The Decoupling Principle: A Practical Privacy Framework

Paul Schmitt
University of Hawai'i / INVISV

Jana Iyengar
Fastly

Christopher Wood
Cloudflare

Barath Raghavan
USC / INVISV

ABSTRACT

The three decade struggle to ensure Internet data confidentiality—a key aspect of communications privacy—is finally behind us. Encryption is fast, secure, and standard in all browsers, modern transports, and major protocols. Yet it has long seemed that network privacy is not unified by core principles but a grab bag of techniques and ideas applied to an equally wide range of applications, contexts, layers of infrastructure, and software stacks.

Here we attempt to distill a principle—one that is old but seldom discussed as such—for building privacy into Internet services. We explore what privacy properties are desirable and achievable when we apply this principle. We evaluate several classic systems and ones that have been recently deployed with this principle applied, and discuss future directions for network privacy building upon these efforts.

CCS CONCEPTS

• **Security and privacy** → **Pseudonymity, anonymity and untraceability**; **Privacy-preserving protocols**;

KEYWORDS

Internet privacy, anonymity, system architectures

ACM Reference Format:
Paul Schmitt, Jana Iyengar, Christopher Wood, and Barath Raghavan. 2022. The Decoupling Principle: A Practical Privacy Framework. In *The 21st ACM Workshop on Hot Topics in Networks (HotNets '22), November 14–15, 2022, Austin, TX, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3563766.3564112

HotNets '22, November 14–15, 2022, Austin, TX, USA

ACM ISBN 978-1-4503-9899-2/22/11…$15.00
https://doi.org/10.1145/3563766.3564112

1 INTRODUCTION

For the first time in human history, nearly every person is under daily surveillance—surveillance not in spite of, but *because of*, the accomplishments of the networking community. Privacy violations are a multi-billion dollar industry, and have for some time now been a core business model of the Internet [33, 40]. People require privacy in their daily lives, but privacy matters beyond the individual: societies progress when we prevent the chilling effects of total surveillance [15, 24, 25, 31, 34]. Individual *privacy* is synonymous with organizational *security*: in each case, the parties involved wish to maintain control over their private data and metadata.

Thankfully, practitioners and researchers alike have recognized the need for, at minimum, data confidentiality. TLS is used for nearly all types of communications in the Internet, and is the default in all major browsers, modern protocols like QUIC [19, 22] and HTTP/3 [3], and much more. Despite TLS's success, Internet communications are nonetheless more heavily surveilled today than ever before, both in the network and at the endpoints. While data is encrypted in flight, significant metadata is typically leaked in transit (e.g., IP addresses, DNS messages, etc.) and at the endpoints (by endpoints themselves and their partner organizations). While for decades the research community, along with numerous scattered deployments, have tried to address communications metadata privacy, reusable design patterns for addressing this problem are notably absent from the protocol designer's toolbox.

In this paper, we call attention to what we call the *Decoupling Principle*. The idea is simple, yet previously not clearly articulated: to ensure privacy, information should be divided architecturally and institutionally such that each entity has only the information they need to perform their relevant function. Architectural decoupling entails splitting functionality for different fundamental actions in a system, such as decoupling authentication (proving who is allowed to use the network) from connectivity (establishing session state for communicating). Institutional decoupling entails splitting what information remains between non-colluding entities, such as distinct companies or network operators, or between a user and network peers. This decoupling makes service providers individually breach-proof, as they each have little

or no sensitive data that can be lost to hackers. Put simply, the Decoupling Principle suggests always separating *who you are* from *what you do*.

Chaum was one of the first to design privacy protocols and systems in this manner, in a series of foundational papers [4–6]. Many systems have built upon Chaum's insights, including some of the most popular privacy systems ever built, such as Tor [13]. However, due to rising pressure to improve Internet privacy for end-users, only in the last decade have Chaum's ideas begun to see widespread application and adoption.

Some prior approaches have failed to heed the Decoupling Principle. For example, VPNs and middleboxes shift trust from a diffuse set of network endpoints (e.g., websites a user might visit, DNS resolvers a user might use, etc.) to a single trusted intermediary (e.g., a VPN provider). Depending on the threat model, this design may address the privacy concerns of end-users, especially if the network is even more untrustworthy. However, here the single trusted intermediary sees all user activity bundled together with user identity, requires more trust than is necessary, and is susceptible to data breaches. This pattern does not adhere to the Decoupling Principle. Examples such as these lend credence to the idea that decoupling is fundamental to network privacy.

Next we discuss some common privacy goals and the ways in which those are achieved, and then consider numerous systems designed to achieve those goals. Some are classic designs due to Chaum and others that are the bedrock upon which we build today. Others include recently-deployed commercial systems to achieve meaningful (though incremental) privacy gains in production networks. We also consider some pitfalls where decoupling was either ignored or proved insufficient to meet the challenge. Finally, we discuss a number of remaining challenges in Internet privacy.

2 PRELIMINARIES

2.1 What is Internet Privacy?

Privacy is being free from observation, and nowhere is this more important than in the Internet, where we must rely upon others to carry our traffic. Since data confidentiality is, thankfully, largely solved, privacy challenges have moved elsewhere: to metadata of traffic (rather than the now-encrypted payloads) and to the endpoints where application-level processing occurs. In addition, privacy challenges abound in ensuring unlinkability between multiple streams of traffic from a single user/entity (in the network) and multiple identifiers (at the endpoints).

Privacy challenges exist across the network stack, and so privacy solutions must also be layered. For example, encrypting application traffic can provide confidentiality of message content, yet unprivileged observers of lower layers (e.g., IP routing infrastructure) can readily observe who is talking to whom by recording IP endpoints. Systems that adhere to the Decoupling Principle must consider privacy holistically, and take into account leakage of information across the stack.

2.2 Authentication, Authorization, and Actors

Privacy interacts with security mechanisms in important ways. As network security has grown in importance, more systems rely upon authentication to confirm the identity of a user or device and authorization to confirm the levels of access that should be conferred. But authentication and authorization, both real-time and for later forensic use, often create a non-repudiable record of who used a network service when, how, and even why. The actors involved are simultaneously decentralized—with authentication and authorization used from the most security-critical applications to low-risk contexts—and centralized (such as OAuth and SSO) with a view into the uses of a huge range of services.

2.3 Trust

Privacy hinges on trust that users must place in the Internet systems with which they interact. When we use systems we place our privacy in their hands. In the past 15 years, the Internet has become increasingly centralized with the majority of traffic being attributable to a handful of cloud providers, CDNs, and content providers deemed hypergiants [21]. For instance, the number of ASNs required to make up 50% of Internet traffic decreased from 150 in 2009 [21] to only 5 in 2019 [27, 38]. This trend has resulted in the unprecedented centralization of trust, and knowledge of users' behavior, into these organizations. This centralization has come with some upsides for users, as large organizations are sometimes capable of securing user data effectively, but this comes with distinct costs and consequences as well [23, 32].

Most networking protocols assume end-to-end coordination and thus end-to-end trust. Baked into this assumption is a separate reliance on authentication mechanisms that ensure that a source is certain of the destination it is communicating with (e.g., using certificate hierarchies or other out-of-band mechanisms). Users often implicitly or explicitly make judgments about whether a particular piece of data should be revealed to a particular service in a particular context, and this judgement requires unenumerable factors that only the user can consider. The key is that the parties involved in the communication can and should have access to data and metadata, and their mutual trust in the intermediaries is the key question we examine in this paper.

2.4 The Decoupling Principle

Earlier we stated the Decoupling Principle concisely as *decouple who you are from what you do*. To make this more

concrete so as to enable analysis, we define ▲ as a sensitive user identity *known by* some entity and likewise △ as a non-sensitive user identity, ● as sensitive data, and ⊙ as non-sensitive data.[1] We define tuples of two or more members, typically with one or more user identity and one or more aggregate of user data, where a tuple defines the knowledge of some entity. A decoupling analysis consists of examining the parties (independent entities or actors) involved in a networked system that interacts with the user or their data. A system is decoupled, and thus benefiting from the privacy gained by applying the Decoupling Principle, if *only* the user is (▲, ●). Other entities may have at most one of ▲ or ●, with all other tuple entries as △ or ⊙.

3 SYSTEMS

We now discuss classic and recent systems that employ the Decoupling Principle. We also discuss some cautionary tales: systems that do not employ the Decoupling Principle and consequently rely on trust in a third-party for user privacy.[2]

3.1 Classic Systems

3.1.1 Access and Authentication. The foundational work in anonymous access and authentication systems is Chaum's blind signatures [4, 5]. When using blind signatures, the content of a message is blinded before it is sent to be signed, typically by a trusted signing authority. As the message is blinded, the signing authority cannot access the message content, but the signing authority's signature can later be verified by a third party that has access to the unblinded content. Blind signatures offer unlinkability in that the signing authority is unable to link a blind-signed message to a prior interaction that produced the message.

Blind signatures provide a straightforward example of the Decoupling Principle in that they allow users to decouple their identity from their actions. In the digital currency case, participants' purchases cannot be linked to identities. In the scheme, neither the seller nor the bank are able to know the identity of the buyer, but simply know that the money being presented is valid. Using the notation introduced in 2.4, the decoupling analysis for the digital currency example is:

Buyer	Signer (Bank)	Verifier (Bank)	Seller
(▲, ●)	(▲, ⊙)	(△, ⊙/●)	(△, ●)

In this example, the Signer and the Verifier are the same entity, but the use of blind signatures enforces decoupling by ensuring that the two actions and the user's identity cannot be linked. It is also possible, but not necessary, to separate the Signer and the Verifier across two distinct organizations.

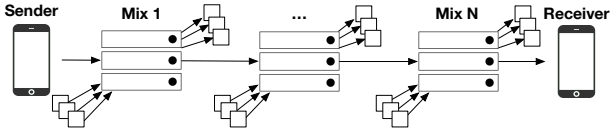

Figure 1: Mix-net decoupling

3.1.2 Actors. Chaum also introduced the first architecture for anonymous communication over the Internet in his classic mix-net paper [6]. This approach introduced the notion of multi-hop relaying across mutually-non-cooperating entities. A message is encrypted using the mix's public key before being sent. The mix decrypts using its private key and forwards to the receiver or to another mix. This basic arrangement is shown in Figure 1. Chaum's design thwarted timing attacks by batch forwarding. Mix-nets offer multiple forms of metadata privacy: 1) sender anonymity: the receiver of a message does not know the sender's identity; and 2) sender and receiver anonymity to third party observers: senders and receivers can exchange messages while non-global observers are unable to determine that the two and communicating with one another. Observers are only able to know that a given sender or receiver is communicating using a mix-net.

Mix-nets were later adapted by Syverson *et al.* for real-time Internet communications in their work on Onion Routing [36], and later improved in the popularly-deployed Tor system [13]. These systems provide metadata privacy through decoupling. Here, identities (i.e., sender and receiver endpoints) are decoupled from their behavior of having a conversation (i.e., metadata surrounding messages or traffic), up to the limits of what is feasible to reconstruct or infer from traffic analysis and other side-channel attack vectors. The decoupling analysis for mix-nets is as follows:

Sender	Mix 1	...	Mix N	Receiver
(▲, ●)	(▲, ⊙)	...	(△, ⊙)	(△, ●)

3.2 Recent Systems

3.2.1 Privacy Pass.

Client	Issuer	Origin
(▲, ●)	(▲, ⊙)	(△, ●)

Internet users making use of privacy-enhancing systems like Tor often faced many challenges from websites asking them to prove that they are legitimate users and not bots. Ideally, only users of legitimate clients can successfully respond to these challenges. Unfortunately, such challenges are often privacy-unfriendly, e.g., they require application-layer authentication prompts or tracking cookies. Such techniques allow

[1] Of course it is in reality impossible to neatly categorize user identities or data as sensitive or non-sensitive, especially as the amount and dimensionality of data being considered increases. For now we will treat these as generally-understood categories to which we will add shades of gray later.

[2] We call them cautionary tales rather than failed systems because they can still be useful, but use of these systems cannot rely on the architectural properties of the system alone for achieving user privacy in threat models without trusted third parties.

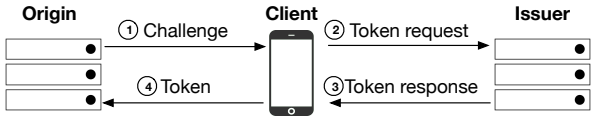

Figure 2: Privacy Pass decoupling

the service to learn high-fidelity information about a client or track them over time.

Privacy Pass [11, 12] addresses this issue by applying the Decoupling Principle to separate privacy-sensitive authentication from authorization. In particular, clients that are challenged to present proof respond with low-fidelity *tokens* produced by a trusted issuer. The issuer, in turn, only presents tokens to clients that are able to successfully prove that they are legitimate. This interaction is shown in Figure 2. Thus, tokens *transfer* trust from the issuer, which learns privacy-sensitive information from the client but nothing of the service (or origin)[3], to the service (or origin), which learns only non-sensitive user information.

3.2.2 Oblivious DNS.

Client	Resolver	Oblivious Resolver	Origin
(▲, ●)	(▲, ⊙)	(△, ⊙/●)	(△, ●)

Nearly all Internet connections are preceded by DNS lookups. As such, recursive DNS resolvers, typically run by ISPs or cloud providers, are able to tie browsing behavior (DNS queries) to individual users (IP addresses and/or application-layer identifiers). Prior work has demonstrated that DNS traffic can reveal users' website usage even when connecting through Tor [16]. To reduce the available information in the DNS hierarchy, Oblivious DNS protocols (ODNS [29] and ODoH [35]) apply the Decoupling Principle by separating knowledge across organizations.

The original ODNS protocol [29] encrypts and obfuscates queries that are sent to the user's recursive DNS provider. The obfuscated queries then reach an oblivious resolver, a server that has been configured to be the authoritative server for the obfuscated queries, and holds the encryption keys needed to decrypt the original query. This server then acts as a recursive resolver for the plaintext query. The end result is the users' regular recursive resolver can learn the users' identities (▲), but cannot observe their DNS queries (⊙), while the oblivious resolver is able to see queries (●) but not users' identities (△). If the recursive resolver and the oblivous resolver are run by two non-colluding organizations, user privacy is maintained.

Oblivious DNS over HTTPS (ODoH) [35] was inspired by ODNS and decouples information about DNS queries by using an Oblivious Proxy (recursive resolver in ODNS) to handle HTTP requests that contain encrypted DNS queries that are sent to an Oblivious Target (oblivious resolver in ODNS), which is a DNS over HTTPS resolver. As with ODNS, user privacy is maintained as long as the Oblivious Proxy and Oblivious Target are run by non-colluding organizations.

3.2.3 Pretty Good Phone Privacy.

User	PGPP-GW	NGC
($▲_H$, $▲_N$, ●)	($▲_H$, $△_N$, ⊙)	($△_H$, $△_N$, ●)

Pretty Good Phone Privacy (PGPP) [30] leverages the Decoupling Principle to achieve location anonymity in the cellular architecture. Traditionally, the cellular architecture relies on the International Mobile Subscriber Identity (IMSI), a permanent, globally-unique identifier that is stored on a SIM card for both billing and authentication functionality as well as mobility and connectivity. As billing and authentication effectively creates a binding between the IMSI and a user's identity, their subsequent usage and physical movements can easily be tracked (and sold [8, 9, 20, 39]) simply as a result of operating a cellular network.

PGPP decouples billing and authentication from the cellular core (the NGC), altering it to use an over-the-top oblivious authentication protocol to an external server, the PGPP-GW, that can be operated by a second organization, while leaving mobility and connectivity functions in the core as they are today. By shifting billing (and the user's human identity $▲_H$) and authentication, IMSIs are altered, which we denote as the non-sensitive network identity $△_N$ which are identical or shuffled periodically. This ensures unlinkability to individual users as they connect and move through the network.

Here, the decomposition of ▲ into $▲_H$ and $▲_N$ illustrates how different components of user data can be visible to system entities, and can still be analyzed in our framework.

3.2.4 Multi-Party Relays.

User	Relay 1	Relay 2	Origin
(▲, ●)	(▲, ⊙)	(△, ⊙/●)	(△, ●)

In 2021, Apple launched the iCloud Private Relay service [1], which employs a proxy architecture akin to Chaum's classic mix-net and subsequent systems like Tor; we term these as Multi-Party Relay (MPR) services. Private Relay differs from prior systems in two key respects: 1) by employing HTTP instead of custom protocols and 2) by using well-provisioned, commercial network infrastructure with just two hops rather than a multi-hop, volunteer network of decentralized nodes. The service uses a proxy architecture with two nested HTTP CONNECT tunnels from the client, the first to the first relay (run by Apple) and the second via the first to a second relay (run by one of three independent infrastructure providers). The second relay issues a connection to the origin server on behalf of the user.

With an MPR service, a user's identity (their network-layer identifier) is known to Relay 1, but their request (the data) is

[3]We use *origin* and *target server* or *service* interchangeably in this paper to mean, for example, a web server from which a client requests content.

not known as it is hidden in an encrypted stream. Relay 2 is not aware of the user except as an anonymous member of a network aggregate, but may learn limited information about the user's request (such as the FQDN of the origin server). Finally, the Origin only learns of the user's request.

3.2.5 *Private Aggregate Statistics.*

Client	Aggregator	Collector
(▲, ●)	(▲, ⊙)	(△, ⊙)

Applications ranging from software telemetry to infectious disease tracking and reporting need to aggregate statistics. One naive approach is to send inputs to a single (trusted) server that computes the aggregate. This is non-private, however, since the single server sees sensitive client data along with their identity.

One approach is to hide sensitive client identifying information from the server using Oblivious HTTP, a generalization of ODoH; clients would send encrypted reports to the collection server through a proxy, thereby decoupling the client's network identity (IP address) from its individual contribution. While this improves the overall privacy posture of the system, it still reveals more than necessary to the relevant parties. In particular, the single server—acting both as aggregator and collector of data—sees all individual data elements.

Further application of the Decoupling Principle can improve the situation. In particular, Privacy-Preserving Measurement (PPM) [14] is a recent effort in the IETF to standardize protocols for privately computing aggregate statistics, and Prio [7] is one concrete instance of the PPM protocol. PPM uses multi-party computation between non-colluding entities to privately compute an aggregate output. In this arrangement, only the client sees sensitive data, whereas other parties in the system only see the aggregate (non-sensitive) output computed from many client inputs.

3.3 Cautionary Tales

Client	VPN Server	Origin
(▲, ●)	(▲, ●)	(△, ●)

Beyond systems that have key privacy weaknesses (e.g., due to no focus on privacy), there are many examples of systems that aim to protect privacy but create new vulnerabilities and new points of surveillance. Such systems often fail to decouple sensitive information and thus offer privacy only under the assumption of trust in some network entity.

Classic examples include centralized VPN and security processing services. The purposes of both VPNs (e.g., point-to-point and perimeter defense) and security processing services (e.g., phishing prevention middleboxes) are often distinct from the privacy goals we discuss in this paper. Nevertheless, by funneling all traffic through a single trusted party, such systems create a single locus of observation.

TLS Encrypted ClientHello (ECH) [28] is another example of a protocol that falls short of fully applying the Decoupling Principle. With ECH, TLS clients encrypt sensitive information in the TLS handshake with the TLS server which terminates the connection. This has the effect of hiding sensitive information from the untrusted network. However, ECH does not alter what information the TLS server sees.

4 DISCUSSION

This section discusses fundamental assumptions and relevant considerations that enable reasonable applications of the Decoupling Principle in practice. It also discusses the impact of the Decoupling Principle on real world systems.

4.1 Non-Collusion

Systems that adhere to the Decoupling Principle often rely on the assumption that multiple organizations will not collude against a user. In this arrangement, active coupling requires active collusion between participants. Certainly, an ideal system design would not require such an assumption. However, such ideal instances of privacy-enhancing protocols are difficult in practice given the implicit trust inherent in complex computing systems of all kinds, a problem as old as Thompson's classic attack (and likely much older) [37]. All users, even the most sophisticated, rely on services offered by a relative few, necessarily giving those services insight into users' identities and behaviors. Further, practical, immediately-deployable solutions can offer significant privacy gains with a slightly relaxed set of trust requirements. Privacy is and will remain a moving target. As such, we are well served to take advantage of incremental privacy gains as they present themselves.

4.2 Degrees of Decoupling

As demonstrated by Private Relay and Private Aggregate Statistics, the *degree* to which information is decoupled can improve the privacy posture of the system. For example, adding more relays to Private Relay may improve the system against timing or collusion attacks. Indeed, Tor embodies this approach by allowing for circuits of 3 or more hops, albeit at greater performance cost. Likewise, adding more aggregators to PPM may help against collusion attacks. In practice, decoupling eventually reaches a point where it offers limited return in privacy at great cost. Adding more hops in Private Relay and aggregators in PPM adds overhead to the system and ultimately reduces performance. Ultimately, any system based on the Decoupling Principle should consider cost / benefit tradeoffs with regard to the degree of decoupling.

4.3 Deployment Considerations

While the Decoupling Principle applies primarily to protocol design and system architecture, it can be limited by practical

implementation issues. For example, consider traffic analysis attacks in the context of mix-net systems like Tor. Encryption protects the confidentiality of data, but it does not protect against other attributes of application data such as the size and timestamps of data while in transit. Specific systems like Tor go to great lengths to mitigate these types of attacks, including via use of constant-size packets and adding additional chaff to make traffic analysis more difficult in practice. These types of enhancements come at a cost, however, as they decrease overall system performance and increase protocol complexity. These types of tradeoffs are well-known in the landscape of privacy-enhancing technologies [10].

Recent commodity CPUs and security chips enable hardware support for bootstrapping Trusted Execution Environments (TEEs). TEEs enable a user to have processing done securely and privately on their behalf on hardware they do not own or directly control. Typically, such hardware can cryptographically attest to the software running in the TEE (thereby ensuring the authenticity of the software), ensure that the memory and execution stream are encrypted for only the TEE to read, and provide some degree of tamper-proofing against local and remote attacks. A TEE moves the locus of trust in which the software runs to the hardware manufacturer, carrying an implicit promise that a hardware vendor is unlikely to target a specific user in an unknown cloud because they likely have no direct incentive to do so. As such, TEEs are a reasonable mechanisms for enabling decoupling in practice. Indeed, CACTI [26]—CAPTCHA Avoidance via Client-Side TEE Integration—is a system similar to Privacy Pass that uses TEEs for the purposes of keeping private state. Similarly, Phoenix [17] uses TEEs to implement CDN-like services (e.g., caching, web application firewalls, etc.) without the CDN seeing any sensitive data.

4.4 Real World Regressions

Currently deployed systems often require user metadata to function correctly, and decoupling the user's identity from their actions can either subvert or break these systems. For instance, video streaming systems enforce DRM (Digital Rights Management) based on a user's approximate location by geolocating the user's IP address, which is obfuscated in systems such as Private Relay. To allow these systems to continue operating correctly, some amount of user metadata is required to be visible to the Origin server. While sharing of this metadata can be done in a privacy-preserving manner, as is done in Private Relay, it violates the Decoupling Principle.

More broadly, systems that employ the Decoupling Principle empower users and, consequently, can reduce control at entities that previously had access to privileged information. Control is not always employed destructively however; network operators often rely on user information to manage their networks, which ultimately serves the user.

5 TOWARD A MORE PRIVATE INTERNET

While the last few years have seen substantial progress in the deployment of privacy-preserving technologies such as those we discussed earlier, much more work remains.

5.1 Architectural Decoupling

The idea of non-colluding entities is an old one in security: it's often the case that an attack model will assume that some entities do not share certain private information or otherwise collude, enabling security or privacy guarantees. The decoupling of entities responsible for network traffic relies upon something similar. However there are legal considerations as well: when a network provider or cloud service only sees part of a network connection, by its very nature that organization cannot reveal the parts it cannot see, and thus it has more than mere plausible deniability. Service providers cannot gain access to decoupled information without illegally colluding with one another (and likely changing software to do so), providing stronger protections to the users of their services.

Non-collusion can be more effective as a system property if a user can dynamically stitch services or stripe usage across multiple providers. For instance, a user can improve DNS privacy by distributing their queries across multiple resolvers, thereby limiting the information available about a given user at each [18]. Future service architectures such as EI [2], which envisions multiple entities on the Internet offering composable services, can further enable dynamic tailoring and construction of decoupled systems.

5.2 Future Directions

There remain many networked systems that can benefit from decoupling, and such work (including privacy-first designs of systems that solve new problems presented by decoupled systems) can and should continue. However, the Decoupling Principle is not a panacea for all user privacy issues. What it does ensure is that common violations of user privacy require subverting the principle itself. For instance, a government can require all relays in Private Relay to collude to get sensitive user information, but doing so forces the system to violate the Decoupling Principle.

Ultimately, the value of decoupling is that it shifts privacy violations to public, legal, or social spaces, and away from the technical design of the system, which we claim is the appropriate space for such conversations.

ACKNOWLEDGMENTS

Many thanks to Tommy Pauly and the anonymous reviewers.

REFERENCES

[1] Apple. 2021. iCloud Private Relay Overview. https://www.apple.com/privacy/docs/iCloud_Private_Relay_Overview_Dec2021.PDF. (Dec. 2021).

[2] Hari Balakrishnan, Sujata Banerjee, Israel Cidon, David Culler, Deborah Estrin, Ethan Katz-Bassett, Arvind Krishnamurthy, Murphy McCauley, Nick McKeown, Aurojit Panda, Sylvia Ratnasamy, Jennifer Rexford, Michael Schapira, Scott Shenker, Ion Stoica, David Tennenhouse, Amin Vahdat, and Ellen Zegura. 2021. Revitalizing the Public Internet by Making It Extensible. *SIGCOMM Comput. Commun. Rev.* 51, 2 (May 2021), 18–24.

[3] M. Bishop. 2022. HTTP/3. *Internet Engineering Task Force, Proposed RFC 9114* (2022).

[4] David Chaum. 1983. Blind signatures for untraceable payments. In *Advances in cryptology*. Springer, 199–203.

[5] David Chaum. 1984. Blind signature system. In *Advances in cryptology*. Springer, 153–153.

[6] David L Chaum. 1981. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM* 24, 2 (1981), 84–90.

[7] Henry Corrigan-Gibbs and Dan Boneh. 2017. Prio: Private, robust, and scalable computation of aggregate statistics. In *14th USENIX symposium on networked systems design and implementation (NSDI 17)*. 259–282.

[8] Joseph Cox. 2019. I Gave a Bounty Hunter $300. Then He Located Our Phone. https://motherboard.vice.com/en_us/article/nepxbz/i-gave-a-bounty-hunter-300-dollars-located-phone-microbilt-zumigo-tmobile. (Jan. 2019).

[9] Joseph Cox. 2019. Stalkers and Debt Collectors Impersonate Cops to Trick Big Telecom Into Giving Them Cell Phone Location Data. https://www.vice.com/en_us/article/panvkz/stalkers-debt-collectors-bounty-hunters-impersonate-cops-phone-location-data. (March 2019).

[10] Debajyoti Das, Sebastian Meiser, Esfandiar Mohammadi, and Aniket Kate. 2018. Anonymity trilemma: Strong anonymity, low bandwidth overhead, low latency-choose two. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 108–126.

[11] Alex Davidson, Ian Goldberg, Nick Sullivan, George Tankersley, and Filippo Valsorda. 2018. Privacy Pass: Bypassing Internet Challenges Anonymously. *Proceedings on Privacy Enhancing Technologies* 2018 (06 2018), 164–180.

[12] A. Davidson, J. Iyengar, and C. A. Wood. 2022. *Privacy Pass Architectural Framework*. Internet-Draft. Internet Engineering Task Force. https://www.ietf.org/archive/id/draft-ietf-privacypass-architecture-03.html Work in Progress.

[13] R Dingledine, N Mathewson, and P Syverson. 2004. Tor: the second-generation onion router', USENIX Security Symposium. (2004).

[14] Tim Geoghegan, Christopher Patton, Eric Rescorla, and Christopher A. Wood. 2022. *Distributed Aggregation Protocol for Privacy Preserving Measurement*. Internet-Draft draft-ietf-ppm-dap-00. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/draft-ietf-ppm-dap-00 Work in Progress.

[15] John Gilliom. 2001. *Overseers of the poor: Surveillance, resistance, and the limits of privacy*. University of Chicago Press.

[16] Benjamin Greschbach, Tobias Pulls, Laura M. Roberts, Philipp Winter, and Nick Feamster. 2017. The Effect of DNS on Tor's Anonymity. In *Network and Distributed System Security Symposium, NDSS*. San Diego, CA.

[17] Stephen Herwig, Christina Garman, and Dave Levin. 2020. Achieving Keyless {CDNs} with Conclaves. In *29th USENIX Security Symposium (USENIX Security 20)*. 735–751.

[18] Austin Hounsel, Paul Schmitt, Kevin Borgolte, and Nick Feamster. 2021. Encryption without Centralization: Distributing DNS Queries across Recursive Resolvers. In *Proceedings of the Applied Networking Research Workshop (ANRW '21)*.

[19] Jana Iyengar and Martin Thomson. 2021. QUIC: A UDP-based multiplexed and secure transport. *Internet Engineering Task Force, RFC 9000* (2021).

[20] Kate Kaye. 2015. The $24 Billion Data Business That Telcos Don't Want to Talk About. https://adage.com/article/datadriven-marketing/24-billion-data-business-telcos-discuss/301058/?mod=article_inline. (26 Oct. 2015).

[21] Craig Labovitz, Scott Iekel-Johnson, Danny McPherson, Jon Oberheide, and Farnam Jahanian. 2010. Internet Inter-Domain Traffic. In *SIGCOMM 2010*. New Delhi, India.

[22] Adam Langley, Alistair Riddoch, Alyssa Wilk, Antonio Vicente, Charles Krasic, Dan Zhang, Fan Yang, Fedor Kouranov, Ian Swett, Janardhan Iyengar, et al. 2017. The quic transport protocol: Design and internet-scale deployment. In *Proceedings of ACM SIGCOMM*.

[23] Tai Liu, Zain Tariq, Jay Chen, and Barath Raghavan. 2017. The barriers to overthrowing internet feudalism. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*. 72–79.

[24] Rebecca MacKinnon. 2013. *Consent of the Networked: The Worldwide Struggle For Internet Freedom*. Basic Books (AZ).

[25] Robert W McChesney. 2013. *Digital disconnect: How capitalism is turning the Internet against democracy*. New Press, The.

[26] Yoshimichi Nakatsuka, Ercan Ozturk, Andrew Paverd, and Gene Tsudik. 2021. {CACTI}: Captcha Avoidance via Client-side {TEE} Integration. In *30th USENIX Security Symposium (USENIX Security 21)*. 2561–2578.

[27] Enric Pujol, Ingmar Poese, Johannes Zerwas, Georgios Smaragdakis, and Anja Feldmann. 2019. Steering Hyper-Giants' Traffic at Scale. In *CoNEXT 2019*. Orlando, FL.

[28] Eric Rescorla, Kazuho Oku, Nick Sullivan, and Christopher A. Wood. 2022. *TLS Encrypted Client Hello*. Internet-Draft draft-ietf-tls-esni-14. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/draft-ietf-tls-esni-14 Work in Progress.

[29] Paul Schmitt, Anne Edmundson, Allison Mankin, and Nick Feamster. 2019. Oblivious DNS: Practical Privacy for DNS Queries. *Proceedings on Privacy Enhancing Technologies* 2019 (04 2019), 228–244.

[30] Paul Schmitt and Barath Raghavan. 2021. Pretty Good Phone Privacy. In *USENIX Security 2021*. virtual.

[31] Bruce Schneier. 2012. *Liars and outliers: enabling the trust that society needs to thrive*. John Wiley & Sons.

[32] Bruce Schneier. 2012. When it comes to security, we're back to feudalism. *Schneier on Security* (2012).

[33] Bruce Schneier. 2015. *Data and Goliath: The hidden battles to collect your data and control your world*. WW Norton & Company.

[34] Bruce Schneier. 2018. Surveillance Kills Freedom By Killing Experimentation. https://www.wired.com/story/mcsweeneys-excerpt-the-right-to-experiment/. (Nov. 2018).

[35] Sudheesh Singanamalla, Suphanat Chunhapanya, Jonathan Hoyland, Marek Vavruša, Tanya Verma, Peter Wu, Marwan Fayed, Kurtis Heimerl, Nick Sullivan, and Christopher Wood. 2021. Oblivious DNS over HTTPS (ODoH): A Practical Privacy Enhancement to DNS. *Proceedings on Privacy Enhancing Technologies* 2021 (10 2021), 575–592.

[36] Paul F Syverson, David M Goldschlag, and Michael G Reed. 1997. Anonymous connections and onion routing. In *Proceedings. 1997 IEEE Symposium on Security and Privacy (Cat. No. 97CB36097)*. IEEE, 44–54.

[37] Ken Thompson. 1984. Reflections on trusting trust. *Commun. ACM* 27, 8 (1984), 761–763.

[38] Martino Trevisan, Danilo Giordano, Idilio Drago, Marco Mellia, and Maurizio Munafo. 2018. Five Years at the Edge: Watching Internet from the ISP Network. In *CoNEXT 2018*. Heraklion, Greece.

[39] Zack Whittaker. 2018. US Cell Carriers are Selling Access to Your Real-Time Phone Location Data. https://www.zdnet.com/article/us-cell-carriers-selling-access-to-real-time-location-data/. (14 May 2018).

[40] Shoshana Zuboff. 2015. Big other: surveillance capitalism and the prospects of an information civilization. *Journal of information technology* 30, 1 (2015), 75–89.