BB-FLoC: A Blockchain-based Targeted Advertisement Scheme with K-Anonymity

EDVINAS KRUMINIS, KEIVAN NAVAIE, and ONUR ASCIGIL, Lancaster University, United Kingdom

New data protection regulations, e.g., General Data Protection Regulation (GDPR), enforced advertisement providers to amend their conventional approaches, enhancing users' data privacy. As a result, major Internet browsers such as Apple Safari and Firefox were quick to announce their plans to remove third-party cookies from their browsers entirely. Google, in an effort to preserve conventional advertising practices, proposed a system of Federated Learning of Cohorts (FLoC) as a way of delivering higher privacy guarantees to users whilst also providing interest-based advertising. In FLoC, users sharing similar browsing histories are put into cohorts, and thus advertisements can be targeted to them as a group, rather than individually. Since each user independently calculates their cohort group, a minimum cohort size cannot be enforced, making them vulnerable to identification and tracking. To address this issue, in this paper, a blockchain-based FLoC (BB-FLoC) system is proposed that guarantees k-anonymity for its users whilst at the same time allowing for effective personalised advertising. We further evaluate the operational feasibility of such a design and demonstrate that in the proposed system, k-anonymity guarantees can be fulfilled in a fully decentralized manner. The proposed system is relatively lightweight, showcasing that it can be adapted for low-end devices such as mobile phones.

CCS Concepts: • **General and reference** → *Design*; • **Security and privacy** → **Privacy protections**; • **Information systems** → **Online advertising**; • **Computer systems organization** → **Distributed architectures**.

Additional Key Words and Phrases: Security and Privacy, Blockchain, Distributed Systems, Advertising

ACM Reference Format:
Edvinas Kruminis, Keivan Navaie, and Onur Ascigil. 2018. BB-FLoC: A Blockchain-based Targeted Advertisement Scheme with K-Anonymity. In . ACM, New York, NY, USA, 17 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Data Privacy is becoming an ever more contentious issue in today's society. Users spend hours browsing the Internet and all the while their digital footprints are inferred by third-party cookies which track their browsing history. The data collected can range from the exact websites a person visited, the contents of the page, and to their actions/time spent on each site. This information can then be monetized by displaying targeted advertisements for them. For example, a user who recently read travel articles may be shown advertisements (ads) on another site with flight ticket deals. This can have a huge impact on user privacy as their individual browsing history can be collected to build an exact profile of the user, including sensitive details such as their race, health, religion, etc.

The main concern with the conventional third-party cookies-based approach is its inherent centralized architecture making users vulnerable to identification and tracking. In combination with the increasing prominence of the topic of privacy among the general public, and the passing of privacy directives such as GDPR, advertisement agencies have been forced to adapt their approaches [22].

Manuscript submitted to ACM

As a potential alternative to third-party cookies, Federated Learning of Cohorts (FLoC) was introduced by Google in early 2020. The key idea behind FLoC's approach is to group people with similar browsing histories into cohorts, and in doing so, protect their privacy by hiding individuals in a (large) crowd with related interests [2]. Although the project was abandoned by Google in January 2022, little explanation was provided as to why, and no analysis was published regarding the results obtained from their public trials.

With FLoC, rather than submitting a unique identifier in each website request (such as a personal cookie), users would respond with a 'cohort ID'. This ID is calculated independently and without any communication with outside sources by using a predefined pseudo-random seed. A locality-sensitive hash (LSH) function is applied to generate a vector point, which is then mapped to a center point that is shared by homogeneous parties. Users would use this hash to get assigned to a cohort group which is populated with others sharing similar browsing interests (depending on the accuracy of the LSH function), ultimately prohibiting the tracking of individuals across the web. There are, however, two main concerns regarding the *quality* and *privacy* of FLoC:

(1) Quality: Users within the same cohort can have diverse interests between them as the similarity between them might be weaker than that of centralized approaches, hence reducing advertisement efficiency.
(2) Privacy: Since each user independently calculates their own cohort group, a minimum cohort size cannot be enforced across the whole domain. Users may be mapped to a small cohort, making them vulnerable to identification and tracking, defeating the whole purpose of the procedure.

Current research attempts in tackling the issue regarding privacy have mainly consisted of implementing centralized servers that track the size of each cohort [2]. In turn, they would either block browser calls within cohorts that do not satisfy the k-anonymous principle (i.e., SimHash), or by post-processing the cohorts to ensure k-anonymity (i.e., SortingLSH). In both of these approaches, a centralized server requires access to users' original browsing history, potentially violating their privacy. Additionally, the servers themselves are at risk of malfunction and attacks, severely threatening the functionality of the entire service.

To address the above issues, in this paper we examine the viability of using blockchain as a means of ensuring that k-anonymity in FLoC is fulfilled in a completely decentralized manner, without the use of any centralized servers. We propose a blockchain-based FLoC (BB-FLoC) system in section 3, providing a comprehensive examination of its design, and engage in a discussion regarding the design choices. Our approach involves anonymously collecting cohort hashes and reorganizing them to attain the k-anonymity threshold. Users are then able to compute their own k-anonymized cohort group based on the latest block contents, and use it as their ID on website request. In section 4, we describe the implementation of our design, outlining its specifics and conducting simulation experiments to evaluate computational complexity, storage requirements, and the utility of the approach. We also introduce the idea of a "lightweight mode" in section 4.5, to guarantee that our design can be accessible to a broad range of users, regardless of their computational resources.

1.1 Contributions

The contributions of this paper are summarised as follows:

(1) To the best of our knowledge, we are among the first to propose a design of a system that can provide privacy-preserving targeted advertising with blockchain technology at its core. The blockchain parameters and model

specifics have been discussed, and their rationales clarified. We have also provided details of full-node and light-node implementations to enable even low-end devices to participate in our system under minimal computational strain.

(2) We have constructed the first functional implementation of FLoC that is able to enforce k-anonymity across the domain in a fully decentralised manner, with no central entity. We have conducted computational tests of such a design and analysed its results, showcasing that the model is able to achieve k-anonymous privacy guarantees.

2 RELATED WORK

Given that the inception of FLoC is relatively new, few research works have been published examining its effectiveness. It must be noted that although the naming of FLoC alludes to the use of "Federated Learning" in its approach, there is in fact no machine learning or artificial intelligence involved in the process; it is believed that Google intended to use some form of federated learning in its future iterations, however, with the project being abandoned, these plans never materialised. The original FLoC whitepaper [2] published by Google mainly focused on introducing the concept as a whole; presenting the idea of cohorts, and describing how SimHash/SortingLSH works. Their work used public datasets to assess the utility of cohorts, and found that the decentralised clustering algorithms achieved at least 95% of conversions per dollar spent when compared to that of cookie-based advertising. They further showed that FLoC produces significant improvements over random user groupings.

2.1 Blockchain and GDPR

For an easily accessible introduction into blockchain basics, we refer the readers to [28]. Due to its decentralised and anonymous nature, blockchain technology has shown promise as a tool for privacy. It has been used in the context of social media [32], healthcare [31], Internet-of-Things [13], among many others. However, the introduction of recent privacy measures (such as GDPR) has thrown into dispute whether blockchain is compatible with such data protection regulations. For example, one of the key aspects of GDPR is article 17, the 'right to erasure', where users can request to have their personal data to be deleted, which is in direct contrast with the immutability and 'append-only' manner of blockchain data.

Blockchain does not inherently have the ability to delete data that was previously stored; alternatively, it introduces the concept of 'burning', where encrypted data can be transferred to an inaccessible account lacking accompanying decryption keys. Consequently, such data becomes inaccessible for further access or transfer. It is important to note that the exact terms of "erasure of data" is not defined in article 17 of GDPR. The compliance of encrypting data without retaining encryption keys is a subject of debate [19]. Additionally, confirming the deletion of encryption keys poses an additional challenge.

Another possible approach to align blockchain with GDPR involves storing data 'off-chain'. In this scenario, the blockchain functions as an access control point, storing a link to the data in the chain while the actual data resides outside the blockchain (e.g., in local databases). Deleting personal data becomes straightforward in this setup, as it can be removed from the local database, rendering the links to the data in the blockchain invalid. However, this comes at the cost of decentralisation, as control over off-chain data falls under a specific authority. In conclusion, while solutions exist for making blockchain potentially GDPR-compliant, combining both approaches will likely require making some concessions on either end.

2.2 Public trials

Following the announcement of FLoC, many in the public were quick to criticise [7, 10, 27]. They claimed that cohort IDs can still be used for tracking through practices such as browser fingerprinting and that FLoC has the risk of revealing sensitive information about its users, such as their access to sensitive topics (e.g. gambling, medical information), and their demographics. Major browser developers, such as Edge, Firefox, Safari, all declined to participate in the implementation of FLoC [1]. Despite the public concerns, Google pressed ahead with their proposal, running its original public trial in mid-2021 [20], before publicly announcing the end of its development in early 2022. Instead, they introduced a new proposal, Topics API, citing feedback received from the community and the earlier FLoC trials. With it, rather than using the domains of visited websites, interest-based advertising is served based on a taxonomy of categories of websites that a user frequently visits, e.g. "/Business & Industrial/Agriculture & Forestry", "/News/Mergers & Acquisitions", etc. [17].

Minimal data or analysis was published after the conclusion of FLoC trials, leaving it out in the open as to how viable FLoC really is. What we do know, is that the trial was ran on Chrome versions 89 to 91, and incorporated multiple clustering algorithms, presumably in an attempt to find out which one performs best. Google set a value of 2,000 minimum users for a valid cohort group, and found that the number of LSH bits used to define a cohort was between 13 and 20, considerably less than the 50-bit that was produced by their algorithms. A total of 33,872 cohorts were produced, 792 of which needed to be filtered due to their alleged sensitivity; such a cohort group either contained too few qualifying users, or the users inside the cohort had a high rate of visits in websites containing sensitive topics, e.g. pornographic websites. It was never shared how many users were a part of such trials, though [3] estimates a total of 0.5% Chrome users in Australia, Brazil, United States and other select regions were affected. It must be noted that Google never tested FLoC on users in the European Economic Area, presumably due to their concerns for its compliance with GDPR [25].

2.3 Privacy concerns

A paper published by Berke and Calacci [5] conducted an empirical analysis of FLoC, and attempted to analyse some of the concerns previously raised by privacy advocates. Their work confirmed the beliefs of those that presumed that FLoC still enables individual tracking; they discovered that after as little as 4 weeks, 95% of user devices can be uniquely identified. Work by Turati et al. [30] examined potential attacks on systems that use locality sensitive hashing, and showcased ways to reconstruct a portion of the private data from LSH hashes. With respect to FLoC, they were able to reconstruct 10% of browsing history of 30% of its users based on only their FLoC hash. Despite that, analysis from [5] showed no correlation between cohort IDs and racial background, refuting the claims that FLoC can leak sensitive information about its users.

FLoC can in theory make fingerprinting a lot easier to do; this is because rather than trying to distinguish between millions of users, trackers would only need to go through several thousands, if not just hundreds of users [10]. Additionally, LSH functions are unlike cryptographic hash functions, and thus do not possess the properties of preimage resistance - malicious parties can exploit this by creating browsing histories that can be attributed to a target cohort. However, compared to individual tracking approaches such as third-party cookies which track users individually, FLoC displays a 'category' of user history; therefore, the information that can be gathered about individuals is only probabilistic. Besides, the tracking of users that was achieved by [5] was done through the storage of cohort ID sequences and cross-site tracking - when used in totality, a k-anonymous cohort cannot be used as a fingerprint [2].

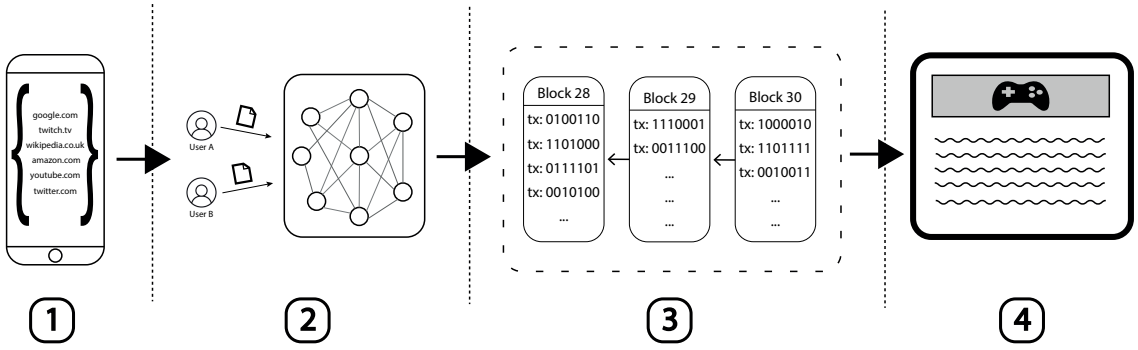

Fig. 1. The BB-FLoC process for displaying targeted advertisements includes the four steps as illustrated in the picture.

3 SYSTEM MODEL

Our main objective here is to evaluate the practicality of using blockchain as a means of decentralising the entire FLoC process. We will first provide a general overview of our proposed design before delving into more detail in later sections.

3.1 Blockchain-Based FLoC Process

The BB-FLoC process is illustrated in Fig. 1, with the following participating parties:

- *User* submits a cohort hash to the network and uses the mined blocks to calculate their cohort group ID. Anyone with an internet connection can become a user of BB-FLoC.
- *Miner* collects a list of valid cohort hashes and adds them to blocks which are propagated across the network. All participating users have the ability to become miners.
- *Ad Agencies* inspect activities of cohorts whilst they are browsing the Internet and use this observed information to display advertisements that could be relevant to users within a certain cohort group ID.

In step 1, the process begins with the user gathering their own browsing history, perhaps for a particular time period such as a week, and using a locality-sensitive hash function to produce a 50-bit cohort hash value that represents their activity. Note that the sites visited by users in incognito mode are not collected. Similarly, websites have the ability to choose to get excluded from the FLoC process, and as such, only participating domains are going to be included in the list for hashing. Only the top-level domain names (eTLD+1) are used, therefore, specific browsing activities (e.g. the articles that were read in a news website) are not used as parameters in the function. After computing their cohort hash, step 2 involves the user creating a transaction object with a portion of their hash and sending it off to the blockchain network, which is then collected by miners and later added to blocks. For the full structure of the 'block' and 'transaction' objects, refer to Figure 3.

As blocks arrive in the network, the participating users individually verify them and, if accepted, place them onto their active chain, as is visualised in step 3. With each new block, users (re)calculate their own cohort group ID by collecting all the currently valid cohort hashes, and, using a cohort reorganisation technique to satisfy k-anonymity guarantees, individually assign themselves to a cohort group. We define a cohort group simply as a collection of users with similar browsing interests. Finally, when visiting a FLoC-participating website, a user submits their cohort group ID value in response to being asked for a cohort ID. This allows the site contact to display relevant advertisements to them; in Fig. 1, this is represented as a "gaming" ad in Step 4.

Table 1. Comparison between Google's FLoC approach, and our Blockchain-Based FLoC design.

	FLoC	BB-FLoC
Decision making	Centralised (Google)	Decentralised
Cohort ID publication	Full 50-bit hash is sent for further cohort reorganisation	Users decide on what substring of the 50-bit hash they wish to publish
Cohort reorganisation	Centralised servers collect cohort hashes and assign users to a group	Users reorganise themselves into a k-anonymous group independently
Data collection period	7 days of user browsing data	24h of user browsing data
Cohort group updating	Cohorts groups are re-updated every 7 days	Cohort groups are re-updated with every block published (1h on average)

3.2 FLoC vs. BB-FLoC

One of the key drawbacks of the original FLoC implementation is that it doesn't inherently guarantee a minimum cohort size. Considering that the idea of FLoC is to hide users in a large group, rather than targeting them directly, we believe this to be a significant flaw. Current FLoC implementations that provide a minimum cohort size guarantee also require a centralised server to collect all the hashes and reorganise them in a way that satisfies the k-anonymity principle. This requires users to put trust into the entity that their privacy will be protected and the process of cohort reorganisation will be faithfully executed. Furthermore, the centralised servers themselves are at risk of attack. They may return incorrect responses to users or be unavailable altogether. We have highlighted the major differences between Google's implementation of FLoC and our BB-FLoC design in table 1.

3.3 Cohort Reorganisation

There are several cohort reorganisation approaches that can be adopted, each with its own pros and cons. Here we advocate for the cohort sorting algorithm *PrefixLSH*, which was used by Google in their real-world experimentation of the system [15], and has the additional quality of being scalable.

PrefixLSH is a locality sensitive hashing algorithm that assigns similar vectors to closely matching inputs. The main advantage of this approach is that the cohort IDs that are related by some factor(s) can be calculated without any communication with other parties in a completely decentralised manner. The only parameters in this algorithm that need to be shared between users is the details of pseudorandom number generation [20]. The algorithm takes in data as an input (domain names, songs listened, etc.) to produce an n-bit locality sensitive hash bitvector, "*where the i'th bit indicates the sign (positive or negative) of the sum of the i'th coordinates of all the floating-point vectors*"[20]. The resultant vector is the unique cohort ID of a user, however, to ensure k-anonymity, these IDs need to be further reorganised. We illustrate an example of this procedure in Fig. 2:

(1) All the n-bit hashes in the network are separated into two cohorts, i.e., those whose first bit is 0, and those whose first bit is 1.
(2) Both of these cohorts are further partitioned by looking at their successive bit, again splitting each cohort by those whose i'th bit is 0 or 1.
(3) The process at Step 2 is continued until it no longer satisfies the minimum cohort size threshold. The final remaining level is the cohort ID of that user.

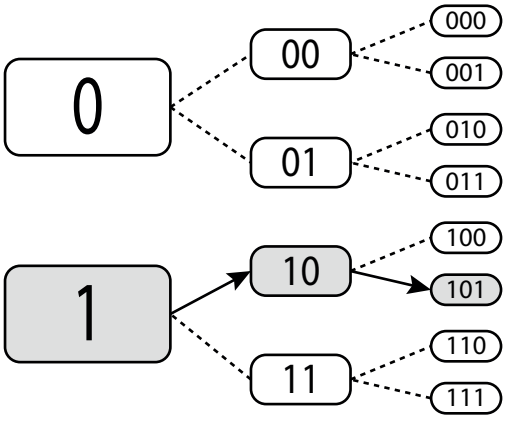

Fig. 2. An example of PrefixLSH, where a user with a cohort hash of *"10110"* would continue along the chain until the minimum cohort size threshold fails, ultimately being assigned to a cohort group ID of *"101"*.

Rather than sending in their full 50-bit hash to a centralised server for reorganisation, in our approach, users distribute only a portion *(50-n)* of it to miners in the blockchain system. The responsibility for deciding the level of risk a user is willing to accept lies with each individual user; indeed, in Google's trial [20] with a *k-value* of 2000, only between 13 and 20 bits were needed to form a valid cohort group, so there is no need for users to fully share the entire hash and expose more information about themselves than needed. Once the user has sent out their hash, they can fulfill the *k*-anonymity principle by reordering the cohorts in lexicographic order. In light of transferring the decision-making back to the individual, the usage of blockchain introduces further benefits. Rather than just following global system parameters, users can independently calculate their own cohort group ID and even decide on their own *k* value based on their particular risk management strategies.

3.4 K-Anonymity and Cohort Sensitivity

Deciding on an appropriate k-value is a complicated task. Ad agencies and individual users are directly at odds with each other; privacy-conscious users would ideally wish to keep the *k* value as high as possible (*k=n*), whilst ad providers would preferably provide individually targeted ads (*k=1*). Clearly, either end of the spectrum is not feasible - one gives no utility to ad agencies, whilst the other provides no privacy to the users. Therefore, a fair balance must be reached between both parties. It should be noted that a high value of *k* by itself does not necessarily equate to strong privacy guarantees, and any such value should in part be based on the numbers of users (or datasets) in totality. Additionally, guaranteeing a zero-percent chance of re-identification is a task that even GDPR legislation has acknowledged is not viable.

As we are dealing with browsing history records in our research, such data can be quite sensitive, potentially dealing with information regarding an individual's medical history, political views, etc.; accordingly, high privacy guarantees are a must. Cohort IDs related to some demographical information may also cause users to be subject to predatory disinformation campaigns and/or price discrimination. Sensitive information is important to protect, however, what is sensitive to one group of people can be acceptable to others, societal norms may change and evolve over time, etc. Therefore, such an issue is difficult to define. Google, in their experiments, suggested dropping cohorts entirely if they are found to have a high rate of visits to sites displaying sensitive topics [20]. However, this approach does not prevent

misuse entirely. Due to the complex nature of this problem, here we deem it to be out of scope of our paper, i.e., all possible cohort IDs are deemed *appropriate* and accepted as valid in our model. Users that are particularly concerned about such privacy risks can always choose to opt-out of the service completely and receive generalised advertisements instead.

3.5 Utility

On the most fundamental level, the higher the k-value (and thus fewer cohorts), the more protection is provided to a user as their identity is obscured to a greater extent. At the same time it also leads to a loss in utility for advertisement agencies, as users are more likely to have diverse interests between them whilst being within the same cohort group. Whilst our algorithm (*PrefixLSH*) produces a cohort hash of 50-bits, as mentioned previously in section 3.3, users are free to decide on the size of the subsection of the hash that they send out in their transaction object to the network.

Since all users in the system must submit some form of their cohort hash to the network, the number of users within a certain cohort group can be inferred by ad agencies. This does not necessarily mean that advertisers will know the exact size of each cohort; the cohort hash sequence that users publish to miners, and the cohort group ID that they use to identify themselves whilst browsing the internet are two separate values. This is because each user can individually pick their k-anonymity value, and some users will be more risk-averse than others. Therefore, from the perspective of ad agencies, the longer the size of the cohort group ID (i.e. the greater number of bits), the more specialised is its demographic, and we suggest the following function to quantify the utility of a cohort ID:

$$U_{C(id)} = \frac{x}{\log_2(n)}, \tag{1}$$

where n is the total number of users in the system and x is the number of bits of the cohort ID. A higher value of U equates to greater accuracy for ad agencies as the advertisements can be more precisely tailored. We use the logarithmic value of n to normalise the effect that a large number of users would have on our function.

The utility function to be used in our analysis is independent of the users' k-value, and in any case, such a number is never to be publicly disclosed. Ultimately, the k-value itself is insignificant to ad agencies as they are more interested in dealing with how specific a certain cohort is (and thus how personalised the ads are to be), not whether the users of such a cohort are adequately disguised.

3.6 Incentivisation

Perhaps the most basic question that must be addressed in our design is the following; *"Why shouldn't users just opt-out of any BB-FLoC system?"*. On the surface, users have no rational reason to devote their own computational, storage, and networking resources for the sole benefit of ad agencies, especially when they can request to receive generalised ads at no cost. Mining incentives are essential factors enabling thriving blockchain ecosystems; whilst this isn't the main focus of our paper, we can envisage several reasonable approaches that could be undertaken.

For many years, popular websites such as Google and Facebook have provided free and open access to their services - in return, the companies would collect data on users. Such data would then be used to predict users' behaviour, allowing companies to subject users with targeted ads, making the marketing process significantly more effective (and lucrative) than ever before. This interest-based form of advertising is a pillar of the ad-serving ecosystem, so we deem it imperative to try to preserve. Conversely, some websites offer users ad-free experiences in exchange for a subscription or one-off

Table 2. The parameters used in our BB-FLoC implementation.

Parameter	BB-FLoC configuration	Parameter governance
Access	Permission-less (peer-to-peer, open to all)	Network-defined
Consensus algorithm	Proof-of-Work	Network-defined
Block size	Unlimited	Network-defined
Block interval	1 hour	Network-defined
Cohort update	24 hours	Network-defined
Hashing algorithm	SHA256	Network-defined
Cohort sorting algorithm	PrefixLSH	Network-defined
Cohort hash size	50-bit hash produced, value sent in a transaction object can be shorter	User-defined
Min. size for a valid cohort group	2000 users	User-defined

payment, e.g. Spotify, YouTube. Other websites simply deny access to users who are found to use ad-blockers until they are disabled.

The relationship of paying for free access by means of personal data could have an evolution into one where users pay via their computational resources by opting into a BB-FLoC system and receiving targeted ads, now with the added bonus of privacy. To further enforce this, websites could deny access to users who are not part of the BB-FLoC process, forcing them to either join the ad-serving regime, or pay a subscription. We can further imagine a system where users are (monetarily, or otherwise) rewarded for actively participating and/or mining new blocks. Displaying targeted ads can result in higher revenue for websites over more generalised ads; conceivably, websites wishing to participate in the BB-FLoC process and display targeted ads may be required to pay a set amount of cryptocurrency to successful block miners. All such fee-paying sites could then be placed inside a list of valid domains that have the ability to display BB-FLoC targeted ads. Lastly, ad agencies themselves have a rational reason to mine new blocks as otherwise cohorts hashes would expire and everyone would request for generalised ads. We leave the feasibility study of such an economic proceeding to future work.

3.7 Blockchain Parameters

The openness of blockchain allows users to re-adjust parameters of the system as they see fit. Since our BB-FLoC implementation is based on proof-of-work, each users' voting power directly correlates to their share of mining power; more powerful miners have a higher probability of mining a block, therefore, their vote carries more weight. For specifics of the parameters that were used in our implementation, see Figure 2. By "Network-defined" we infer that the parameters need a majority (51%) of collectively miners to agree, compared to "User-defined" where each individual user has the authority to decide. We must emphasize that the specifics listed by us are merely suggestions, and users of blockchain systems can come together and decide on different such variables, perhaps after assessing their computational/storage resource capacity or system ideals.

In Google's original trial [20], a subset of valid domains from users' history was sequentially collected over a period of seven days, i.e., cohort IDs changed every seven days. In our implementation, we suggest that a shorter block interval of one hour and a cohort hash recalculation of 24 hours work in the fair interest of the whole network. The values of one hour and 24 hours were based on Google's cohort reorganisation of every seven days; shorter intervals here allow us for a more up-to-date calculation of the cohort hashes, meaning that the displayed ads will be more relevant. It also

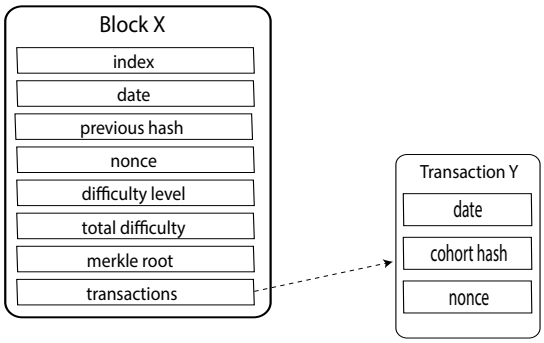

Fig. 3. The structure of *"Block"* and *"Transaction"* objects.

enables more frequent user-side updating of cohort groups, further safeguarding user privacy. We believe that distinct values of the block interval and cohort hashing serve an important purpose; a shorter block interval enables users to join the system at more regular intervals (rather than waiting for the full day-long cycle to run its course), whilst also curbing the possibility of network throttle that one large block with the entire network of users could cause on the system.

4 PERFORMANCE ANALYSIS

Here, we will now describe our implementation of a BB-FLoC system and evaluate its performance.

4.1 Architecture

Our fully-functional model is developed primarily in 'Java' programming language, and can be found on [24] as an open-source project with an easy to use .jar file. We closely followed the specifications of the Bitcoin [26] blockchain model, however, our BB-FLoC can be incorporated into any basic blockchain model. We used TomP2P [6] as our peer-to-peer component, which allows users to freely connect with others; the blockchain in BB-FLoC is non-restrictive and permission-less, allowing anyone with an internet connection to join the network. Any user can choose to be a miner of blocks, however, this is optional; users can exist on the network simply as block receivers only. Our blockchain-based FLoC model will use the proof-of-work consensus protocol, however, we must point out that other forms of consensus (e.g. Proof-of-Stake) are also applicable in our design. We specifically selected PoW due to its high degree of decentralisation and security guarantees [26], two aspects that we consider crucial for our use case.

The cohort reorganisation PrefixLSH done in 'Go' [18], and simply takes in as input a list of eTLD+1 domains collected in a JSON file. The output is a 50-bit hash that encompasses a users browsing history, and this process is repeated every 24 hours with the updated list of a users' browsing history. Blocks are created on average every hour, and upon receival of a new block, users recalculate their cohort group ID according to the latest state of the network and store this value on a file. This file would then be accessed by a browser that requests the user for their cohort ID. As this process would only happen whenever the user is actively participating in the system, users could end up taking longer than 24h to re-update the network with their cohort. Thereby, users in our system cannot be tracked solely based on the time they resubmit an updated cohort transaction object. Such a tracking process could also be further hampered by potential network issues such as latency, poor node interconnection, etc. A more detailed description of the process can be found in 3.1.

4.2 Implementation

For our testing, we used an Intel i7-9700k CPU at 3.60GHz, 32GB of RAM, and an NVIDEA GeForce RTX 2080 graphics board for mining purposes. A simulation involving multiple clients was executed on a single computer, with all the clients connected to each other locally. To reduce the number of independent variables affecting our analysis, we assign the same k-value (k=*2000*) to all users in our test network, adopting the number from the data published by Google in its trials [20]. In the context of our model, we define satisfying the k-anonymity threshold as simply having at least k number of active users within the same cohort; this idea of "hiding in a crowd" theoretically prevents them from being individually tracked and identified. Thus, in a worst-case scenario, the information that could be harvested about a specific user is reduced to a group of k individuals. To mitigate the possibility of Sybil attacks [14], where a malicious node forges multiple identities in a peer-to-peer network, in BB-FLoC, we require users to solve a (simple) proof-of-work puzzle when submitting their cohort hash value. Likewise, blocks that contain duplicate transaction objects are dropped. Whilst this does not solve the issue entirely, such measures still have a positive impact on the security of the system. Furthermore, to prevent users from being tracked across the network, they are not required to sign their cohort messages or identify themselves with a public key. Only the block miners are identifiable, as this is necessary to assign them compensation for their computational and storage resources.

4.3 Computational Complexity

It is crucial that the computation of cohorts is a simple process that can be effortlessly done on lightweight devices such as mobile phones which may have limited computational resources and data storage. Cho et al. [8] contended that to effectively apply blockchain to an IoT network, three points must be met - it must be lightweight, secure, and applicable. In other words, it must be usable by devices with sparse computational resources whilst not compromising on security guarantees, and any such system must also be appropriate and not cause issues to the core objective of the operation at hand.

Our model enables the user to fully calculate their cohort group independently, without any communication to central nodes and uses highly secure PoW protocols both on the block itself and the individual transaction object. As the objective of BB-FLoC is to display targeted ads to users browsing the internet, we can reasonably take some liberations regarding the operating devices at hand being especially lightweight - nowadays, personal computers, and even smartphones/mobile devices are powerful enough to undertake robust security measures with little constraint.

The process of obtaining a cohort group ID begins with users individually calculating a cohort hash that requires reading data from a single JSON file containing the domain names of websites visited over a 24h period. This is followed by combing through a chain of blocks to gather all the currently valid cohorts in the network. Both such operations are at $O(n)$ complexity, whilst sorting through them in a lexicographic form until the k-anonymity threshold is no longer satisfied is $O(\log n)$. Sending out a legitimate transaction object (which includes the cohort hash of a user) to the network requires solving a PoW puzzle, which in our model was set to 2-leading numbers (e.g., transaction object hash of *'00xxx..'*), and took an average of just 1.2ms to finish, whilst the difficulty of block mining was set to allow a new block to be made every 24h.

4.4 Storage Requirements

One of the most prominent issues with any blockchain system is its data storage problem. The data contained in a blockchain is immutable, and every node in the network typically stores a full copy of the blockchain. In our model,

Table 3. The storage requirements of 100 blocks with varying numbers of TX objects per block

no. of blocks	no. of TX per block	Total size
100	0	132 KB
100	1	159 KB
100	100	1.03 MB
100	100000	904 MB

such data is users' cohort hashes, and due to the open nature of its objective, the block size has to be unbounded to enable free access to any user wishing to participate in the BB-FLoC process. Additionally, since everyone's cohort hash has to be re-evaluated every 24h, the storage of all such information proves to be a nontrivial task.

The data stored in BB-FLoC is a chain of block objects, each of which includes metadata information and a varying set of transaction objects which contain within it a cohort hash ID. Data from our system was stored via LevelDB [12], which is a fast key-value store library, developed by Google, and is used by major blockchain services such as "Bitcoin Core" and "Go-Ethereum". Additionally, each user stores their own browsing history in a JSON file, however this information only contains the top-level domain of the site, and this information is refreshed every 24h. A file containing the users' own calculated cohort hash group is also stored and is sent over to websites in place of cookies. Regardless, the storage cost of these 2 JSON files can be considered negligible. Analysis from our tests found that, on average, a single transaction object (refer to Fig. 3 for its structure) weighs as little as *0.19 KB*, whilst an empty block by itself was only *1.32KB*. Although the storage requirements are tolerable under a small-scale setting, confronted with a large number of users it scales to substantial levels. This is also seen in Table 3, where 100,000 transactions per block over a relatively short period of ≈14 weeks rises to nearly *1 GB* in storage.

Considering the storage costs of a blocks' metadata (1.32KB), the block interval parameter has minimal impact on costs - the real impingement comes from the number of users participating in the system and the cohort update interval. With a network of 100,000 total active users and a cohort refresh time of 24h, in a 30-day period the storage requirements amounted to 267MB, with 4167 transactions per each block; transfixed over a period of a year, we can project this value to be *3.25GB*. 100,000 active users with a cohort update time of 1 hour (i.e. all cohorts refresh each block interval cycle which is also 1 hour), the storage costs were 6.6GB for 30 days, and 80.5GB for a full year. Conceivably, such amounts may not be reasonable for storage-constrained devices such as mobile phones. However, it's crucial to emphasise that individuals with constrained computational resources are not necessarily excluded. They have the option to automatically request generalized advertisements, leverage cloud computing services for storage and computational tasks, or, participate in the network as just 'lightweight' nodes.

4.5 Lightweight Mode

Evidently, our system, much like many other blockchain systems, faces challenges when operating on devices with limitations such as mobile phones. As a viable resolution to address this concern, our system is capable of functioning without requiring users to store the entire blockchain history on their devices. Such users would still be able to independently calculate their cohort IDs, however, they would lose the ability to mine blocks and participate in the parameter voting process. For clarity, the BB-FLoC process described in Section floc-process changes in the aspect that lightweight nodes cannot operate as miners; they still validate blocks but only end up storing the most recent ones that still contain valid cohort hashes.

The concept here is simple: cohort hashes that have expired (e.g. >24h) will be permanently deleted from user storage, and only the currently valid hashes are kept in storage. The bandwidth requirements for lightweight nodes would remain the same as they would still need to download the full blocks when first connecting to the network. But, once a block is validated user-side, it can get indexed and its contents can be erased. The security guarantees remain the same since users still independently verify all the data in the blockchain, and as a trade-off for losing authority in the network they can save a lot of storage by deleting the expired cohort hashes. To put this into context, with 100,000 active users, a year-long blockchain would normally amount to *3.25GB* in storage for full nodes; for lightweight nodes, their requirements would only be around *9.1 MB*. On a larger scale of hundreds of millions of active users, in order to even further lower the storage demands, the blockchain could be split into multiple smaller blockchains with only a set number of users participating in each one.

4.6 Evaluating Utility

To examine the utility of our system we will use the function described previously in section 3.5. Whilst Google was coy in regards to its trial details, [3] claims that 0.5% of Chrome users were affected; combining this with Chrome statistics in 2022 [11], we can reasonably assume that as many as 13,250,000 users were in the trial. According to its trial publishing [20], the number of Locality-Sensitive Hashing (LSH) bits used to define a cohort was between 13 and 20, and its k-value was set at 2,000 users. Taking the low end of the bit value (*13*), we find that the advertiser utility was *0.549*.

To evaluate our BB-FLoC system, we will use the MovieLens dataset [21], which was also used in FLoC's initial whitepaper to assess cohort utility. This dataset contains a total of 27 million entries of movies watched and rated by a combined 276,224 users between January 1995 and September 2018. This acts as a good substitute for browsing history data as the type of movies watched should reveal user preference for a particular category/genre, similar to how users' web browsing habits can reflect their possible interests. After randomly selecting 5,000 cohort hashes from our total user list, and a set k-value of 2,000, we found that the median number of bits in a k-anonymised cohort group ID was 9, with its size ranging between 8 and 10 bits. This corresponds to a utility of *0.498*. Whilst Google's trial was of a significantly larger scale, we find that our results closely match; moreover, in a general sense, the utility is higher as the number of users in the system is lowered, assuming that both population sizes have the same skewness in the popularity distribution of websites. This is because under the scenario that the full 50-bits are used as a cohort group ID, there is a higher chance for that very same cohort to be more populated with a larger number of users participating. Therefore, if the number of total users increases to excessive levels, system operatives could decide to use sharding [23] as a potential scaling approach.

The mutability of blockchain enables users to individually pick their k-value, so although our initial utility calculation was done to be comparative to Google's trial, the k-value of 2000 is merely a suggestion. To further evaluate the utility function, we examined the average utility based on a users' set k-value. We reiterate that the utility function is based on the advertisers' perspective, that is, a higher utility value means that users' cohort group is more specialised, and more specifically tailored ads can be displayed to them. We show our results in figure 4, where using the MovieLens dataset, we picked out a randomised subset of user cohorts, and calculated the size of their cohort ID based on k-values from 1 to 10,000 and listed the minimum, average, and maximum bit lengths of cohorts based on specific k-values. For the utility function, we used the average cohort bit size and 276,224 as the number of total users. With k-value of 1, where the user can be clearly identified, the average cohort group size was 20 bits, and gave a utility of 1.11; on the other end of the scale, with a k-value of 10,000, the average cohort size was 6.3 bits, and had a utility of 0.35. However,

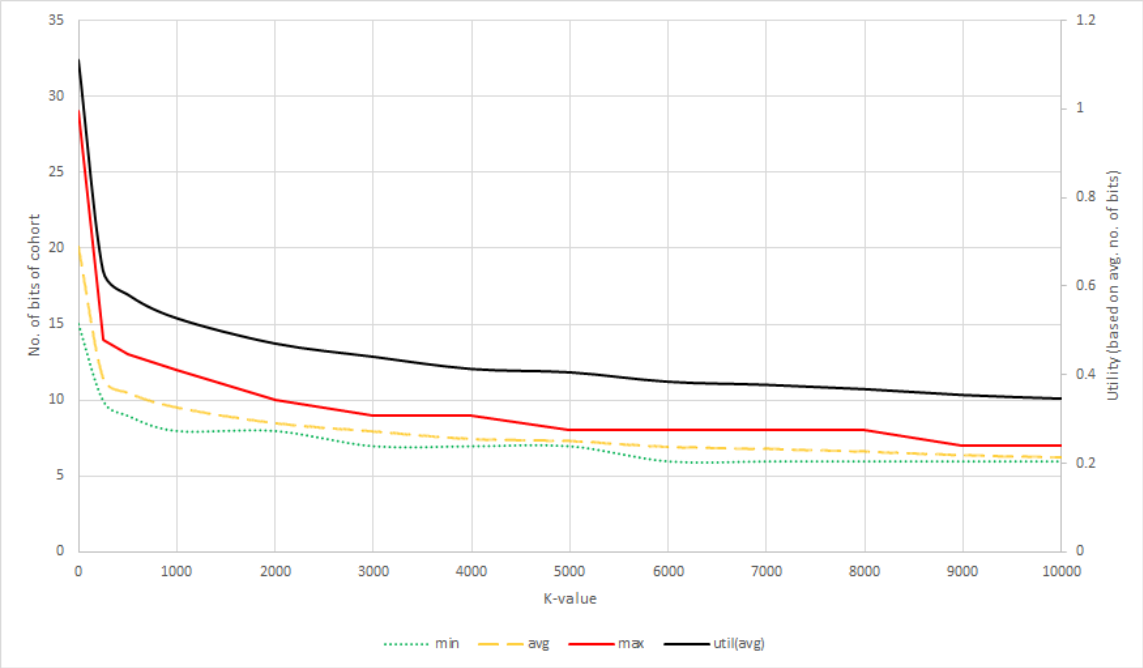

Fig. 4. The utility based on the average number of bits found for a particular k-value.

Table 4. Overview of potential privacy attacks in BB-FLoC

Attack Name	Definition
Sybil [30], [5]	Spam the network with fake cohort hashes, inflating cohort group size
Homogeneity [5], [16]	Users within a cohort ID may also share a sensitive attribute between them, which can be used to target them
Inference [33], [30]	Exploit the lack of pre-image resistance in LSH to infer the websites visited by users
Fingerprinting [29], [5]	Collecting and storing additional information (IP addresses, device-specific data) in addition to cohort IDs can allow users to be individually tracked

as can be seen, once the k-value increases to a certain extent, the utility remains relatively uniform, so users can choose higher privacy guarantees without severely impacting advertiser utility.

5 POTENTIAL ATTACKS

Although FLoC protects users from having to share their private browsing histories, they can still be subject to a variety of attacks. These may include, but are not limited to, the examples provided in Table 4. Note that just as in the original FLoC implementations, a blockchain-based FLoC model is unauditable. Each individual user in our system calculates their cohort ID independently, therefore they are free to respond with fake IDs that do not represent their true browsing history (e.g. by locally modifying browsing data on their computers), and the advertisers/site owners have no way of

detecting such behaviour. However, provided the cohorts are adequately large in size, user privacy is not going to be breached, so they have nothing to gain by submitting false IDs.

A study conducted in [5] aimed to assess the feasibility of a homogeneity attack. Whilst the precise definition of sensitive information can be open to interpretation, the European Commission has outlined a set of criteria in a published list [9] that businesses should adhere to for specific processing conditions. Among them was data revealing racial or ethnic origin, which was used by Berke et al. [5] to evaluate FLoC privacy, and their findings indicated no discernible correlation between users' race and their browsing history in FLoCs, therefore, we can reasonably dismiss homogeneity attacks. In terms of the inference attack, our blockchain-based FLoC design gives the power back to the users to individually pick both the *k-anonymity value* and the number of bits from their *(50-bit)* hash that they share with the network. Whilst neither of these approaches fully prevents the possibility of a successful attack, it should make it more difficult for malicious parties to do so. The smaller the number of bits of the cohort hash that users commit publicly, the more likely it is that cohort group sizes are going to be bigger, hiding an individual in a larger crowd; indeed, it would be reasonable to assume that the k-value that users would choose would be higher than whatever value Google or some other centralised party whose main intention is to satisfy advertisement agencies would pick. Additionally, with our approach, users never have to share the full 50-bit hash of their browsing history, further increasing their privacy. To safeguard against Sybil attacks, one may request users to complete a proof-of-work puzzle when sending out their transaction object to the network, in a similar practice to how Hashcash [4] was used to prevent email spam. Without loss of generality, security risks related to k-anonymity, blockchains, and LSHs have been unattended by our model; our focus here is chiefly on the feasibility of decentralising the processing of information about a crowd without compromising individual identities.

6 CONCLUSIONS

In this paper, we designed a blockchain-based FLoC system that enabled us to achieve k-anonymity guarantees in a fully decentralised process, whilst at the same time allowing for ad agencies to conduct targeted advertising. We further conducted testing of our model, providing information about storage requirements, and utility guarantees, showing that the model is viable even for low-end devices such as mobile phones. We must note that the application of BB-FLoC can be extended beyond just as an alternative to third-party cookies; any problem requiring the grouping of users in an anonymous, yet useful and applicable way can be potentially fulfilled with BB-FLoC.

Whilst our analysis has shown that blockchain is a useful addition to the FLoC design, our analysis was done on a low scale of just a maximum of 276,224 users. In the future, this number could be increased, perhaps matching Google's assumed test pool of over 13 million users to further certify the viability of our blockchain-based approach. It would also be crucial to further evaluate the viability of the various attacks on FLoC, such as Sybil and inference attacks. Additionally, we believe there are insights to be gained from testing different kinds of differential anonymity methods such as ε-anonymity.

Overall, although the FLoC project was abandoned by Google, we consider it to be a system that still has the potential to be viable as an alternative to third-party cookies. There is much research to be conducted, especially in regards to its effectiveness in providing privacy guarantees to users when combined with other tracking techniques such as fingerprinting [29], however, we believe our work is another step towards a fully practical system that will replace cookies in the digital world.

ACKNOWLEDGMENTS

This work was supported in part by the Ph.D. Scholarship provided by the School of Computing and Communications (SCC) Department, Lancaster University.

REFERENCES

[1] Lawrence Abrams. 2021. *Microsoft disables Google's FLoC tracking in Microsoft Edge, for now.* Retrieved March 28, 2023 from https://www.bleepingcomputer.com/news/microsoft/microsoft-disables-googles-floc-tracking-in-microsoft-edge-for-now/

[2] Google Research & Ads. 2020. *Evaluation of Cohort Algorithms for the FLoC API.* Retrieved March 28, 2023 from https://github.com/google/ads-privacy/blob/master/proposals/FLoC/FLOC-Whitepaper-Google.pdf

[3] AmIFloced. 2021. *Am I FLoCed?* Retrieved March 18, 2023 from https://amifloced.org/

[4] Adam Back. 2002. Hashcash - A Denial of Service Counter-Measure. (09 2002).

[5] Alex Berke and Dan Calacci. 2022. Privacy Limitations Of Interest-based Advertising On The Web: A Post-mortem Empirical Analysis Of Google's FLoC. *arXiv preprint arXiv:2201.13402* (2022).

[6] Thomas Bocek. 2023. *TomP2P.* Retrieved January 22, 2024 from https://github.com/tomp2p/TomP2P

[7] Matt Burgess. 2021. *Google's plan to eradicate cookies is crumbling.* Retrieved March 17, 2023 from https://www.wired.co.uk/article/google-floc-trial

[8] Sunghyun Cho and Sejong Lee. 2019. Survey on the Application of BlockChain to IoT. In *2019 International Conference on Electronics, Information, and Communication (ICEIC).* 1–2. https://doi.org/10.23919/ELINFOCOM.2019.8706369

[9] European Commission. 2024. *What personal data is considered sensitive?* Retrieved January 15, 2024 from https://commission.europa.eu/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/sensitive-data/what-personal-data-considered-sensitive_en

[10] Bennett Cyphers. 2021. *Google's FLoC Is a Terrible Idea.* Retrieved March 29, 2023 from https://www.eff.org/deeplinks/2021/03/googles-floc-terrible-idea

[11] Brian Dean. 2021. *Google Chrome Statistics for 2022.* Retrieved March 11, 2023 from https://backlinko.com/chrome-users

[12] Ghemawat & Dean. 2021. *LevelDB is a fast key-value storage library written at Google that provides an ordered mapping from string keys to string values.* Retrieved March 14, 2023 from https://github.com/google/leveldb

[13] Ali Dorri, Salil S Kanhere, Raja Jurdak, and Praveen Gauravaram. 2017. Blockchain for IoT security and privacy: The case study of a smart home. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops).* IEEE, 618–623.

[14] John R. Douceur. 2002. The Sybil Attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems (IPTPS '01).* Springer-Verlag, Berlin, Heidelberg, 251–260.

[15] Alessandro Epasto, Andrés Muñoz Medina, Steven Avery, Yijian Bai, Robert Busa-Fekete, CJ Carey, Ya Gao, David Guthrie, Subham Ghosh, James Ioannidis, Junyi Jiao, Jakub Lacki, Jason Lee, Arne Mauser, Brian Milch, Vahab Mirrokni, Deepak Ravichandran, Wei Shi, Max Spero, Yunting Sun, Umar Syed, Sergei Vassilvitskii, and Shuo Wang. 2021. Clustering for Private Interest-Based Advertising *(KDD '21).* Association for Computing Machinery, New York, NY, USA, 9 pages. https://doi.org/10.1145/3447548.3467180

[16] Medina et al. 2021. *Measuring Sensitivity of Cohorts Generated by the FLoC API.* Retrieved March 17, 2023 from https://docs.google.com/viewer?a=v&pid=sites&srcid=Y2hyb21pdW0ub3JnfGRldnxneDo1Mzg4MjYzOWI2MzU2NDgw

[17] Medina et al. 2022. *Get to know the new Topics API for Privacy Sandbox.* Retrieved March 27, 2023 from https://blog.google/products/chrome/get-know-new-topics-api-privacy-sandbox/

[18] Ohtsu & Foote. 2021. *FLoC Simulator.* Retrieved March 26, 2023 from https://github.com/shigeki/floc_simulator

[19] David Giessen. 2019. *Blockchain and the GDPR's right to erasure.* B.S. thesis. University of Twente.

[20] Google. 2021. *FLoC Origin Trial & Clustering.* Retrieved March 27, 2023 from https://sites.google.com/a/chromium.org/dev/Home/chromium-privacy/privacy-sandbox/floc

[21] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (dec 2015), 19 pages. https://doi.org/10.1145/2827872

[22] Garrett Johnson, Julian Runge, and Eric Seufert. 2022. Privacy-Centric Digital Advertising: Implications for Research. *Customer Needs and Solutions* 9 (06 2022), 1–6. https://doi.org/10.1007/s40547-022-00125-4

[23] Michał Król, Onur Ascigil, Sergi Rene, Alberto Sonnino, Mustafa Al-Bassam, and Etienne Rivière. 2021. Shard scheduler: object placement and migration in sharded account-based blockchains. In *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies.* 43–56.

[24] Edvinas Kruminis. 2023. *A blockchain-based implementation of FLoC (Google).* Retrieved March 11, 2023 from https://github.com/ekruminis/floc-blockchain

[25] Natasha Lomas. 2021. *Google isn't testing FLoCs in Europe yet.* Retrieved March 22, 2023 from https://techcrunch.com/2021/03/24/google-isnt-testing-flocs-in-europe-yet/

[26] Satoshi Nakamoto. 2009. Bitcoin: A Peer-to-Peer Electronic Cash System. *Cryptography Mailing list at https://metzdowd.com* (03 2009).

[27] Rescorla. 2021. *Privacy analysis of FLoC.* Retrieved March 10, 2023 from https://blog.mozilla.org/en/privacy-security/privacy-analysis-of-floc/

[28] s Sarmah. 2018. Understanding Blockchain Technology. 8 (08 2018), 23–29. https://doi.org/10.5923/j.computer.20180802.02

[29] Rescorla & Thomson. 2021. *Technical Comments on FLoC Privacy.* Retrieved March 29, 2023 from https://mozilla.github.io/ppa-docs/floc_report.pdf

[30] Florian Turati, Carlos Cotrini, Karel Kubicek, and David Basin. 2023. Locality-Sensitive Hashing Does Not Guarantee Privacy! Attacks on Google's FLoC and the MinHash Hierarchy System. arXiv:2302.13635 [cs.CR]

[31] Xiao Yue, Huiju Wang, Dawei Jin, Mingqiang Li, and Wei Jiang. 2016. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. *Journal of medical systems* 40, 10 (2016), 1–8.

[32] Shiwen Zhang, Tingting Yao, Voundi Koe Arthur Sandor, Tien-Hsiung Weng, Wei Liang, and Jinshu Su. 2021. A novel blockchain-based privacy-preserving framework for online social networks. *Connection Science* 33, 3 (2021), 555–575.

[33] Ping Zhao, Hongbo Jiang, Chen Wang, Haojun Huang, Gaoyang Liu, and Yang Yang. 2019. On the Performance of kk -Anonymity Against Inference Attacks With Background Information. *IEEE Internet of Things Journal* 6, 1 (2019), 808–819. https://doi.org/10.1109/JIOT.2018.2858240