The Right to Be Zero-Knowledge Forgotten

Ivan Visconti
visconti@unisa.it
University of Salerno
Fisciano, (SA), Italy

ABSTRACT

The main goal of the EU GDPR is to protect personal data of individuals within the EU. This is expressed in several rights and, among them, in this work we focus on the Right to Erasure, more commonly known as the Right to Be Forgotten (RtBF).

There is an intriguing debate about the affordable costs and the actual technical feasibility of satisfying the RtBF in digital platforms. We note that some digital platforms process personal data in order to derive and store correlated data raising two main issues: 1) removing personal data could create inconsistencies in the remaining correlated data; 2) correlated data could also be personal data. As such, in some cases, erasing personal data can trigger an avalanche on the remaining information stored in the platform.

Addressing the above issues can be very challenging in particular when a digital platform has been originally built without embedding in its design specific methodologies to deal with the RtBF.

This work aims at illustrating concrete scenarios where the RtBF is technically hard to guarantee with traditional techniques. On the positive side, we show how *zero-knowledge* (ZK) proofs can be leveraged to design affordable solutions in various use cases, especially when considered at design time. ZK proofs can be instrumental for compliance to the RtBF revolutionizing the current approaches to design compliant systems. Concretely, we show an assessment scheme allowing to check compliance with the RtBF leveraging the power of ZK proofs. We analyze the above assessment scheme considering specific hard-to-address use cases.

CCS CONCEPTS

• **Security and privacy** → **Data anonymization and sanitization**; **Privacy-preserving protocols**; • **Theory of computation** → *Interactive proof systems*; • **Applied computing** → Law.

KEYWORDS

Right to Be Forgotten, Zero Knowledge Proofs, Security By Design

ACM Reference Format:
Ivan Visconti. 2024. The Right to Be Zero-Knowledge Forgotten. In *The 19th International Conference on Availability, Reliability and Security (ARES 2024), July 30–August 02, 2024, Vienna, Austria.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3664476.3669973

ARES 2024, July 30–August 02, 2024, Vienna, Austria

ACM ISBN 979-8-4007-1718-5/24/07
https://doi.org/10.1145/3664476.3669973

1 INTRODUCTION

Typically, users provide personal data to digital platforms with the goal of obtaining some desired services. Modern digital platforms involve advanced computations to build and maintain sophisticated data structures, machine learning models and resilience mechanisms (e.g., decentralized ledgers) that embed personal data. Obviously, users would like to receive services from digital platforms with high degrees of usability, efficiency and security. In contrast, for business reasons, digital platforms are usually designed targeting a revenue as major goal, and thus the resulting quality of service does not always match users' expectations.

Compliance to the GDPR. Digital platforms dealing with personal data of individuals within the EU are required to comply to the EU General Data Protection Regulation's (GDPR) [16]. Traditional cryptographic tools seem insufficient to enforce GDPR compliance, and recently (e.g., see in [23]), the use of advanced cryptographic tools like secure multi-party computation [20] has been investigated to mitigate the tension generated by contrasting requirements.

In some scenarios, one of the most challenging requirements imposed by the GDPR is the so-called "Right to Be Forgotten" (RtBF, for short). Indeed, depending on the specific design adopted to build a digital platform, erasing some data from the platform could have an avalanche effect on the validity of several other correlated data stored in the same platform.

In general digital platforms collect personal data for various purposes. In some cases personal data are just stored and retrieved upon request. In other cases they are used as inputs to computations producing other correlated data that in turn can be stored and used in subsequent computations. For various reasons (e.g., efficiency, integrity, accountability) the utility of such computed data still could depend on the existence (implicit or explicit) of the initial data that was used as input for their computation. Moreover, data obtained as outputs of computations that were performed using as input personal data could in turn be, to some extent, also considered personal data. Therefore, in some unfortunate cases, the need to remove a single information could correspond to pushing forward a domino piece generating a very significant chain reaction with a large amount of information that needs to be deleted to restore consistency.

The tension between users and digital platforms. While users would like personal data to be quickly and effectively removed upon request, digital platforms might actually try to avoid or at least delay such procedures for various reasons. First of all, the success of the business of a digital platform might crucially rely on personal data stored in it, and thus there is an evident interest of the platform manager in minimizing the amount of data removal requests that are accepted and implemented, in contrast with the interests of users desiring to be forgotten. A classical well-known

example involves Google LLC refusing to de-index pages associated to users, and the interested reader can refer to [12].

Additionally, removing personal data of a user might impact on other correlated data that are vital for the business of the digital platform. Such data might have been computed investing enormous resources and an update on the original data, as in the case of the RtBF, might require very expensive re-computations to obtain new correct correlated data. In this last case, not only data erasure affects the quality of the remaining data in the digital platform, but removing some data might be very hard or nearly impossible without compromising the integrity of an important sector of the platform.

Last but not least, some ambiguous interpretations on the definition of personal data can impact also on computed data. Indeed, depending on subtle considerations related also to the chosen encoding and the links among data items, in some cases, correlated data could also fall in the domain of personal data and thus, indirectly, are also the target of data removal requests.

Example of correlated data that is personal data. A cryptographic hash h of some personal data d is a value that has been obtained as output of a computation through a special function H on input d. According to various interpretations, it turns out that whenever d is not unpredictable, h belongs to the category of personal data too. Indeed, consider the case in which d is an email address, h is its cryptographic hash (i.e., $h = H(d)$), and h can be retrieved along with a description of old events associated to that email address. Next, everyone can pick any email address d' (since usually email addresses are not unpredictable) computing H on it and checking the equality of such output with h, therefore detecting (when $d' = d$) that those events involved the original email address d. This can be verified even in case d is removed and only h and the description of the associated events remain in the platform. As such, one might desire to remove h, but this could be not so straight-forward: indeed h could be part of a data structure (e.g., a block of a blockchain) that would have an inconsistent state once h is removed, and making it consistent again could be highly non-trivial.

This tension among privacy features desired by users and effectiveness/efficiency of digital platforms can have a serious impact on the actual application of the RtBF. Users' requests could end up being denied even when legit since digital platforms might find more convenient to speed up their systems moving resources towards their legal office rather than embarking in compliance by design to the RtBF. Expensive escalations to courts would then be required (e.g., see [12]) and in the end citizens might actually prefer to give up on their rights.

The gap between legislators and developers of digital platforms. The design of a digital platform could be unfriendly to the RtBF even when designers had in mind from the very beginning the requirement of allowing deletion of personal data upon request. This is due to the fact that interpretations of what is or is not personal data can change over time. Moreover, the content of the GDPR is way far from technical specifications required by a developer of a digital platform. The long bridge between the content of a law and the corresponding precise behavior of a compliant system is dense of guesses, misunderstandings, and thus it is strongly error-prone. Data Protection Authorities sometimes provide clarifications only when digital platforms are almost ready to be deployed[1]. Interestingly there are attempts to formalize the meaning of data deletion and various definitions have been proposed in recent papers [11, 18, 19].

Contribution of this work. In addition to discussing issues related to actual difficulties that can be faced when deleting personal data from a digital platform, in this work we will also discuss the power of zero-knowledge (ZK) proofs [21] and their applications to the RtBF. A ZK proof system is a building block in the foundations of cryptography that has remained for long time confined in theoretical research papers, but that has found in the last decade several impressive real-world applications. We will indeed discuss how zero-knowledge proofs can be an effective solution to guarantee compliance to the RtBF in digital platforms even in those cases that seem to be very hard to address.

We will show an assessment scheme aiming at determining whether ZK proofs can be helpful to make a system compliant to the RtBF by design. Next we will analyze three notable uses cases (i.e., machine unlearning, redactable blockchains, image authentication) that can be seen as concrete instantiations of our general-purposes assessment scheme.

2 THE RTBF IN DIGITAL PLATFORMS: TOUGH CASES

In Section 1 we have discussed a simple toy example about removing an email address d while its cryptographic hash h is still stored on the platform. The goal of that example was only to illustrate why removing personal data from a digital platform does not always consist of a simple erasure of some records in a database.

In the section we focus on three popular use cases of digital platforms that seem to be extremely unfriendly to data removal, and thus unfriendly to compliance with the GDPR.

2.1 Machine Learning Models

A machine learning model is an algorithm that, during a training phase, processes large datasets in order to be then able to perform a task exploiting what it learnt (rather than being explicitly programmed for accomplishing it). The training phase can consist of very expensive computations that can take days, weeks or months depending on the amount of data to process and the way data are elaborated by the machine learning model. There can be pretty large storage requirements during training and during the inference phase (i.e., when the model is used for a task), that can span from megabytes to hundreds of gigabytes.

Datasets processed by a machine learning model can potentially involve personal data. This means that even in case in the end the actual dataset that has been processed will be deleted, the information stored on the digital platform can implicitly include such personal data. Therefore, it is fairly possible that personal data used to train the model, including data for which the RtBF should be enforced, remain silently/implicitly stored on the digital platform and can later on be revealed during the inference phase.

[1]This happened for instance when EU countries adopted the contact tracing systems based on Exposure Notifications [22].

It is already known [32] that during the inference phase, by appropriately querying the model, it is possible in some cases to obtain portions of datasets that were used during the training. In this direction, there has been great visibility for the result of [28] that reported their successful attempt to extract megabytes of ChatGPT's training data. Notice that ChatGPT [31] is a large language model that is a specific machine learning model having the goal to learn and use human language. The training of ChatGPT was performed with public data available on the internet.

How is this related to the RtBF and what can possibly go wrong? First of all, the above data extraction from a large language model is a typical example in which results of extremely expensive computations can embed personal data that are seemingly hard to remove. Even considering the specific case of ChatGPT where datasets were publicly available, nothing prevents that personal data are originally available on a public source, and later on might be removed precisely upon a legitimate request to be forgotten. Still, those personal data would remain silently in ChatGPT and could at some point be revealed to whoever asks a specific query.

Machine unlearning for the RtBF. Enforcing the RtBF in machine learning models can be very problematic when the possibility of "unlearning" was not considered at design time. The extreme solution is extraordinary expensive and consists of repeating the training using datasets that do not include anymore those data that must be forgotten. Better solutions can instead consist of a different training allowing the model to be updated in order to forget some information [9, 34]. However, even in case such unlearning possibilities exist, there is no guarantee that a digital platform has actually run the unlearning procedure to update the model[2]. Indeed, recall that data can still be silently/implicitly stored in the model and the actual way to extract them could be known only in the future (e.g., when it will be discovered how to exploit inference to reconstruct parts of the dataset). As such, a digital platform could be lazy and lie, claiming that everything possible has been done in compliance to the RtBF while instead this is not true[3].

An alternative approach to guarantee that the process of forgetting personal data has been successfully completed on a digital platform, consists of proving that there has been a transition from the previous state of the platform to the next state and the transition consisted of running the unlearning procedure. The proof proving such transition should be privacy preserving in order not to leak personal data that have been used to produce the state of the system. We will see that this theoretical approach can be concretely viable and can be seen as a specific case of our generic assessment scheme.

2.2 Blockchain Technology

Blockchain technology allows one to construct, through consistent replicas, a decentralized digital platform that is fault-tolerant and resilient to high degrees of corruption. Such a platform usually achieves its desired security through the concept of immutability. This property guarantees that the entire history of ordered transactions that moved the system from its original state (i.e., the genesis) to the current state is publicly verifiable.

The immutability property of blockchains makes the resulting platform extremely transparent about its state, since everyone can check it on her own, trusting nobody. Transparency is a feature that makes blockchains an appealing technology to realize robust systems in a decentralized setting, therefore without relying on trusted parties that could be single points of failure.

The chain of blocks. The name blockchain is due to the basic mechanism used by such platforms to guarantee immutability: ordered transactions are stored in blocks that are chained to each other through hard-to-find but easy-to-check strings. Finding such special strings can be seen as finding the solution to a puzzle or winning a scratch card lottery. Examples of techniques used to implement the puzzle and the lottery are the "proof of work" used in Bitcoin [27] and the "proof of stake" used now in Ethereum [8].

There exist so-called "permissioned" blockchains where the governance is limited to well-known organizations that through honest majority guarantee the correctness of the included data. Because of their limited decentralization, permissioned blockchains are much easier to construct and even immutability is not that hard to relax as shown in [1]. We will stick for now with Bitcoin but the discussion can be extended also to Ethereum and several other mainstream permissionless blockchains.

Adding a new block in Bitcoin requires to find a special input to a cryptographic hash function, named proof of work, such that the output is a string belonging to a very small subset of the output space. Computers that try to find such strings are called miners, and currently the effort required to add a new block consists of more than 2^{70} evaluations of a cryptographic hash function every 10 minutes. Immutability is guaranteed by the fact that an update of a single bit in a block would invalidate the solution to the puzzle that appears in the next block, and thus it would invalidate the link with all next blocks.

In Bitcoin there can be personal data. Each block in Bitcoin contains transactions and there are two specific ways to encode text in transactions. The first mechanism can be used only by the miner and consists of using a field of the special transaction that only the miner of a block can add (i.e., the coinbase transaction). The second mechanism can be used by any user and consists of submitting a transaction with a special keyword OP_RETURN, that allows to embed free text in a transaction.

In [26] it has been shown that there are transactions in Bitcoin that include illegal data, for instance links to web resources storing material related to child pornography. The above injection of such data in Bitcoin transactions has been performed precisely through the use of the above OP_RETURN keyword. Fortunately a link can be made meaningless by deleting the linked resources (rather than the link), but still there can be explicit and fully specified contents in a transaction that one might want to see deleted.

Summing up, there is a major problem due to the possibility that personal data be stored in a public blockchain like Bitcoin. Indeed, there are several blockchain projects that use the blockchain of

[2]Digital platforms do not like to invest too many resources on tasks that do not bring a corresponding revenue.

[3]We are not considering the case of a digital platform that keeps a copy of the dataset claiming to have satisfies all data deletion requests. The reason is that we are focusing on the technological possibility of correctly and efficiently implementing the RtBF through a proper design and in showing that tasks specified in the design are correctly executed. The fact that one can also keep copies of data illegally, besides being unavoidable, is out of the scope of our work.

Bitcoin for applications unrelated to the transfer of cryptocurrencies (e.g., applications interested in leaving a permanent message like EternityWall). If one of such blockchain projects ends up uploading personal data in a Bitcoin transaction, then by the immutability of the blockchain there will be no way to forget that information. In the second part of the paper we will see that, somewhat surprisingly, this intuition is false.

2.3 Image Authentication

There is a growing debate about issues deriving by deepfakes, and in general by the fact that disinformation can exploit the impact of sophisticated fake pictures. In order to try to limit the spread of fake news and give solid guarantees about the originality of pictures, the Coalition for Content Provenance and Authenticity (C2PA) [10] has defined a standard that some cameras are already implementing aiming at linking original pictures to digital signatures. The vision of the C2PA standard is that cameras should include in a tamper-proof area a secret key and a circuit computing signatures of the images captured by the camera. Then such pictures can be published and verified through the verification procedure of the signature scheme, therefore recognizing as genuine the signatures verified with public keys of camera producers. The above approach aims at ruling out fake pictures since they would not be equipped with a signature that is verified according to a public key with good reputation.

The RtBF from digital images. Once a digital signature is computed to assess the authenticity of a picture, even an update of a single bit of the picture would make the signature invalid hurting authenticity. It is therefore straight-forward to see the clear issue about a signed picture including the recognizable face of an individual that at some point might ask to have the blur operation applied to the area of the picture including her face. While this transformation is easy to implement through a common photo editing software, such an edit makes the original signature meaningless therefore loosing authenticity of the overall picture.

Summing up, a digital platform might have a picture including personal data and some correlated information (i.e., the signature) such that removing personal data hurts the consistency of correlated data that can not be re-computed.

3 ZERO-KNOWLEDGE PROOFS

In [21], Goldwasser, Micali and Rackoff proposed a revolutionary concept: the existence of a Zero-Knowledge (ZK) proof. This is a game played by a prover and verifier both sharing a claim. The prover also holds as input an evidence of the veracity of the claim and, moreover, can use it to convince the verifier, without disclosing any additional information about her private input (i.e., the evidence of the prover used to convince the verifier). At first sight, one might think that ZK proofs are a mechanism to perform secure identification or to show possession of credentials. Those are just two very immediate applications but stopping with them would mean to look at the finger while instead ZK proofs indicate the moon.

The strong power of ZK proofs that makes them appealing in several applications lies in their paradoxical ability to relax the tension between accountability and privacy. ZK proofs enable anyone to prove that a computation has been carried out correctly, while at the same time the confidential information that has been used to computed the proof is hidden in the proof, in a way that nobody can extract it (in a computational sense).

We now discuss the meaning of the definition of ZK proof. This will be useful to concisely describe our ZK-based assessment scheme.

3.1 Definition of a Zero-Knowledge Proof System

There exist several different definitions that corresponds to different flavors of ZK proofs. We report here, standard definitions widely used in the literature. In particular we follow the description given in [33].

Notation. We will use $\epsilon(\cdot)$ to denote a negligible function (i.e., for every constant c and all sufficiently large n it holds that $\epsilon(n) < 1/n^c$). Negligible functions are useful to bound the probability of events that one would like to happen extremely rarely. Given an $\mathcal{NP}$ language L, we consider the polynomial-time relation R_L consisting of pairs (x, w) such that $x \in L$ and w is a witness for an efficient (i.e., polynomial in the size of the input) membership verification procedure for L. Roughly, x will be the output of some computations and w is the input that has been used for those computations, and that might include confidential data, therefore it must remain a secret of the prover.

Proof system. A proof system is a two party game with a first player that is a prover and is usually denoted by P and a second player that is a verifier and is usually denoted by V. Both are probabilistic polynomial-time (PPT) interactive algorithms in the sense that they run efficiently in the size of their inputs, and they are randomized, therefore they have access to their own private sources of randomness. While both P and V know x, P also knows a witness w such that $(x, w) \in R_L$. In other words, w is possibly a confidential secret that can be used to explain that x is well formed, and this is denoted through membership into an $\mathcal{NP}$ language. The output of V at the end of the above execution is usually denoted by $\langle P(w), V\rangle(x)$, and corresponds to 1 when V accepts the proof and to 0 when instead V rejects it.

Definition 3.1. A proof system $\Pi = (P, V)$ for an $\mathcal{NP}$-language L is a pair of PPT interactive algorithms satisfying the following two properties.

- Completeness: for all $(x, w) \in R_L$, $\Pr[\langle P(w), V\rangle(x) = 1] = 1$.
- Soundness: there exists a negligible function ϵ such that for every $x \notin L$ and for every adversary $P^\star$, $\Pr[\langle P^\star, V\rangle(x) = 1] < \epsilon(|x|)$.

Essentially, completeness means that if both players behave correctly then P will manage to convince V that the claim is true (i.e., that x is the correct output of some computation on a secret input w). Soundness instead models the case in which P is malicious and therefore can deviate from the prescribed protocol with the goal of convincing V about the veracity of a claim that instead is false. Still, we want that the probability that V will be cheated is negligible. Completeness and soundness in the above two-party game characterize a proof system.

Zero knowledge. Proof systems are easy to construct, indeed the prover could simply send the witness. However when privacy of the witness is a concern, more sophisticated constructions are required. In order to evaluate their ability to preserve the privacy of the witness, zero knowledge has been defined.

Definition 3.2. A proof system $\Pi = (P, V)$ for an NP-language L is computational zero knowledge if for any PPT algorithm $V^\star$ there exists an expected PPT algorithm S such that for any $(x, w) \in R_L$ and any $z \in \{0,1\}^\star$ the following two distributions are computationally indistinguishable:

$$\{\langle P(w), V^\star(z)\rangle(x)\},\ \{S^{V^\star}(x, z)\}.$$

The above third property of an interactive proof system is the guarantee for the honest prover that her secret will remain protected even in case the adversarial verifier misbehaves arbitrarily. Indeed, whatever could have been learned by an even adversarial $V^\star$ during the execution of the protocol with a honest P, essentially the same could have been computed locally (i.e., without any interaction with P) by $V^\star$ running the algorithm S, that, unlike P, does not receive the secret as input and still can output an indistinguishable view in the eyes of $V^\star$.

The use of a simulator to model the security of a protocol has been extremely influential in many scenarios, and several security definitions (e.g., secure multi-party computation) follow the same approach that is usually referred to as *the simulation paradigm*.

It is worthy to note that the above definition is oriented to an interactive system where P and V exchange multiple messages. Such proofs are convincing only for the very specific verifier that engaged in the execution of the protocol.

Non-interactive ZK. There is also a non-interactive form of ZK proof where P outputs a single message that can be then tested by any verifier without sending a message to the prover. This non-interactive setting achieves public verifiability that is a desired property in several applications. A non-interactive ZK (NIZK) proof can be instantiated in two different settings. In the first setting, there are some parameters that are generated either by a trusted third party or by a distributed computation [30][4]. Such parameters, sometimes referred to as a "common reference string", are received as input both by the prover and by any verifier interested in checking the correctness of the proof. The second setting instead relies on a heuristic assumption about a cryptographic hash function, assuming that it behaves as a random oracle [3] (i.e., the output of a query is a random string and identical queries are answered consistently).

Arguments, knowledge soundness and SNARKs/STARKs. The adversarial prover in the definition of soundness is often relaxed to a PPT algorithm and in this case the word *proof* is replaced by *argument*. The notion of soundness is cumbersome when a claim is certainly true and in this case the entire goal of the prover is showing possession of a valid witness. In this case a variant of the notion of soundness, referred as *knowledge soundness* is used to guarantee some meaningful security to the honest verifier.

[4]In this case, honesty of at least a specified number of participants guarantees that the generated parameters are correctly formed and no player owns specific trapdoors correlated to the parameters.

There exist succinct non-interactive arguments of knowledge (SNARKs) [6] that are essentially some NIZK arguments with knowledge soundness, producing a compact proof that is very fast to verify. When there is no trapdoor associated to the generation of the parameters of the SNARK, then the commonly used acronym is STARK [4] that stays for succinct transparent argument of knowledge. While STARKs are obviously preferable for security, they usually are less succinct and slower to verify.

While both SNARKs and STARKs can be computed to prove any claim in $\mathcal{NP}$, the computation of such proofs can be extremely demanding depending on the type of claim that they are supposed to prove and validate. In the literature the terms ZK-SNARK/ZK-STARK are used when the ZK property is also enjoyed. In our scenarios, since the ZK property is obvious we will leave it implicit therefore using the terms SNARK/STARK only.

4 AN ASSESSMENT SCHEME FOR GDPR COMPLIANCE THROUGH ZERO-KNOWLEDGE PROOFS

In Section 2 we have discussed specific complex cases where enforcing the RtBF is extremely problematic. Here we would like to propose an assessment scheme allowing to test, to some extent, if the design of a digital platform can be compliant to the RtBF, therefore allowing efficiently to remove personal data upon request, without hurting the consistency of the platform.

We depict in Figure 1 our assessment scheme that in particular takes into account the power of ZK proofs to maintain consistency in the presence of expensive to re-compute correlated data. Ours is an initial attempt to push forward the notion of RtBF by design leveraging ZK proofs.

The depicted assessment scheme is beneficial to evaluate the compliance of a design of a digital platform to the Art. 17 of the EU GDPR. The spirit of this scheme is to have a simple and direct method to detect issues about possible requests of removing personal data. It goes without saying that this is an initial effort and modelling what can actually happen in the wild is certainly more complex and richer of challenges.

Details on the assessment scheme. We now discuss in details the assessment scheme shown in Figure 1 following the alphabetic order of the letters assigned to the shapes.

- **(a)** All possible personal data that could be removed should be evaluated upfront in order to make sure that all cases of data removal requests can be efficiently managed. The scheme proceeds therefore in **(b)** considering some personal data α to be removed.
- **(b)** Considering a single request of removing personal data α of an individual, the first question to address is whether α has been used to compute correlated data β that is also stored and thus is part of the digital platform. A negative answer will bring us to **(c)** while a positive answer will bring us to **(d)**.
- **(c)** The conclusion is that α can be safely and efficiently removed. This will make the platform unaware of such data, and moreover there is no negative impact on the consistency of remaining data.

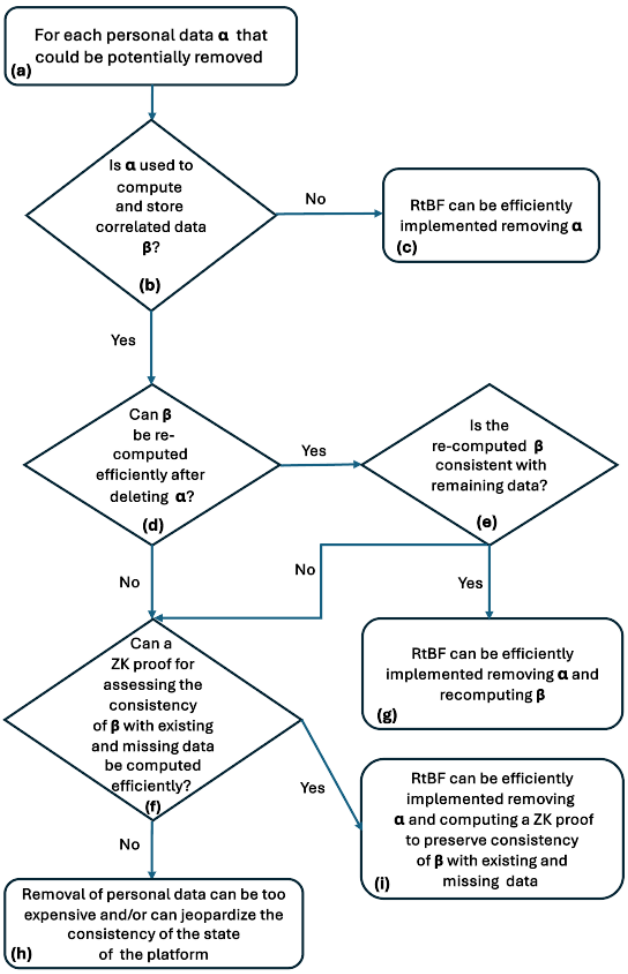

Figure 1: Assessment scheme. It can be used to determine possible issues when receiving a request to delete personal data as required by the Art. 17 of the EU GDPR.

(d) Here we wonder how expensive is to fix β so that it is consistent with data available in the platform after deleting α. If the re-computation of β is efficient, we continue with **(e)**, otherwise we continue with **(f)**.

(e) In this case, re-computation of β can be performed efficiently, and we ask whether the re-computed β would be consistent with other data available in the digital platform. Indeed, it is possible that some other stored data γ was computed taking the old version of β into account. If not (i.e., other data would not be consistent with the new value of β), we continue with **(f)**, otherwise we continue with **(g)**.

(f) This point is reached by tough cases in which removing personal data might be considered too expensive or even unfeasible. Indeed, here we have that α must be removed but unfortunately there exist correlated data β that are too expensive to re-compute, either because of the complexity of its re-computation or because re-computing it would in turn require other re-computations. Nevertheless, after deleting α, the platform would not be in a consistent state if β remains untouched. This is the key point where the power of ZK proofs can be leveraged. Indeed, our assessment scheme suggests to check if it is possible to delete α, to keep correlated data β as it is, but repairing the lost consistency (due to β being related to a non-existing value α) through a ZK proof. In the positive case we continue with **(i)**, otherwise we continue with **(h)**.

(g) In this case we conclude that removing α and updating β can be efficiently performed leaving the platform in a consistent state.

(i) The ZK proof must be computed by a prover for the following informal claim: "there existed α such that the current β was correctly computed". The witness used by the prover consists of α since it induced the correct calculation of β. This proof must be non-interactive and fast to verify, therefore SNARKs/STARKs are good options depending on the possibility of having trusted parameters or of relying on a transparent setup only. The ZK property is required in order to make sure that the string of the computed proof does not convey information about α besides the mere fact that an α compatible with β existed.

(h) If the computation of such a proof is not viable[5], then there is a serious risk that the design of the platform is not compliant with Art. 17 of the GDPR.

In the next three subsections we analyze the above assessment scheme considering the three tough use cases introduced in Section 1.

4.1 The RtBF in Robust Machine Learning via ZK Proofs

Recall that the problem of reliable machine unlearning is to make sure that the digital platform has performed the machine unlearning procedure without leaking data used for the training. Even though this task might seem excessively hard to accomplish, we can here comment how this can actually be realized leveraging the power of ZK proofs and how the proposed approach can be seen as an instantiation of our assessment scheme.

In [15], the authors presented a design enforcing the digital platform to prove the correct execution of a learning or unlearning procedure. More precisely, the digital platform when using a dataset to train the model will also compute a proof. The goal of the proof is to make sure that the state of the system has been updated as consequence of the execution of the expected procedure. The same will happen in the presence of a request to unlearn data therefore producing an updated state. The proofs are zero knowledge so that no information about the unlearned data can be acquired from the proof. Another reason explaining why the proof should not leak information is that a request for data deletion, that in this case is a request for machine unlearning, should receive as answer only a guarantee that the unlearning has been performed, and other

[5]We remark here that while in theory one can construct ZK proofs for any claim that is used in such scenarios, the concrete efficiency of such proofs can be extremely unsatisfying in some cases.

information related for instance to the non-erased datasets should still remain confidential.

Since datasets are often large, proofs are in turn prone to be large and verifying them can be expensive. Therefore SNARKs are used since succinctness and fast verification is of major importance.

The construction of [15] keeps two efficiently updatable data structures named Merkle trees, one for the data used for training and one for data that instead has been *unlearned*. At each round of training and/or unlearning the states of the two Merkle trees change and a SNARK is computed to ensure that the change of state has been correctly performed according to the previous state and to the requests of training/unlearning performed.

The above system has been also validated in [15] by an implementation and an analysis. While, as the authors admit, there are still various aspects to address in future research, the proposed solution is an impressive milestone about how ZK proofs can allow to obtain levels of data protection, including the case of data removal, that at first sight might look impossible to reach.

Robust machine unlearning through the lens of our assessment scheme. Whenever a dataset includes personal data α, the corresponding machine learning model (depending on how the training is performed) can include some information β that is correlated to α. Considering our assessment scheme, it is obvious that step (**d**) is reached and that β represents data in the model that can allow to recover α. As such, the approach of [15] is such that one can modify β into β' so that β' will not be correlated to α but will not be consistent with remaining data, therefore reaching (**f**) through (**e**). Then, their system allows for an efficient ZK proof that β' is correct, therefore reaching (**i**).

4.2 The RtBF in Blockchains: ZK Proofs for Cut and Patch

As discussed in Section 1, the presence of personal data in a Bitcoin transaction t is a major problem since any change to a single bit of that transaction t would invalidate the proof of work that allowed to connect the block B' following the block B including t. In turn, removing data from a transaction, even though the removed data has no impact on the actual transfer of units of the cryptocurrency (which is supposed to be the main goal of a transaction), would make inconsistent the connection with the remaining chain of blocks (i.e., with all blocks that followed the one in which the change takes place).

The use of ZK proofs in the context of blockchains has been considered in the past mainly for the goal of providing privacy-preserving smart contracts [14], a privacy-preserving cryptocurrency [5] and in general for using blockchains also in applications that include confidential data [17]. Here we now illustrate a solution presented in [7] that, again surprisingly, relies on ZK proofs to allow data removal from Bitcoin transactions. The same approach was later on considered again for Bitcoin in [25]. The approach of [7] has inspired a technique for redaction of smart-contract enabled permissioned blockchains [2].

The proposed data removal mechanism of [7] implements the idea of cutting the piece of data (i.e., the part of a transaction that included text that must be erased) and then applying a patch. The phase of cutting is digitally performed by zeroing out the involved bits. However, recall that once bits in a block change, the connection with the next block is broken. To tackle this issue it is therefore necessary to apply a patch, that in this case would be a ZK proof that will certify that the link between the blocks must be considered valid, since it was a valid link with a previous version of the block and only some "neutral" data were replaced in the block. Since computing a ZK proof requires the special information that can explain the consistency among the blocks, the sequence of events in the system of [7] goes as follows.

(1) The instance x corresponds to the new block B' (i.e., the one with a transaction t' that has some zeroed bits), the next block A that includes the solution of the puzzle connecting A to a block B (i.e., B is the block before the update takes place changing a transaction t into t'). The witness w for P will be the block B, therefore including t. Essentially the common instance x is the result of the computation that zeroed some bits of t, and w is the secret information that can be used to show that B' and A are still virtually well connected since B' is just the new version, through some neutral[6] updates, of a block that was well connected to A.
(2) P computes the ZK proof π, that in this case is non-interactive since it must be publicly verifiable.
(3) Next P deletes B.

The consequence of the above steps is that even though a Bitcoin node (P in the above example) has deleted data from its own copy of the Bitcoin blockchain, it is still possible for whoever connects to this node to download the blockchain and to verify its consistency/correctness since when checking the connection between B' and A, instead of verifying as usual the solution of the puzzle included in A, the patch, which is the ZK proof, will be verified instead. Since Bitcoin relies on decentralization, trusted parameters would be not acceptable and thus the ZK proof is implemented through STARKs.

Redactable blockchains through the lens of our assessment scheme. Whenever a transaction includes personal data α, the blockchain includes α in a block and the next block includes β that is the link to a previous block. Therefore, β is clearly correlated to α. Considering our assessment scheme, it is obvious that step (**d**) is reached and that recomputing β is in general hard in the context of blockchains (e.g., it could correspond to producing a new proof of work that in turn would require to update also next blocks). The approach of [7] consists of keeping β as it is reaching directly (**f**) in the assessment scheme. Then, their system allows for an efficient ZK proof (i.e., a STARK) that β is correct (despite the available transactions do not show so), therefore reaching (**i**).

4.3 The RtBF on Authentic CP2A Images: ZK Proofs of Correct Transformations

As discussed in Section 2.3, a digital platform including a authentic picture, might be asked to update the picture as part of a request of an individual appealing to the RtBF. When the picture is compliant to the C2PA standard there is also a signature that is crucial to

[6] Recall that the zeroed bits only affect some free text of the transaction that has no impact on the possession of bitcoins.

guarantee the authenticity of the picture and altering the picture would hurt authenticity.

The approach of PhotoProof. The work of [29] showed that one can maintain consistency of a signature over and edited picture through the use of ZK proofs. In their proposal, whoever edits a digitally signed picture can then compute a SNARK proving that the modified picture corresponds to an original picture (where originality comes from the existence of signature under some respectful public key) that however has been modified according to a well-specified transformation. In this way there are strong guarantees about what the current (modified) picture shows and what was originally signed.

Transformed authentic images through the lens of our assessment scheme. Whenever a C2PA-compliant picture includes personal data α, there is also a correlated information β that includes a signature of α. Considering our assessment scheme, it is obvious that step (**d**) is reached and that recomputing β is unfeasible since the digital platform does not have the secret key corresponding to the respectful public key guaranteeing the authenticity of the picture. As such, the approach of [29] is such that one keeps β as it is, therefore reaching directly (**f**). Then, their system allows for an efficient ZK proof that β is correct, therefore reaching (**i**). While such ZK proof computed in [29] can only be computed for pictures with very low resolution, there are more recent constructions [13, 24] allowing to efficiently certify the correctness of the involved transformations maintaining the desired degree of authenticity also of high-resolution pictures.

5 CONCLUSIONS

The RtBF is a fundamental milestone of the GDPR. Unfortunately, individuals that expect their data to be protected by the GDPR can be in trouble when desiring to be forgotten.

We have discussed the problem of enforcing data removal in digital platforms in those cases where it could correspond to a tremendous waste of resources.

Our work contributes to a better understanding of the power of ZK proofs in mitigating the tension between security/privacy-oriented desires of users and businesses-oriented desires of managers of digital platforms. We have shown that in general, digital platforms can be designed to be compliant to the Art. 17 of the GDPR and at the same time can perform intensive computations and store large amounts of data that are correlated to personal data. Moreover we have shown three specific use cases where data removal in seemingly tough, but that, as abstracted by our assessment scheme, through ZK proofs and a re-design process can actually allow data removal with an affordable effort.

Interesting future directions consist of improving the assessment scheme by expanding the part that checks whether an efficient ZK proof exists, providing guidelines to assess if, depending on the specific data deletion scenario, an efficient ZK proof fitting the proposed scenarios can be concretely deployed.

ACKNOWLEDGMENTS

The author is member of the Gruppo Nazionale Calcolo Scientifico-Istituto Nazionale di Alta Matematica (GNCS-INdAM) and his research contribution on this work is financially supported under the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.1, Call for tender No. 104 published on 2.2.2022 by the Italian Ministry of University and Research (MUR), funded by the European Union -NextGenerationEU - Project Title "PARTHENON" - CUP D53D23008610006 - Grant Assignment Decree No. 959 adopted on June 30, 2023 by the Italian Ministry of Ministry of University and Research (MUR).

REFERENCES

[1] Giuseppe Ateniese, Bernardo Magri, Daniele Venturi, and Ewerton R. Andrade. 2017. Redactable Blockchain - or - Rewriting History in Bitcoin and Friends. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017*. IEEE, 111–126.

[2] Gennaro Avitabile, Vincenzo Botta, Daniele Friolo, and Ivan Visconti. 2024. Data Redaction in Smart-Contract-Enabled Permissioned Blockchains. In *Proceedings of the Sixth Distributed Ledger Technology Workshop (DLT 2024), Turin, Italy, May 14-15, 2024 (CEUR Workshop Proceedings, Vol. to appear)*. CEUR-WS.org.

[3] Mihir Bellare and Phillip Rogaway. 1993. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In *CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993*, Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby (Eds.). ACM, 62–73.

[4] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. 2018. Scalable, transparent, and post-quantum secure computational integrity. IACR cryptol. eprint arch.:2018, 046

[5] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. 2014. Zerocash: Decentralized Anonymous Payments from Bitcoin. In *2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014*. IEEE Computer Society, 459–474.

[6] Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. 2013. Recursive composition and bootstrapping for SNARKS and proof-carrying data. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, Dan Boneh, Tim Roughgarden, and Joan Feigenbaum (Eds.). ACM, 111–120.

[7] Vincenzo Botta, Vincenzo Iovino, and Ivan Visconti. 2022. Towards Data Redaction in Bitcoin. *IEEE Trans. Netw. Serv. Manag.* 19, 4 (2022), 3872–3883.

[8] Vitalik Buterin and Nathan Schneider. 2022. *Proof of Stake: The Making of Ethereum and the Philosophy of Blockchains*. Seven Stories Press.

[9] Yinzhi Cao and Junfeng Yang. 2015. Towards Making Systems Forget with Machine Unlearning. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*. IEEE Computer Society, 463–480.

[10] Coalition for Content Provenance and Authenticity. 2023. *C2PA Specifications*. Retrieved April 27, 2024 from https://c2pa.org/specifications/specifications/1.3/index.html

[11] Aloni Cohen, Adam D. Smith, Marika Swanberg, and Prashant Nalini Vasudevan. 2023. Control, Confidentiality, and the Right to be Forgotten. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, CCS 2023, Copenhagen, Denmark, November 26-30, 2023*, Weizhi Meng, Christian Damsgaard Jensen, Cas Cremers, and Engin Kirda (Eds.). ACM, 3358–3372.

[12] Court of Justice of the European Union. 2014. *Google Spain v AEPD and Mario Costeja González*. Retrieved April 27, 2024 from https://en.wikipedia.org/wiki/Google_Spain_v_AEPD_and_Mario_Costeja_Gonzalez

[13] Trisha Datta and Dan Boneh. 2023. *Using Zk-proofs to fight disinformation*. Retrieved April 27, 2024 from https://rwc.iacr.org/2023/acceptedpapers.php `https://medium.com/@boneh/using-zk-proofs-to-fight-disinformation-17e7d57fe52f`.

[14] Dmitry Khovratovich and Mikhail Vladimirov. 2019. *Tornado Privacy Solution Cryptographic Review Version 1.1*. Retrieved April 27, 2024 from https://tornadoeth.cash/audits/TornadoCash_cryptographic_review_ABDK.pdf

[15] Thorsten Eisenhofer, Doreen Riepel, Varun Chandrasekaran, Esha Ghosh, Olga Ohrimenko, and Nicolas Papernot. 2022. Verifiable and Provably Secure Machine Unlearning. arXiv:2210.09126

[16] European Parliament and Council of the European Union. 2016. *Regulation (EU) 2016/679 of the European parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (general data protection regulation)*. Retrieved April 27, 2024 from https://eur-lex.europa.eu/eli/reg/2016/679/oj

[17] Michèle Finck. 2019. *European Parliamentary Research Service: Blockchain and the General Data Protection Regulation.* Retrieved April 27, 2024 from https://www.europarl.europa.eu/RegData/etudes/STUD/2019/634445/EPRS_STU(2019)634445_EN.pdf

[18] Sanjam Garg, Shafi Goldwasser, and Prashant Nalini Vasudevan. 2020. Formalizing Data Deletion in the Context of the Right to Be Forgotten. In *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12106)*, Anne Canteaut and Yuval Ishai (Eds.). Springer, 373–402.

[19] Jonathan Godin and Philippe Lamontagne. 2021. Deletion-Compliance in the Absence of Privacy. In *18th International Conference on Privacy, Security and Trust, PST 2021, Auckland, New Zealand, December 13-15, 2021*. IEEE, 1–10.

[20] Oded Goldreich, Silvio Micali, and Avi Wigderson. 1987. How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, Alfred V. Aho (Ed.). ACM, 218–229.

[21] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. 1985. The Knowledge Complexity of Interactive Proof-Systems (Extended Abstract). In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA*, Robert Sedgewick (Ed.). ACM, 291–304.

[22] Google LLC and Apple Inc. 2020. *Exposure notification—Cryptography specification.* Retrieved April 27, 2024 from https://storage.googleapis.com/gweb-uniblog-publish-prod/documents/Exposure_Notification_-_Cryptography_Specification_v1.2.1.pdf

[23] Lukas Helminger and Christian Rechberger. 2022. Multi-Party Computation in the GDPR. In *Privacy Symposium 2022*, Stefan Schiffner, Sebastien Ziegler, and Adrian Quesada Rodriguez (Eds.). Springer International Publishing, 21–39.

[24] Daniel Kang, Tatsunori Hashimoto, Ion Stoica, and Yi Sun. 2022. ZK-IMG: Attested Images via Zero-Knowledge Proofs to Fight Disinformation. arXiv:2211.04775

[25] Enrique Larraia, Mehmet Sabir Kiraz, , and Owen J Vaughan. 2024. How to Redact the Bitcoin Backbone Protocol. IACR cryptol. eprint arch.:2024, 813

[26] Roman Matzutt, Jens Hiller, Martin Henze, Jan Henrik Ziegeldorf, Dirk Müllmann, Oliver Hohlfeld, and Klaus Wehrle. 2018. A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin. In *Financial Cryptography and Data Security - 22nd International Conference, FC 2018, Nieuwpoort, Curaçao, February 26 - March 2, 2018, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 10957)*, Sarah Meiklejohn and Kazue Sako (Eds.). Springer, 420–438.

[27] Satoshi Nakamoto. 2008. *Bitcoin: A Peer-to-Peer Electronic Cash System.* Retrieved April 27, 2024 from https://bitcoin.org/bitcoin.pdf

[28] Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. 2023. Scalable Extraction of Training Data from (Production) Language Models. arXiv:2311.17035

[29] Assa Naveh and Eran Tromer. 2016. PhotoProof: Cryptographic Image Authentication for Any Set of Permissible Transformations. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*. IEEE Computer Society, 255–271.

[30] Valeria Nikolaenko, Sam Ragsdale, Joseph Bonneau, and Dan Boneh. 2024. Powers-of-Tau to the People: Decentralizing Setup Ceremonies. In *Applied Cryptography and Network Security - 22nd International Conference, ACNS 2024, Abu Dhabi, United Arab Emirates, March 5-8, 2024, Proceedings, Part III (Lecture Notes in Computer Science, Vol. 14585)*, Christina Pöpper and Lejla Batina (Eds.). Springer, 105–134.

[31] OPENAI. 2023. GPT-4 technical report. arXiv:2303.08774

[32] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. IEEE Computer Society, 3–18.

[33] Ivan Visconti. 2022. *Asymmetric Cryptography: Primitives and Protocols.* John Wiley & Sons, Ltd, Chapter Zero-Knowledge Proofs, 63–84.

[34] Howard Wu, Wenting Zheng, Alessandro Chiesa, Raluca Ada Popa, and Ion Stoica. 2018. DIZK: A Distributed Zero Knowledge Proof System. In *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018*, William Enck and Adrienne Porter Felt (Eds.). USENIX Association, 675–692. https://www.usenix.org/conference/usenixsecurity18/presentation/wu