Accountability Platform

A specification for open, auditable data structures and standard practices to reliably demonstrate digital advertising supply chain conformity to preferences and restrictions set by users and the digital properties they visit.

March 2021

Draft in Request for Public Comment until May 7, 2021

This document has been developed by the Rearc Accountability Working Group, in cooperation with the Partnership for Responsible Addressable Media (PRAM).

With impending disruption to the identifier landscape, Project Rearc is a global call-to-action for stakeholders across the digital supply chain to re-think and re-architect digital marketing to support core industry use cases, while balancing consumer privacy and personalization. *The Rearc Accountability Working Group* is responsible for creating interoperable global accountability standards to give users assurance that their personal data is processed in the way they expect and to support addressability, informed by input from the global business and policy dialogue within the Partnership for Responsible Addressable Media.

Rearc Accountability Working Group Roster

At the date of publishing, Rearc Accountability Working Group Roster is made up of 267 individuals representing 117 organizations. Full roster details can be viewed here.

About IAB Tech Lab

Established in 2014, the IAB Technology Laboratory (Tech Lab) is a non-profit consortium that engages a member community globally to develop foundational technology and standards that enable growth and trust in the digital media ecosystem. Comprised of digital publishers, ad technology firms, agencies, marketers, and other member companies, IAB Tech Lab focuses on solutions for brand safety and ad fraud; identity, data, and consumer privacy; ad experiences and measurement; and programmatic effectiveness. Its work includes the OpenRTB real-time bidding protocol, ads.txt anti-fraud specification, Open Measurement SDK for viewability and verification, VAST video specification, and Datalabel.org service. Board members/companies are listed at https://iabtechlab.com/about-the-iab-tech-lab/tech-lab-leadership/. For more information, please visit https://iabtechlab.com.

IAB Tech Lab Contact

Alex Cone
Sr. Director, Product Management

Accountability Platform

Overview

The Accountability Platform is a set of standard practices intended to reliably demonstrate, via standard data structures and reporting, digital advertising supply chain conformity to data use preferences and restrictions set by users and the digital properties they visit. While standards alone cannot guarantee conformity, the aim of this design is to lay a foundation for continuous improvement in pursuit of a future where for digital advertising people trust that data is used, or not, in the ways they expect.

Guiding Principles

Before describing the Accountability Platform architecture, it is important to understand the principles guiding its design.

- It is possible to detect nonconformity to transparency and control signals and to demonstrate responsible and irresponsible use of user identifiers in digital advertising supply chains.
- Transparency to and control from users is dependent on the digital properties a user interacts with making clear decisions about who to work with and what set of standard data uses they are comfortable with.
- Any user control combined with the initial choices of the digital properties users interact with is the best representation of uses allowed for any user identifier in a given ad-related transaction.
- Consistent and continuous demonstrations of adherence to what is allowed and detection of violations for a given ad-related transaction across the full supply chain and each of its participants can create a virtuous cycle of incentives for responsible data collection and use.
- Open approaches are vastly preferred to closed approaches and making Accountability Platform data available to many parties with diverse motivations (e.g., researchers, academics, regulators, auditors) strengthens the design.
- This standard should not create any new opportunities for breaches of user privacy or data protection.

Scope

Participant Profile

This proposal tackles the means of providing consistent technical audit opportunities for digital ad industry self-regulatory regimes, regional authorities and other participants within the supply chain. The Accountability Platform describes the process of generating sample logs from participants that indicate whether addressable user identifiers are shared between participants. In the spirit of transparency and balancing the commercial realities of the supply chain, the proposal aims to not disrupt participants' ongoing logging activity in a significant way.

This document uses the term “participant” and “participants” as short-hand to indicate what the specification requires. A potential ecosystem participant is any company or entity represented by a domain that passes an addressable user identifier to another domain. The passing of an identifier could be for a number of uses. The type of participants can generally be labeled as:

- Brands (presumed to be supported here by consent management platforms, or “CMPs”)
- Publishers (presumed to be supported here by CMPs)
- Supply side technologies (ad servers, SSPs, ad network products, verification products)
- Buy side technologies (ad servers, DSPs, ad network products, verification products)
- Identity resolution technologies
- Other third party data providers and processors

If would-be participants are precluded from participating in the Accountability Platform based on regulation or contractual agreements, participation should be excluded. Some laws like the GDPR define data “processors” as separate from data “controllers” and therefore there are different rights. In a regulatory environment like the GDPR a “processor” may need permission from data “controller(s)” to process and participate in the logging and submission activities laid out in this RFC. When a participant fails to deliver on the sample logs described later in this document this may be mitigated by the other participants which reference them in the supply chain as long as a minimal threshold of participation is met. Individual firms who only have inferential signals in the ecosystem will likely be targeted for further scrutiny by regulators and consumers. If certain supply chains don’t participate en-masse vs. other supply chains, one can expect the demand side to shift budgets to compliant supply chains to mitigate brand safety challenges.

There are cases where entities do not log addressable user identifiers when there is no consent or to comply with similar policies and contracts. The Accountability Platform doesn’t suggest that participants should start logging and storing addressable user identifiers for all such cases. The Accountability Platform expects that participants only log the last 5 characters of hashed (logic as described further in the “Proposed Architecture” section) user identifiers so that they match the query criteria described in the proposed architecture.

Use Cases

This proposed specification is concerned with ad-related transactions where an addressable user identifier is passed between a sender and receiver. The following list of addressable user identifiers is covered:

- User-enabled addressable user identifiers like what is currently described by The Trade Desk’s UID2

- "1st party" addressable user identifiers passed between a digital property and their supporting monetization or campaign tech. These identifiers may not travel end to end through a regular ad supply chain, but at least pass from the digital property to its supporting technology vendors (ex., publisher123.com to publisher123's ad server)
- Device provided addressable user identifiers (i.e., GAID, IDFA, TIFA in Connected TV)
- Other system or ad technology vendor generated identifiers intended for granular addressability

This proposed specification also only works where privacy signals exist for the transaction and are transmittable through the supply chain. While we realize this limits the initial scope of this RFC, IAB Tech Lab members are also seeking to address this full supply chain compatible privacy signal with the Global Privacy Platform proposal currently in public request for comment. **If there is a privacy signal which can be communicated through the supply chain then the transaction is in-scope.**

Proposed Architecture

Logging and Retention

The foundation of the Accountability Platform is recording and storing ad related transaction data in a standard structure so that it can be analyzed for conformity with data use expectations encoded in privacy signaling. A minimum set of properties for each ad related transaction for each participant must be made transparent, recorded and stored for later uniform submission for various compliance analysis.

> ***Note:*** *this specification does not preclude real-time mechanisms, but those are not the focus of this RFC. See Further Decentralizing these Designs and Making them Real Time for recently developed ideas to extend this proposed architecture.*

Sender/Receiver Pairs

Sender/Receiver Pairs provide a foundation for ad supply chain relationship analysis. Each Accountability Platform participant by nature connects with at least one other participant (i.e., web page to ad server, ad server to SSP, etc.). Participants can be a "**Sender**," "**Receiver**" and are commonly both. A "Sender" is the participant calling a Receiver where an addressable user identifier is part of the call. The "Receiver" is the participant to which a Sender calls with an addressable user identifier. This is true whether the transfer is regarding a single addressable user identifier or in batch. A Sender may call multiple Receivers. Subsequently, a Receiver itself becomes a Sender by calling additional Receivers, again with an addressable user identifier. Pair participants are obligated and even incentivized to report accurately as they can expect each other to do so.

This version of the specification helps compliance enforcers determine:

- *who is sharing data with whom*
- *whether both Sender and Receiver are reporting the same transactions*
- *whether both Sender and Receiver are recording the same privacy signals*
- *whether both Sender and Receiver respect signals which do not allow any use of an addressable user identifier (i.e., lack of ePrivacy opt in for Europe, an opt out like that described in UID2 designs, etc.)*

Senders create a **Transaction ID** used to keep track of the same transaction across a pair in a reporting interval. Transaction IDs are recorded and sent to each Receiver when any persistent ID, used for maintenance of state enabling allowed data uses, is present.

Receivers log the Transaction ID from the Sender along with the other transactional metadata required by this specification (see below). If a Receiver connects to additional Receivers, the Receiver itself then becomes a Sender as well, creating an additional Transaction ID for each of its Receivers. This flow repeats per Sender/Receiver pair.

Transaction IDs are not designed or useful for maintaining state on a user or her device. They are arbitrary per transaction and Sender/Receiver pair and only need to remain unique for the pair given reporting duration. When Sender/Receiver Pairs are subsequently joined on Transaction IDs (see Log Data Submission), the combined records should show consistency between the two sides of the pair.

Identifying Senders and Receivers

Senders and Receivers must be tied to a legal entity. A key focus of this specification is enabling regional privacy and data protection compliance enforcement throughout the digital advertising supply chain. This specification proposes Sender and Receiver IDs piggy-back on the rather robust DNS system. Most entities already have a unique identifier, their domain (eTLD+1), and if they do not it is very simple to acquire one for participation in this standard. Thus, this eTLD+1 provides the seed into creating a verifiable organizational identifier, while relying on a decentralized architecture for more rapid adoption.

***Note:** the initial mapping of domain name to legal entity would be the existing DNS system. However, this specification realizes the existence of company registration lists which themselves will need to join on the eTLD+1 to perform compliance analysis. An example of this is IAB Europe's TCF v2.0 which Accountability Platform wishes to support.*

Verification of Sender and Receiver IDs

In addition to entity identification, it is valuable to verify that certain information originated from a specific entity. When a Sender and Receiver communicate over standard digital communication methods, Transport Layer Security (TLS) provides such verification between those two entities. This type of verification uses a cryptographic signature, which allows anyone with the document,

signature, and the authenticator's public key to verify that the signer must have the private key. For the use cases described in this specification, including such a signature in each log allows anyone to verifiably prove the sender is who they claim.

The only additional requirement for identification, then, would be a certificate authority (CA), which keeps track of the mapping of the legal data possessor associated with this domain and the public key. Any entity with multiple participants' logs can then leverage the domain name's existing certificates, which should already be registered with a public CA. A new certificate could be issued specifically for this functionality and hosted at a well-known path on the eTLD+1, such as `/.well-known/accountability.pub`. This public key can be certified with a chain of trust tied to the domain name's primary certificate.

> ***Note:*** *while using private/public key encryption ensures the verifiability of the identity of organization that signed the transaction, to protect against spoofing or replying a valid ID, it would make sense for the signing of the payload that contains at least a timestamp if not also the originating transaction ID in the chain. Thus, a bad actor attempting to steal a digital property's identity (or any other intermediary for that matter) would be easily found out, given the timestamp of the "replayed" sender ID would not match the current timestamp of a transaction. By also including the originating transaction ID in the signed payload, this would also help detect fake or altered transactions even when the organizational identifier itself is not in question.*

Transmitting a Transaction ID

The Accountability Working Group is weighing several options for how Sender/Receiver pairs will also transmit a Transaction ID. We seek public input on these options summarized below as well as ideas for other mechanisms that wouldn't require significant infrastructure changes.

- Concatenation to the addressable user identifier shared
  - Pros:
    - No need for a new transport mechanism
  - Cons:
    - No entity available to do concatenation with the addressable user identifier when that identifier is from the device (an unlikely participant to this specification, though desired)
    - Entities not participating in this standard might have trouble with the concatenated Transaction ID
    - When a privacy signal does not allow passing of the addressable user identifier what would the Transaction ID be concatenated to?
- Extension to OpenRTB
  - Pros:
    - This is a common way of supporting new values passed through the digital advertising supply chain
  - Cons:

        - Not every potential participant has an OpenRTB integration
- URL Macro
    - Pros:
        - Addresses server side and client side integrations
    - Cons:
        - Length issues
- Concatenation to privacy signal
    - Pros:
        - No need for a new transport mechanism
    - Cons:
        - Entities not participating in this standard might have trouble with the concatenated Transaction ID
        - Length issues

Data to Log for Submission

For all covered ad related transactions specification participants will log the following fields which will be submitted per the process in this specification:

Field Name	Description	Type	Values
`version`	The version of this specification the record represents.	VARCHAR	Insert
`timestamp`	The time a transaction is logged	TIMESTAMP	Insert
`senderId`	Non-repudiable, eTLD+1 domain signed over TLS of the entity sending (or allowing the retrieval of) a persistent identifier usable for data processing purposes laid out by whatever privacy and data protection regime(s) the digital property is operating in.	BIGINT	A 64 byte array containing the signature described above
`receiverId`	Non-repudiable, eTLD+1 domain signed over TLS of the entity receiving (or carrying out the retrieval of) a persistent identifier usable for data processing purposes laid out by whatever privacy and data protection regime(s) the digital property is operating in.	BIGINT	A 64 byte array containing the signature described above
`transactionRole`	Whether the entity logging this transaction is logging as the Sender or Receiver. *Note: a*	BINARY	`0` for “Sender” `1` for “Receiver”

	Receiver can also be a Sender for a subsequent Sender/Receiver pair in the same supply chain transaction.		
`transactionId`	A single use transaction identifier unique to the Sender/Receiver pair for a given reporting interval (described below) as far as the Sender, who generates the `transactionId` knows.	BIGINT	Insert
`privacySignal`	The Tech Lab supported privacy signal (TC string, USPrivacy string or Global Privacy Platform string) where available and if not any known opt-out signal related to the transaction.	BIGINT	Insert

Senders and Receivers will also log the addressable user identifier in use for each in-scope transaction. However, **user data will not be submitted to any clearing house or auditing entities.** Instead, the persistent user ID is only used by the participant for later retrieval of a sample for the reporting interval (described below). To make those sample queries easier to run and allow for consistent overlap in transactions for an interval between Sender/Receiver pairs, the persistent user ID for that transaction should be hashed using MD5 to return a consistent value between a given Sender/Receiver pair. The participant should follow all relevant data protection, security and control procedures for storing any logged user data. Methods for that sensitive storage or any other storage are not prescribed by this specification.

> ***Note:*** *there are cases currently where entities do not log user identifiers when there is no consent or to comply with similar policies. The Accountability Platform doesn't suggest that participants should start logging and storing identifiers for all such cases. The Accountability Platform expects that participants only log last 5 characters of hashed user identifiers so that they match the sample query criteria as described in the Log Data Submission section below.*

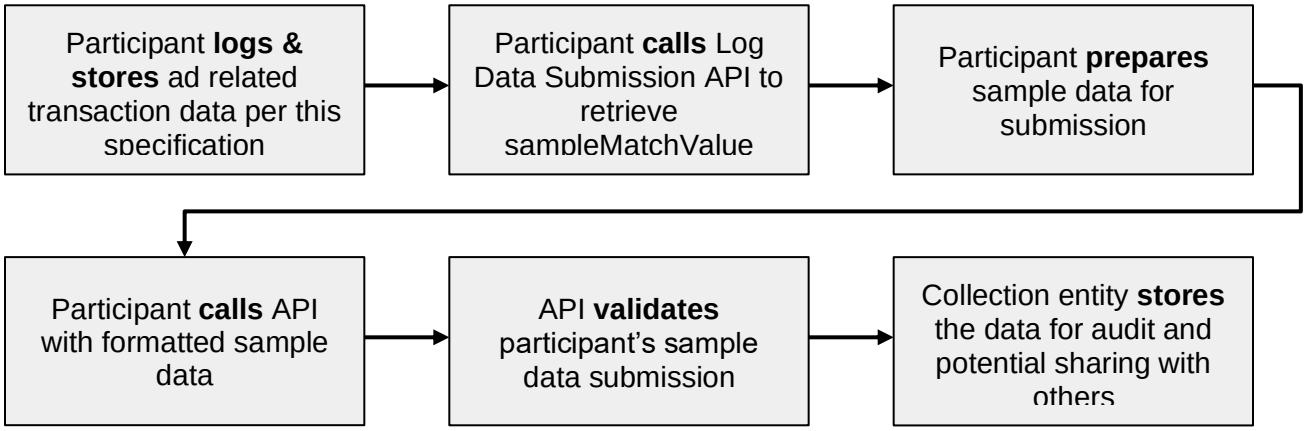
Expectations for Retention

Ultimately, the data submitted (see "Log Data Submission" below) are a sample of the transaction data logged and recorded for the prescribed period of 1 day. Sampling does not occur at the time of transaction, but at time of data submission. Participants are expected to log and record the above data structure for *all* transactions in the prescribed period. The method in which the records are stored is not dictated by this specification.

Log Data Submission

Participants submit a daily sample of transaction logs described above. The Log Data Submission API accepts sampled log data from participants to be validated and made available to entities wishing to analyze the data. This specification and guidelines describe how participants will interact with the Log Data Submission API.

Workflow



Log Data Preparation

Participants log and store ad related transaction data following the methods laid out previously in this specification. Participants will submit their sample logs daily.

> *Daily submission is necessary to manage file sizes and to reduce the consequence of a single log submission which results in failure.*

Call `/job`

At the regular appointed time defined on participant registration, a participant calls the Log Data Submission API `/job` endpoint with its `senderId` (or `receiverId` if receiver only) to retrieve the job definition including a job ID, a sample match value and the latest operational version value.

> *The Log Data Submission API is a secure, non-authenticated endpoint hosted by a central entity which provides sample keys to participants, validates submissions and stores them to be available to entities seeking to analyze transaction data.*

```
GET /job

{
  "senderId": 123
```

```
}
```

Response:

```
{
  "jobId": 999111,
  "maxRecords": 1000000000,
  "sampleMatchValue": "*001",
  "alternateMatchValues": "[*1,*01,*1001,*12001"...],
  "operationalVersion": "1.0"

}
```

Attribute	Type	Description
jobId	INTEGER	Included with the metadata for each submission to facilitate the management of data by the collecting service
maxRecords	NUMBER	Maximum records that a participant is expected to submit if the sample match value yields records that cross this value, select more granular match value from alternateMatchValues
sampleMatchValue	STRING	Randomly generated value to define what subset of identifiers is to be included in the submission The format is simple regex based criteria with wildcards (*,?)
alternateMatchValues	ARRAY STRING	Provides additional match values when sample size of records cross max number for participant or go below min threshold. In that case participants can pick these more coarse/granular match values. The format proposed

		is simple regex with wildcards(*,?) to better communicate the match criteria. The values will be in descending order of granularity.
`operationalVersion`	STRING	Modified if there are schedule changes, changes to SLAs, changes to end-points or any other updates to the relationship between participants and the collecting service

Query Logs

Participants use `sampleMatchValue` returned by calling `/job` to query their logs for all records where the last digits of a MD5 hashed user ID in a covered ad-related transaction match the sample match value.

Example of finding transaction with sampleMatchValue:
`95c32139aa84e75d6c1da773aa88a8f6dad44802d862dd241a7155f7e10`**`159e9`**

After generating the sample data, participants then write the results to Avro with a header with metadata that includes the `jobId`, `senderId`, `createdOn`, and `numberRecords`.

Submission Workflow

Participants will follow this workflow and format when submitting daily logs.

Call `/submit`

Once sample submission data is correctly formatted participants call the Log Data Submission API with the formatted data.

```
POST /submit

{
  "jobId": 999111,
  "senderId": "123",
  "fileLocation": "participanturl.com/path/to/file.type",
  "createdOn": "2021-01-01T07:27:00.000Z",
```

```
  "numberRecords": 100000,
  "checksum": 0
}
```

Attribute	Type	Description
jobId	LONG	Value provided from calling /job endpoint prior to data preparation. Used to facilitate the management of data by the collecting service.
senderId	INTEGER	Non-repudiable, eTLD+1 domain signed over TLS of the entity sending (or allowing the retrieval of) a persistent identifier usable for data processing purposes laid out by whatever privacy and data protection regime(s) the digital property is operating in.
fileLocation	STRING	URL where file containing transaction data structure described by this standard is made available to be fetched by collection entity.
createdOn	DATE	Time internal data preparation job and formatting completed.
numberRecords	INTEGER	The number of records in the data submission.

Log Data Status

Call ../status

```
GET /job/123/status
```

Incomplete processing response

```
{
  "jobId": 123,
  "senderId": "123",
  "fileLocation": "path/to/bucket/file.ext",
  "startedOn": "2021-01-01T07:27:00.000Z",
```

```
  "numberRecords": 10000,
  "numberRecordsProcessed": 555,
}
```

Completed response

```
{
  "jobId": 123,
  "senderId": "123",
  "fileLocation": "path/to/bucket/file.ext",
  "startedOn": "2021-01-01T07:27:00.000Z",
  "numberRecords": 10000,
  "numberRecordsProcessed": 10000,
}
```

Attribute	Type	Description
`jobId`	LONG	Value provided from calling `/job` endpoint prior to data preparation. Used to facilitate the management of data by the collecting service.
`senderId`	INTEGER	Non-repudiable, eTLD+1 domain signed over TLS of the entity sending (or allowing the retrieval of) a persistent identifier usable for data processing purposes laid out by whatever privacy and data protection regime(s) the digital property is operating in.
`fileLocation`	STRING	URL where file is made available to be fetched by central entity.
`startedOn`	DATE	Time internal data preparation job and formatting completed.
`numberRecords`	INTEGER	The number of records in the data submission.
`numberRecordsProcessed`	INTEGER	The number of records the API has processed at the time of calling

Success

Once a participant successfully submits a sample for the period, the participant retains all data for that period for an additional 4 days so that the central entity or others may subsequently request additional data for analysis. After that time, participants continue to follow their own policies for data retention.

Log Data Availability

The collecting service joins submitted logs across Sender/Receiver transaction pairs all participants on a compound key made up of Sender ID, Receiver ID and transaction ID. When participants successfully record each transaction and submit their logs, and those they transact with do as well, the joined set will contain a pair of records for each transaction and the privacy signals in both the records should be the same. Furthermore, those same signals should say whether or not the Sender should have passed an addressable user identifier or not. Over time and in addition to other audit methods information can be a valuable tool to help establish the trustworthiness of a participant. A participant who is demonstrably not conforming to privacy signals could be identified via normal corporate communications and asked to explain themselves based on hard evidence. Their ability to transact on addressable user identifiers could be called into question. On the other hand, those that seen consistently conforming to privacy signals would gain positive reputational benefits concerning their trustworthiness.

A few potential types of entities and analysis are:

- Self-regulatory privacy audit programs which augment their audits with this data
- Researchers or regulators interested in determining adherence to transparency and control mechanisms for digital advertising
- Self-regulatory transparency and control signaling standards with policies for accountability
- Commercial auditors engaged for participant partner relationship audits

Sample ad-related transaction data is available for download to entities requesting it.

Further Decentralizing these Designs and Making them Real Time

Decentralization is an important principle underlying the Open Web. Decentralization fosters increased competition and innovation, increasing the diversity of organizations and people can choose to interact with. In relation to web architecture, this proposal relies on decentralized generation of organizational identifiers, by relying on private/public key encryption to generate identifiers as opposed to a centralized licensing authority to issue organizational identifiers. The benefit of this design is that the organizational identifiers are not only easily verifiable, but do not add incremental cost as each organization sending and receiving digital data already initiates or receives data transfers with a specific internet domain. Moreover, by relying on private/public key encryption these organization identifiers become non-repudiable, rather than easily spoofed by copying or replaying an organizational identifier licensed by a central authority.

There are additional ways we can decentralize this current proposal's design and at the same time provide additional real-time hooks. These are presented here as additional opportunities for public reflection as this document is in a period of public comment.

To determine whether further decentralization improves the efficacy of the architecture proposed in this document, we should briefly restate its key design goals. The summarized goals of this proposed standard are to accurately and effectively improve the accountability of digital advertising, by improving the detectability of non-conformity with privacy and data protection signals. In meeting these goals, we should be cognizant of impacts on improving detectability, speed of detection and incremental costs. The larger the sample rate, the greater the chance bad actors will not get away with their violations. In contrast, the longer the delay in auditing, the longer the bad actor will continue to perpetrate its bad actions. Finally, the lower the operational costs, the lower the barriers to entry and greater the likelihood of more rapid adoption.

Extending Sender/Receiver Architecture

This proposal recommends the pairwise data transfers be logged by each participant for future auditing. One approach to further decentralize the design is to also require participants to include the chain of signed organizational identifiers they received, when they onward send to the next Receiver. The benefits of this approach would be increased transparency to people, publishers and marketers for the data transfers involved in interacting with digital properties. For example, if each organization inserted its identifier into the transaction, people would be able to see the complete chain from ad request to ad render involved with the delivery of content to the publisher page. The resulting chain could be recorded into custom-key fields of digital properties existing ad servers—at the ends of most every ad-related transaction.

Potential Benefits of Further Decentralization and Real Time Hooks

The real time nature of decentralized Accountability Platform data flowing through the supply chain could enable some real-time audit hooks not available today through current standards. Ad technologies like DSPs and SSPs could add real time accountability hooks to their supply chain control features. Trade bodies, browser vendors, operating system vendors or privacy advocates could each operate different real-time solutions to support audits. Such organizations might even publish regular reports in the public domain to identify common failures or potential bad actors. By removing the delay inherent in a process of requesting log files, sending log files, joining pairwise transactions to understand what expectations were originally presented to the end user associated with this transaction, we can more rapidly detect violations and more quickly address bad actors.

By passing the entire chain of signed transactions back to the digital property (or their agent) it would provide another vector to identify fraud in real time and for good actors to not only stop the advert being served but also alert other members of the supply chain. Taken to conclusion the user's agent would be able to verify the supply chain in real time and provide an icon to the

user to indicate that the advertising respected their privacy preferences. It would also be possible for web browser vendors, or extension providers, with the permission of the user for the entire transaction supply chain to be passed to any interested party for inspection. With such aggregate information such parties could monitor the industry for compliance and assist in the identification of bad actors and harms.

Browser plug-ins could be built to inspect the transaction chain and apply privacy signal rules or increased analytics associated with this information. Browsers and OSs themselves could build to these outputs. A future feature aided by the real time nature of decentralization could enable users to respond back to that complete chain that they did not want to see a particular ad.

Summary of Further Decentralization Proposal

So long as the complete chain is available to auditors, whether it was joined by the organizations being audited in real-time (so they have access to same information as the auditor), or the data is joined daily from sample transactions, the auditability of the transactions included are the same.

Beyond the faster access to this transaction data, the complete log avoids incremental join costs involved with the recording of only pair-wise transactions in log files. This could result in a reduction in operational costs if there is a central entity which must be financially supported in the daily sample submission. Though each audit entity building an API like that described in this document could be even more costly.

Decentralization may also reduce some barriers to entry where participants do not already log and retain data. In addition to potential cost reductions, this larger sample rate might also reduce the chance bad actors can get away with violations by increasing the auditability and detectability of these violations within data transfers.

Pair-wise, sample based audit alone results in reduced transparency (vs complete audit of data), thus potentially reducing the likelihood of detecting violations and increased cost of joining batch-mode submitted data. A non-sampled, real-time transaction chain is not without its own drawbacks, which include increased data transferred in real-time proportional to length of complete chain and public disclosure of business relationships that parties may not want disclosed (vs pair-wise). Additionally, real-time processing of data by a wide range of potential audit entities is unlikely unless those entities raise their fees. The Accountability Working Group is excited to receive public comments on this late arising proposal to further extend decentralization and create more real-time hooks for rapid applications.

FAQs

How do Sender/Receiver Paris and the sample methodology discourage manipulated reporting and multiparty collusion?
It is easy to recognize that the transaction volume within the ads ecosystem is going to be too high to allow for submission and processing of comprehensive log data, so instead the intent is to have participants provide only small samples.

One challenge with reporting samples is making sure they are generated for a consistent set of records for all participants, so that the reported sets of Sender records match the sets of Receiver records. To do this, each participant must use a method of identifying entries to be included in the sample that will be consistent across all participants, which in turn requires that the method use a value available to all participants. Such a value is available in the addressable user identifiers in the the subject dataset which will be included (but not submitted) in all records with privacy signals, and which participants will communicate unaltered across the supply chain. Since there may be many varieties of identifiers used, the sample selection will be based on a consistent MD5 hash that generates a standard result given an alphanumeric input. Sampling is then done by matching low-order digits to a sample match value. Using this process, all participants will submit log samples generated for the same subset of user identifiers and submissions from participants sending and receiving data with each other should contain records that can be properly paired.

By having all participants submit records matching a given sample match value consistently, a longitudinal data set can be created that would allow for the building of a baseline that provides a sense of how data normally flows through the ecosystem which would help in detecting anomalous behaviors. It would also be useful in gauging how much data would be generated by a given sample match value magnitude.

During design discussions a potential challenge identified with sampling was the possibility that a participant might treat the sample set differently from the rest of the data stream, resulting in a sample that did not accurately represent the larger set. To address this concern, the proposed design requires participants to log data for *all* transactions within a reporting window and then use a randomly generated sample match value for that reporting submission period provided all participants to query the sample from the logs. The provision of this randomly generated sample match value is post-data collection. Since participants wouldn't be able to anticipate what record sample was going to be asked for, they would be unable to treat records differently without risk of detection. Additionally, since Sender and Receiver pairs are reporting on the same underlying transactions, it will be possible for audit entities to detect when one side of a pair reports differently from the other.

How might Sender/Receiver pairs aid compliance investigations?
The output from the joining of the submitted logs will be a record set in which each Sender record is paired with a Receiver record with the same transaction identifier and the privacy signals in both will match. A query which outputs non-matching privacy signals and unmatched

Sender, Receiver, Transaction ID instances would be straightforward to write for entities running analysis.

In cases where the privacy signals do not match, analysis outputs could be used to determine the severity of the issue as well as identify the likely source. If multiple Receivers showed errors in data coming from a given Sender, the Sender would be the likely cause. In cases where a Receiver showed errors from multiple Senders, the Receiver would likely be the cause. In either case, the records with the errors would identify areas for further investigation.

If there are unpaired records in a result set, it would be relatively straightforward to determine if it was a Sender or Receiver that was having issues providing data. In the case of Sender-related issues, the orphans would be Receiver records, while Receiver-related issues would result in orphaned Sender records. Again, in either case the orphaned records would identify who needed to be contacted for further investigation.

How to Submit Comments

Comments on this RFC may be submitted to accountability@iabtechlab.com.

