid. 2021. An Overview of Perceptual Hashing. *Journal of Online Trust and Safety* 1, 1 (2021).

[14] Drew Fudenberg and Jean Tirole. 1991. *Game Theory*. MIT Press, MIT, Cambridge, MA, USA.

[15] Thomas Mueller Graf and Daniel Lemire. 2020. Xor Filters: Faster and Smaller Than Bloom and Cuckoo Filters. *ACM Journal of Experimental Algorithmics* 25, Article 1.5 (March 2020), 16 pages. https://doi.org/10.1145/3376122

[16] Thomas Mueller Graf and Daniel Lemire. 2022. Binary Fuse Filters: Fast and Smaller Than Xor Filters. *ACM Journal of Experimental Algorithmics* 27, Article 1 (March 2022), 15 pages. https://doi.org/10.1145/3510449

[17] Caitlyn Gribbin. 2022. Revenge porn: Facebook teaming up with Government to stop nude photos ending up on Messenger, Instagram. ABC. (2022). https://www.abc.net.au/news/2017-11-02/facebook-offers-revenge-porn-solution/9112420.

[18] Baisa L. Gunjal and Suresh N.Mali. 2011. Secured color image watermarking technique in DWT-DCT domain. *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT)* 1, 3 (Aug. 2011), 36–44. http://airccse.org/journal/ijcseit/papers/0811ijcseit04.pdf

[19] JPEG. 2022. Exploration on JPEG Fake Media. (2022). https://jpeg.org/jpegfakemedia/index.html.

[20] Petros Maniatis, Devdatta Akhawe, Kevin Fall, Elaine Shi, and Dawn Song. 2011. Do You Know Where Your Data Are? Secure Data Capsules for Deployable Data Protection. In *HotOS XIII*. USENIX Association, Napa, CA, 5.

[21] Geoffrey A Moore. 2007. *Crossing the chasm: marketing and selling disruptive products to mainstream customers*. HarperCollins, New York City, NY.

[22] Mozilla. 2020. Mozilla VPN. (2020). https://www.mozilla.org/en-US/products/vpn/.

[23] Mozilla. 2022. Mozilla Policy Requirements for DNS over HTTPs Partners. (2022). https://wiki.mozilla.org/Security/DOH-resolver-policy.

[24] David-Octavio Muñoz-Ramirez, Volodymyr Ponomaryov, Rogelio Reyes-Reyes, Volodymyr Kyrychenko, Oleksandr Pechenin, and Alexander Totsky. 2018. A Robust Watermarking Scheme to JPEG Compression for Embedding a Color Watermark Into Digital Images. In *IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*. IEEE, Kyiv, UKraine, 619–624.

[25] Stefan Santesson, Michael Myers, Rich Ankney, Ambarish Malpani, Slava Galperin, and Carlisle Adams. 2013. X. 509 Internet Public Key Infrastructure Online Certificate Status Protocol-OCSP. *IETF Request for Comments (RFC)* 6960 (2013), 1–41. https://doi.org/10.17487/RFC6960

[26] Paul Schmitt, Anne Edmundson, Allison Mankin, and Nick Feamster. 2019. Oblivious DNS: Practical Privacy for DNS Queries. *Proc. Priv. Enhancing Technol.* 2019, 2 (2019), 228–244. https://doi.org/10.2478/popets-2019-0028

[27] Scott Shenker. 2022. Technology Ecosystem Transformation: Raising Our Sights and Looking at Bigger Problems Than We Are Used to Looking at. VMware RADIO 2022 (talk). (2022). https://drive.google.com/file/d/12JYLa1NiK_ArZnY5kgWUiXKPynPnKIKE/view.

[28] Milivoj Simeonovski, Fabian Bendun, Muhammad Rizwan Asghar, Michael Backes, Ninja Marnau, and Peter Druschel. 2015. Oblivion: Mitigating Privacy Leaks by Controlling the Discoverability of Online Information. In *Applied Cryptography and Network Security - 13th International Conference (ACNS), Revised Selected Papers (Lecture Notes in Computer Science)*, Vol. 9092. Springer, New York, NY, USA, 431–453. https://doi.org/10.1007/978-3-319-28166-7_21