Hedera™ Hashgraph

DATA PRIVACY COMPLIANCE USING HEDERA CONSENSUS SERVICE

CO-AUTHORS:

Noah Buxton
DIRECTOR, BLOCKCHAIN PRACTICE LEADER
armanino

Paul Madsen
TECHNICAL LEAD

Sam Brylski
ASSOCIATE GENERAL COUNSEL

SUMMARY AND DISCLAIMERS

The Hedera Consensus Service (HCS) enables a decentralized architecture that can help solve for certain data privacy compliance challenges presented by naïve implementations of distributed ledger technology (DLT), while making possible new data privacy compliance mechanisms that are consistent with principles of user empowerment and control over how their identity attributes are collected, stored, processed, and shared. These principles are central to the European Union's General Data Protection Regulation (the GDPR) and other data privacy regulation frameworks around the globe.

This publication is intended to provide an overview of some of the features and functionality of HCS and should not be considered or relied upon as legal advice. Data privacy regulations continue to evolve, may vary significantly between jurisdictions, and can be ambiguous when applied to uses of emerging technology. Other obligations and regulations not discussed in this paper may apply to participants in the Hedera ecosystem depending on the context of their participation in the Hedera Network or characteristics of their application. Developers utilizing HCS should seek legal and compliance advice from an independent qualified professional.

CONTENTS

INTRODUCTION

The regulatory landscape for consumer data privacy was drastically broadened in 2018 with the European Union's sweeping, complex General Data Protection Regulation (GDPR) becoming effective in May of that year. Now, over two years later, the incredible pace of change, as well as continued regulatory uncertainty among businesses and consumers, still abounds. Indeed, a movement to protect personal data is fermenting in several of the individual US states. Three states have passed personal data privacy legislation since 2018, including the California Consumer Privacy Act (CCPA), and many other states have bills moving through the legislative process.

The data privacy regulations of this new class often have a number of common principles or grant a similar set of rights to the data subject. Among other things, both the GDPR and the CCPA convey the rights to correct, update, or delete personal data on a business database and require businesses to obtain consent for certain collection, processing, transfer, or sale of personal data. Developers must be prepared to consider these and other fundamental principles when designing applications and business models.

Scrutiny of the way organizations manage consumers' data privacy rights has never been higher. Regulators have robust enforcement programs, consumers across the globe are learning how they can take action to protect their own data privacy, and enterprises large and small are tackling the operational, technical, and reputational challenges to business-as-usual. At the same time, there remains significant uncertainty in how these data privacy regulations apply to distributed ledger technology (DLT) and a lack of compliance mechanisms available to developers on most distributed platforms.

Members of the Hedera Governing Council and other large global enterprises are actively working on projects that will use Hedera Network's Consensus Service (HCS) to solve these and other new challenges. Other businesses are sprouting in the fertile grounds of the Hedera ecosystem, many of which are developing decentralized applications (dapps) to leverage the Hedera Network's Asynchronous Byzantine Fault Tolerant (ABFT) algorithm. Meanwhile, developers, professional investors, retail investors, and consumers alike are all playing their part in the Hedera ecosystem's Cambrian explosion of innovation. **The purpose of this paper is to provide actionable information for business application network developers using the Hedera Consensus Service to help comply with certain data privacy principles when building on the Hedera Network.**

The challenges presented by new technology often manifest as an incongruity between the promise of the new technology fitting in with the "old way" of regulating, integrating, protecting, controlling, monitoring, etc. Below, this paper will (1) briefly discuss the data mutability and data residency challenges that arise in the context of DLT networks, (2) demonstrate some advantages of HCS as it relates to these challenges, and (3) introduce various compliance mechanisms that become possible with HCS. Finally, this paper will conclude with a case study to illustrate these concepts in action.

DATA PRIVACY COMPLIANCE CHALLENGES ON DLT NETWORKS

Data privacy regulations, such as the GDPR, are often seen as difficult to reconcile with the potentially permissionless, geo-distributed, and immutable nature of public ledgers. Two of the key challenges associated with reconciling data privacy regulations with DLTs are (1) data mutability, and (2) data residency.

Data Mutability

Under the GDPR and other regulations, some data subject rights require the modification of previously collected personal data. For example, data subjects may have the right to correct inaccurate or outdated personal data, commonly referred to as the right of rectification, and the right to deleted collected personal data, referred to as the right of deletion or right of erasure.

Blockchains are invariably described as an immutable or uneditable ledger or database. This immutability is portrayed as a key enabler of the trust in the blockchain through resistance to malicious modifications. Once a transaction is added to a block and that block added to the end of the chain, then after some number of additional blocks are added, the transaction in question is effectively written in stone, as it would be impractical for an attacker to modify that transaction.

Such immutability, so valuable to protect the integrity of a cryptocurrency or similar, is a double-edged sword. Immutability would appear to make it difficult, or perhaps impossible, to satisfy the requirements of the right of erasure, at least with respect to any data stored on-chain.

As an example, consider a cryptocurrency exchange that facilitates the buying and selling of different currencies. Such exchanges will typically maintain a database of user account data on their own off-chain systems, including Know Your Customer (KYC) status and balances. In the typical custodial model, it is only the exchange itself that acts as an account on the blockchain, reflecting the aggregate balance of its users. An individual user's balance is maintained off-chain in the exchange's own database. If a customer were to request the cancellation of their account, transfer of their balances to their own non-custodial wallet, and erasure of any data the exchange had stored, the exchange would be able to comply by deleting its off-chain data for that customer. But the transfer of balance to the user's wallet would manifest as a transaction submitted to the DLT network and persisted in the immutable history of the relevant blockchain, and it would not be possible for the exchange to delete that data.

Data Residency

Data privacy regulations often recognize that their protections may be rendered useless if the relevant personal data is able to be freely transferred to another jurisdiction with less stringent data privacy requirements. In the context of the GDPR, there are detailed requirements around the transfer of EU personal data to, or viewing of

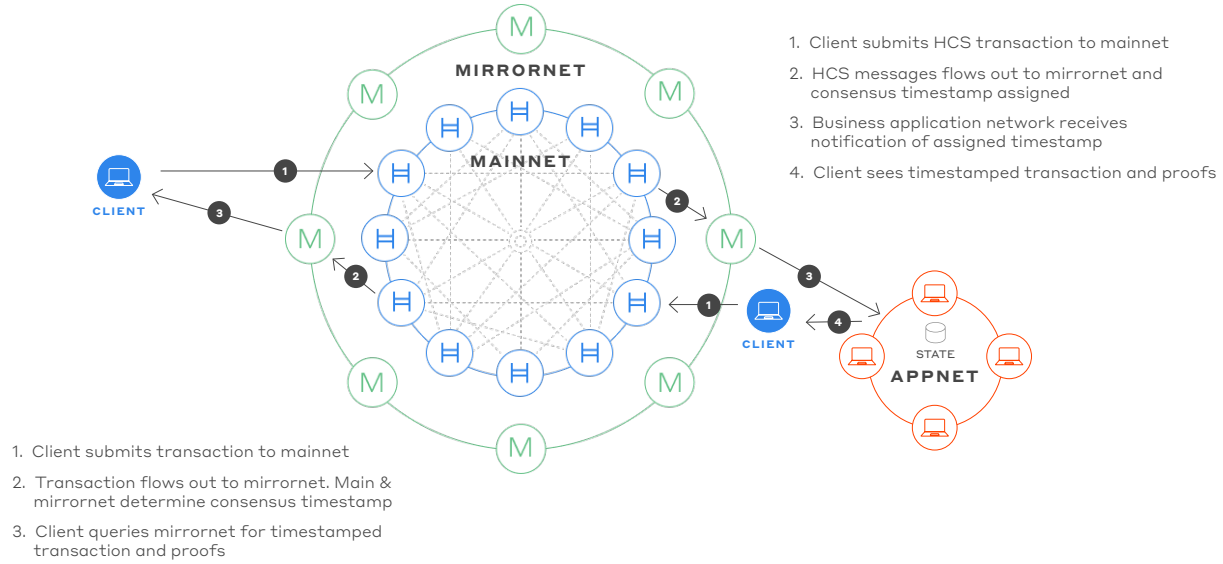
such personal data from, other locations. In general terms, the recipient of such personal data must be under a legally binding obligation to follow GDPR data protection principles or their equivalent.

On a typical public DLT network, the network data will be distributed and duplicated across all or many geo-distributed nodes. That is, of course, a key value proposition of a DLT network, namely that no small number of actors are solely entrusted with network data. Consequently, it may be difficult or impossible for a developer to ensure that personal data does not move to a particular geographic region, particularly if the DLT is permissionless and allows anyone to run a node.

HEDERA CONSENSUS SERVICE

The Hedera Network is a public distributed ledger technology (DLT) built on the hashgraph consensus algorithm. The technology is governed by a council of large, global enterprises that provide a mature and stable platform to the enterprises and distributed applications that build on the network.

The Hedera Network contains four services – cryptocurrency, file storage, smart contracts, and a consensus service. As the service of choice for many enterprise developers, we focus here on the Hedera Consensus Service (HCS) and its applicability to certain principles of the GDPR.



In the HCS model, a transaction logically flows through the main Hedera Network out to a business application network, in the process being assigned a consensus timestamp and order. The members of that application network process the transactions in that same order, and thereby ensuring that all members of the application network can maintain a consistent, synchronous representation of some application state.

In Step 1, a client submits a transaction to a mainnet node. The client is able to encrypt the transaction such that only authorized parties will be able to read it. This node submits the transaction to the rest of the network.

In Step 2, the nodes of the mainnet then assign that transaction a consensus timestamp and order within a particular HCS topic. The mainnet nodes store the encrypted transaction for only a very short time, approximately three minutes.

The transaction also flows out to the mirror network, as shown in Step 3. The mirror network is a parallel network of nodes, designed to remove from the mainnet nodes the burdens of storing transaction history and responding to queries from clients, and so allowing those mainnet nodes to be optimized for calculating consensus timestamps at high throughput. A mirror network node may choose to store a history of all transactions or only some transactions.

If the transaction was initially encrypted, neither the nodes of the mainnet nor mirror network will be able to read it. They can nevertheless still assign it a consensus timestamp and order, and add it to a historical record, respectively.

In Step 4, the transaction flows from the mirror network to whatever business application networks have subscribed to the topic. It is the members of the business application network that have the decryption keys necessary to read the business data within the transaction. Once decrypted, the members of the business application network then process that business transaction and update their local copy of state, as shown in Step 5.

In the HCS model, the nodes of the three different networks play different roles:

- **Mainnet nodes**
  - Determine consensus timestamp and order for transactions
- **Mirror network nodes**
  - Calculate consensus timestamp and order for transactions (not currently enabled)
  - Optionally persists history of transactions
  - Support APIs by which HCS messages can be subscribed to and retrieved
- **Business application network nodes**
  - Maintain state for relevant transactions
  - Run business processes on state
  - Support APIs by which clients can query the application network state

A given computer could play the roles of both mirror network node and business application network node; that is, a business application network node could receive HCS messages directly from the mainnet and not via a mirror node hosted by some other entity. A particular business application network might have one or more of its members act in this capacity.

A business application network could itself be a private permissioned ledger, relying on HCS for fast, fair and secure transaction ordering through a public DLT, rather than using some consensus algorithm between the members of the private network. For instance, there exists a plugin to HyperLedger Fabric whereby a particular HyperLedger network can call out to HCS rather than use the existing consensus algorithms that Fabric supports. By essentially outsourcing transaction ordering to the Hedera Network via HCS, the smaller private ledger can benefit from the more decentralized trust and governance of Hedera as well as the improved security, performance, and fairness of the hashgraph consensus algorithm. Similarly, a Quorum or Corda deployment could leverage HCS in lieu of those framework's own integrated consensus algorithms.

We explore below how the above separation of duties between mainnet, mirror network, and business application network nodes that HCS enables can help facilitate compliance with certain data privacy regulatory principles while still providing valuable decentralized trust to business application networks.

HCS AND DATA PRIVACY COMPLIANCE

It is generally acknowledged that data privacy regulations, such as the GDPR, can be more readily reconciled with DLTs by using a private and permissioned DLT rather than a public permissionless DLT, as the relatively small number of known participants simplifies the compliance challenge. However, the smaller number of participants in a permissioned DLT can generally not provide the same level of decentralized trust as a permissionless model.

Consequently, another emerging trend is to use a private permissioned ledger, but also record key snapshots of the private ledger's state to a permissionless public ledger. The private ledger makes more practical the GDPR's requirements of identifiable and legally responsible entities storing personal data in a secure manner consistent with the wishes of the corresponding data subjects, while the public DLT provides cryptographic protections against inappropriate manipulations of the stored personal data by those entities.

HCS is a concrete manifestation of the above general direction, enabling an architecture where private permissioned business application networks can store, process, and share personal data among themselves, but with the trust, transparency, and provability provided by the larger, public, and more decentralized Hedera Network.

By placing the burden of maintaining application state on business application network nodes and not the Hedera Network, the HCS model keeps raw personal data where it belongs – on the computers of those businesses with a justifiable need for its storage, analysis, and sharing.

Finally, it is important to note again that these methods are only part of a complete data privacy compliance program, and their effectiveness may depend significantly on the applicable regulations and specific characteristics of the business application network. For example, the GDPR defines regulated personal data to be any information relating to an individual who can be directly or indirectly identified; and, conversely, where an individual can no longer be identified from data, that data is no longer considered to be personal data. However, the extent to which any technique sufficiently reduces the risk of re-identification or linkability to a particular individual so as to refer to the resulting data as "anonymous" depends on a range of factors, including the underlying data and the application of the relevant technology.

How HCS Addresses the Challenges

We identified above two challenges that data privacy regulations present for DLT network – data mutability and data residency. We discuss below how HCS and the model of permissioned business application networks can help to address those challenges.

DATA MUTABILITY

The fundamental challenge for supporting a user's right to demand modification or erasure of their personal data if maintained on an immutable chain is not that it is impossible to delete a given piece of data, but rather that doing so would prevent subsequent validation of the chain.

In the HCS model, the personal data is encrypted before being submitted to HCS. It will not be persisted on the mainnet nodes and will be persisted in that encrypted form on whichever mirror network nodes keep the messages.

It is true that deleting messages from the mirror network node would invalidate the integrity of the topic running hash. In that sense, the data that the mirror network nodes store is "immutable," that is, deletion of one message would prevent external parties from subsequently validating the history and integrity of the list of messages sent to

the corresponding topic.

But the mirror network nodes do not have the keys necessary to decrypt the personal data, which are held only by the business application network members. If business application network members delete the keys that were used to encrypt the personal data, then, even if the encrypted personal data still sits on a mirror network node server, it is deleted in effect from the network as it can no longer be accessed by any participant.

Consequently, there is no need to delete the encrypted messages from the mirror nodes, with consequent invalidation of the relevant running hash. In the HCS model, the copies of the personal data persisted by the members of the business application network are mutable and therefore support the GDPR's erasure requirements, while the encrypted copies persisted on the mirror network nodes remain immutable.

DATA RESIDENCY

With the HCS model, business application networks may exercise control over both the location of raw personal data and possession of the necessary cryptographic keys that would allow access personal data communicated via HCS messages. By permissioning such business application network, only computers in jurisdictions chosen by the business application network would be granted membership and the ability to access or view the unencrypted data.

The encrypted HCS messages will have been communicated over the internet to main and mirror network nodes that may not (indeed, likely will not) be in the same jurisdictions, as is true of messages communicated over IP networks in general.

How HCS Enables New Compliance Mechanisms

HCS can also enable a range of additional functionality that creates new mechanisms consistent with common principles in data privacy regulations of user empowerment and transparency and are potentially not possible on DLT platforms other than the Hedera Network. We list three such examples below.

DECENTRALIZED IDENTITY

New data privacy regulations are causing businesses to reconsider their default model of aggressively collecting and storing user personal data. Decentralized identity, sometimes referred to as self-sovereign identity, enables a model where users are given greater control and responsibility over the storage and management of their own personal data, and in so doing shift that burden (and associated costs and risk) away from businesses.

This model can help to remove those authorities that issue identities (e.g., governments, employers, universities, social providers, etc.) from necessarily being involved in each engagement between the user with a relying party (as was the case with the federated identity model), and so give to the user greater autonomy and control over their identity and credentials. In this model, authorities still issue identities, but the corresponding credential is stored and managed by the user, and so disintermediates the authority from real-time participation in the use of that credential.

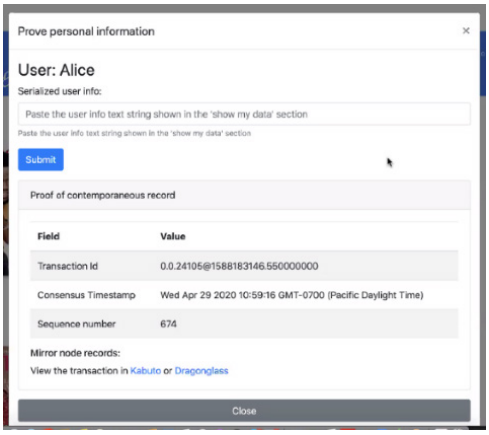
These new identity architectures often advocate using a public decentralized blockchain or DLT to anchor identities – specifically to store artifacts for those identities like pointers, hashes, and metadata for identities on a public ledger to facilitate subsequent verification and discovery by the relying parties at which the identity is presented.

The Hedera Network can enable decentralized identity with HCS. In this architecture, identity artifacts are submitted to HCS for timestamping and ordering before being persisted in a business application network. Hedera is participating in the W3C Decentralized Identifiers Working Group to standardize this architecture. Hedera has authored a DID Method to normalize how such decentralized identifiers can be managed on an HCS business application network.

CONSENT RECEIPTS

A consent receipt is a structured document that provides detail about such consent grants. A receipt would typically specify what was the personal data for which consent was obtained, the reason for its collection, and any limitations on storage and usage. For a business, a consent receipt, signed by both parties provides a consistent firm legal ground for data processing. For the data subject, a consent receipt provides a new mechanism for keeping a clear record of what data about them a company has been given, how that data is being used, and any other business it is being shared with.

If such a consent receipt were timestamped by a public ledger like the Hedera Network, then the time at which consent was collected could be more reliably determined in the future, should there be a dispute. As a concrete example, a data subject might be able to prove that while they did indeed give consent to some particular use of their personal data, that consent was given only after the first instance of that usage, and so the action of the data processor was not authorized.



In the HCS model, after a data subject authorized a particular use or sharing of some piece of personal data, a consent receipt capturing the specifics of that authorization would be submitted to HCS where it would be assigned a timestamp. Any subsequent modifications of that consent, whether modifying the terms or perhaps even its deletion to indicate withdrawal of that consent would also manifest through HCS messages. Subsequently, if there was a dispute over some usage of the personal data and whether it was consistent with the terms of the data subject's consent at that moment in time, this history of consent would be retrieved and validated.

This graphic is a representation of such a consent receipt. A user, Alice, is able to view the details of the consent she has granted to a website. Importantly, the fact of that grant of consent, but not Alice's specific details, was timestamped by HCS and could be validated against public mirror network nodes should Alice and the website have a disagreement over that consent or its later withdrawal.

PROVABLE DELETION

Another common principal found in modern data privacy regulations is the principle of accountability, that is, the entities in control of personal data must be able to demonstrate compliance with their data privacy obligations.

For example, as discussed above, the GDPR stipulates that a data subject has the right to request that a given piece of their personal data be deleted from a database. Upon receiving such a request from a data subject, and performing the necessary deletion of the personal data from its systems and servers, the business may need to prove to the data subject that the personal data was indeed deleted.

It is possible for a business to demonstrate that they followed a particular process around data deletion and some of these steps could be made provable via timestamping of HCS messages. The business will log key steps of their business process by sending appropriate messages to an HCS topic and archive the resultant transaction records retrieved from the mirror network.

1. Data subject files a request that the Processor delete a specific piece of personal data;
2. The business sends an HCS message indicating that it acknowledges the request and is committing to a specific date and time for deletion;
3. The business marks their copy of the personal data as 'to be deleted' and sends a hash of that data structure via an HCS message;
4. The business deletes the personal data;
5. The business sends an HCS message indicating that they did indeed delete the personal data.

The subsequent stream of timestamped and ordered messages are stored and made available – transparently demonstrating to both the data subject and regulators that the business acted in good faith when the data subject requested deletion of their personal data.

EXAMPLE SCENARIO

To make concrete how HCS and the business application network model can support GDPR compliance, consider a group of university research labs that wish to share patient health data as part of a clinical drug trial.

The universities will use HCS as the messaging channel by which they share the patient data – in other words the universities will create and participate in a business application network on the Hedera Network. By sharing data via HCS, all participants can be confident that they are indeed seeing the exact same dataset at a given moment in time without needing to trust any one university to manage that dataset in a centralized manner.

The universities will all agree to abide by a governance framework for the patient data, outlining allowed applications of the data, the patient's rights with respect to their data, etc.

An administrator will create an HCS "topic" on the Hedera's mainnet. Messages subsequently sent to this topic by the different universities will be timestamped and assigned a place within the topic specific order. The administrator will create the topic with a list of public keys corresponding to each university. Only messages signed by the corresponding private keys will be accepted by the Hedera's mainnet nodes and processed into consensus.

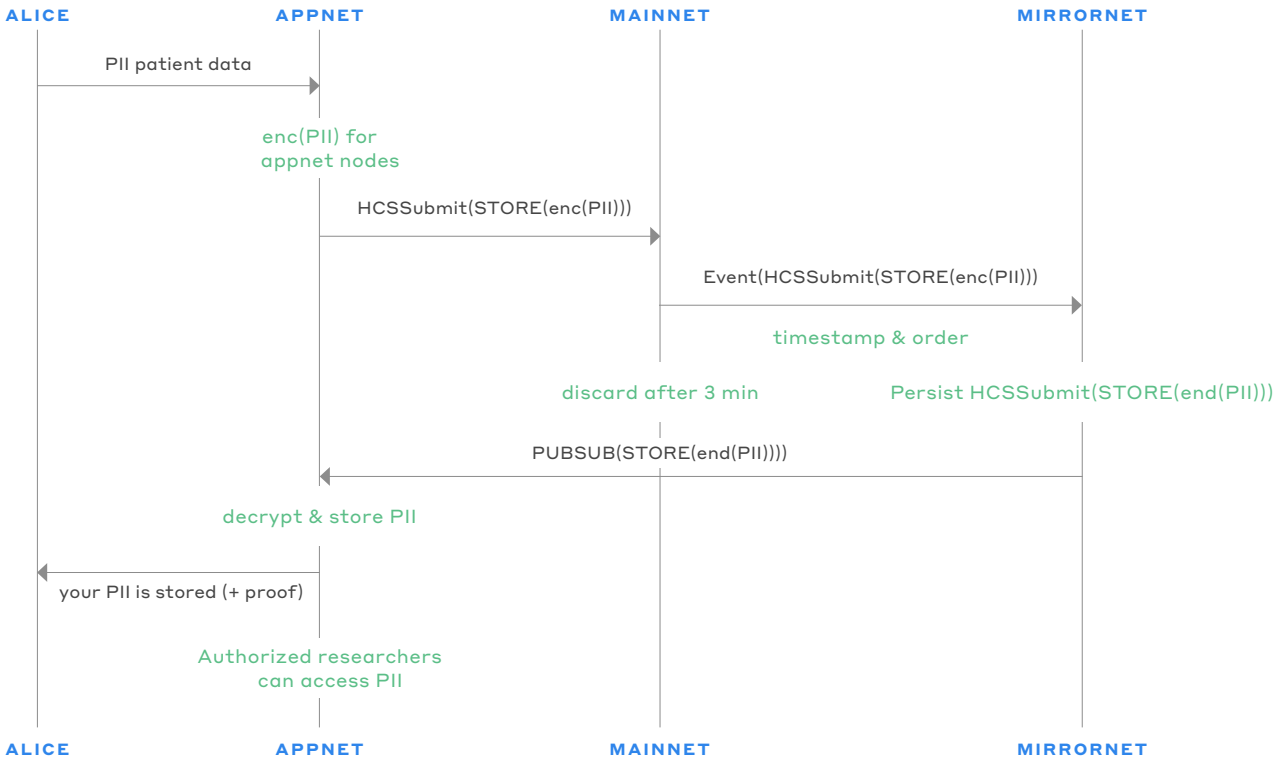
The universities will also exchange amongst themselves cryptographic keys. All messages sent via HCS will be encrypted such that only authorized universities can access the patient data.

Patient data will be distributed to the universities via an encrypted HCS message carrying the data. The message will be sent to the Hedera mainnet and then flow to the mirror network as like any other HCS message.

The universities will subscribe to that topic at a mirror network node. Whenever a message is submitted to that topic and received by a mirror node, the subscribed universities will also receive the message. Those universities that are able to decrypt the messages will do so and then store that data such that researchers can access it.

As only the universities themselves will store the decrypted patient data, the requirement that the data stay in the EU can be satisfied (assuming all universities are in the EU). While the Hedera mainnet and mirror nodes may not necessarily also be in the EU, those nodes will either not store the data, or store it in only in an encrypted format that they cannot access.

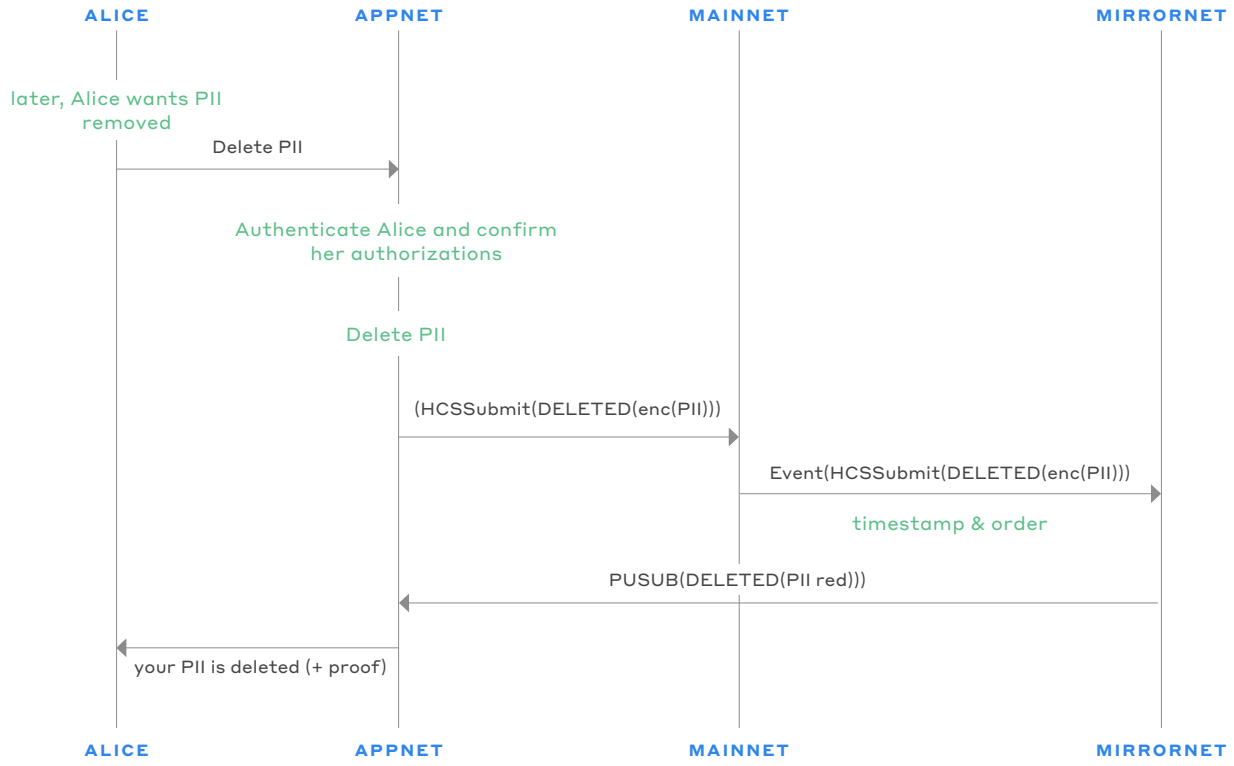
This sequence is represented below for a specific patient Alice that logically gives her personal data, labeled as personally identifiable information (PII), to the universities in support of their research.



NOTES:

1. Only authorized members of the business application network would be able to submit messages to the topic. No other party would be able to poison the University's data set.
2. The mainnet nodes discard the encrypted message soon after it is assigned a consensus timestamp and place within the topic's order.
3. The mainnet and mirror network nodes maintain a running hash and sequence number for all messages within the Topic. Any party can query a mirror network node to validate that a given message was sent to the topic and when. And they can verify that no messages were tampered with and no messages were eliminated from the sequence of messages.
4. A mirror network node may store the history of messages (and the encrypted PII within) but, without the decryption keys, would be unable to decrypt them.
5. Business application network members can choose to use different keys for each message, ensuring perfect forward secrecy. If a single key is hacked, then any message encrypted with that key could be read, but not other messages encrypted with different keys.

If, as is her right under the GDPR, Alice later decided that she wished her patient data to be deleted, she would be able to direct the universities to do so. As for the message that distributed her data in the first place, the directive to all universities to delete her data would flow through HCS.



NOTES:

1. The message containing the directive to delete Alice's personal data could be subsequently provable; that is, any subsequent dispute would have clear cryptographic evidence of that message being sent on a given date.
2. On receiving the directive to delete Alice's patient data, the universities would delete both the personal data and any keys that had been previously used for its encryption. Even if Alice's data still existed on a mirror network node, it would become inaccessible because it could never again be decrypted.
3. The universities could acknowledge receipt of the delete directive by themselves submitting a message via HCS. This timestamped ACK message could be used by a university in any subsequent dispute as additional evidence that they did indeed delete Alice's data when directed to do so.
4. Deletion of Alice's personal data and keys need not invalidate the cryptographic guarantees provided by the Hedera mainnet node's calculation of a running hash and sequence number for the business application network's topic – the "chain" of messages would remain. The mirror network nodes would continue to persist the encrypted personal data, and so be able to create cryptographic proofs even if the encrypted data would never again be accessible.

CONCLUSION

The Hedera Network and its Consensus Service can enable an architecture where private permissioned business application networks can store, process, and share personal data among themselves, but with the trust, transparency, and provability provided by the larger, public, and more decentralized Hedera Network. This architecture provides significant advantages over naïve implementations of DLT that developers can leverage to build compliant dapps on a distributed platform.