Blockchain-Based Smart Advertising Network With Privacy-Preserving Accountability

Dongxiao Liu, *Member, IEEE*, Cheng Huang, *Member, IEEE*, Jianbing Ni, *Member, IEEE*, Xiaodong Lin, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—In a smart advertising network (*SAN*), a broker builds user profiles from its wealth of user data, manages advertisements for retailers, and disseminates the advertisements through multiple channels. However, the broker sometimes provides insufficient transparency explanations of advertising activities, which may result in the increasing popularity of ad-blocking software and lower advertising investments from retailers. In this paper, we propose a blockchain-based Smart Advertising Network with Privacy-preserving Accountability (*SANPA*). Specifically, we design a composite Succinct Non-interactive Argument (*SNARG*) system, that commits advertising policies as cryptographic authenticators in a smart contract. By doing so, *SANPA* is compatible with the existing *SAN* without posing prohibitive implementation cost over the blockchain architecture. Users or retailers can require explanations of an advertising activity by sending a challenge to the smart contract. With the succinctness and privacy preservation of the *SNARG* system, the smart contract can efficiently verify whether the challenged advertising activity follows committed advertising policies without exposing user profile privacy. If any misconduct is identified, the contract enforces public accountability on the misbehaving party by confiscating its cryptocurrency deposits. We conduct extensive experiments to provide both on-chain and off-chain benchmarks, which demonstrates the application feasibility of *SANPA*.**

***Index Terms*—Smart advertising network, blockchain, transparency, accountability, privacy.**

I. Introduction

The technical advances of Internet of Things (IoT) and the next generation wireless technology are reshaping the advertising industry [1], [2]. Specifically, a smart advertising network (*SAN*) of connected intelligent objects, such as smart vehicles and smart home devices, can help retailers to effectively reach users through multiple advertising channels. For example, a user can receive advertisements of healthy diets from a smart watch, or promotion codes of nearby shopping centers from mobile devices. At the same time, with the explosive volumes of data in *SAN*, retailers are able to profile behaviors of their users and personalize their advertisements to improve ad recommendation efficiency. As a result, *SAN* is surpassing traditional advertising strategies, such as TV and billboard in 2018, and will be dominating the advertising industry in the future [3].

In practice, *SAN* is managed by a third-party broker [4]. With its ubiquitous devices and applications, the broker collects massive user behavior data to build user preference profile. For example, Google records user activities from its ecosystem including Android and Google Home devices, and assigns keyword tags to users based on their activities. Retailers can also choose a set of keywords as their targeting policies and rely on the broker to disseminate their ads to users of related interests. Later, the broker charges the retailers if any user views the ads (per view) or clicks the links in the ads (per click). By doing so, retailers can enjoy the broker's wealth of user data and advertising channels for effective ad disseminations. As a result, *SAN* has achieved great commercial success. According to eMarketer [3], Google and Facebook occupy a quarter of overall digital ad spending in the US in 2018.

There is an emerging challenge for the continuous success of *SAN*: the lack of advertising transparency [4]–[6]. First, users feel offended by the broker when they are unknowingly assigned with keyword tags. For example, a simple click on a link of sports news may give users a tag of 'football'. Second, users often find themselves receiving annoying ads that are irrelevant or even biased. For example, ad dissemination based on gender, age, and nationality is considered as ad discrimination by users [7]. Without proper countermeasures, many users prefers ad-free applications or install ad-block extensions [8] to filter out advertisements. This leads to the retailers' decrease in advertising investments, which greatly hinders the developments of *SAN*.

To regain users' confidence on *SAN*, both the industry and the academic are making efforts on increasing the advertising transparency. An initial attempt by the broker is to provide users with personal profile management tools. The brokers also provide users with options of "transparency explanations" regarding why users are receiving specific ads. For example, Facebook explains to users the sources and targeting policies of the ads. However, such explanations are usually insufficient to users [4] and can sometimes be incomplete [5]. To address the issue, there have been many research activities, that utilize

Manuscript received March 29, 2020; revised July 21, 2020; accepted September 2, 2020. Date of publication September 30, 2020; date of current version September 16, 2021. Recommended for acceptance by Dr. Sangheon Pack. *(Corresponding author: Xiaodong Lin.)*

Dongxiao Liu, Cheng Huang, and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (e-mail: dongxiao.liu@uwaterloo.ca; cheng.huang@uwaterloo.ca; xshen@bbcr.uwaterloo.ca).

Jianbing Ni is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, Ontario K7L 3N6, Canada (e-mail: jianbing.ni@queensu.ca).

Xiaodong Lin is with the School of Computer Science, University of Guelph, Guelph, Ontario N1G 2W1, Canada (e-mail: xlin08@uoguelph.ca).

Digital Object Identifier 10.1109/TNSE.2020.3027796

trusted hardware [9] and transparency extensions [6] at user side or introduce an independent organization to enhance transparency of *SAN* [10], which mainly rely on the trustworthiness of a single authority to provide transparency explanations. However, the single authority may not always be reliable. For example, it is reported that major brokers pay the developers of ad-blocking tool to make their ads on the 'whitelist' [11]. Moreover, it can sometimes be difficult to provide effective accountability against the advertising misconduct due to the profit considerations and the slow auditing process. As a result, it is critical to have a decentralized architecture to enforce publicly verifiable transparency and enforce effective accountability in *SAN*.

A solution that builds upon the blockchain architecture [12] with distributed consensus [13] is more promising for promoting public accountability in *SAN*. The blockchain is a public ledger with blocks of peer-to-peer transactions in a fully distributed network. Secured by the cryptography and consensus protocols [14], [15], the blockchain ensures a consistent and transparent view of the shared ledger among mutually distrustful nodes. If we view the blockchain as a state machine, every valid transaction will change the state of the blockchain. As a result, blockchain can be utilized as a trusted environment to execute computer programs, i.e., smart contract [16]. Specifically, the broker can store all retailer policies and user profiles onto the blockchain and design an advertising smart contract to implement the ad delivery. By doing so, a blockchain-based solution achieves two distinctive features: (1) Decentralized transparency. The ad dissemination is publicly verifiable in a distributed network [17]. (2) Automatic accountability. Any advertising misconduct can be automatically detected and publicly held accountable by confiscating cryptocurrencies of misbehaving parties. However, the solution may not be practical in real-world implementations due to the following challenges. (1) Efficiency. Since on-chain storage and computation resources are limited, directly implementing the ad delivery on the blockchain is expensive [18], [19] for *SAN*. (2) Privacy. User profiles may contain sensitive personal information, e.g., locations and interests [20], which may be exposed to the public due to the transparency nature of the blockchain. At the same time, retailer polices are required to be transparent in *SAN*. The conflict between user profile privacy and retailer policy transparency requires specific designs in a blockchain-based architecture.

In this paper, we propose a blockchain-based Smart Advertising Network with Privacy-preserving Accountability *(SANPA)*. Specifically, the broker commits to the retailer policies and ad dissemination algorithms with succinct cryptographic authenticators. The authenticators are updated to an accountability contract on the blockchain to serve as a public commitment of advertising transparency. Instead of directly implementing *SAN* on the blockchain, *SANPA* enables the broker to manage the ad dissemination in an off-chain manner. Both users and retailers can require transparency explanations about advertising activities, e.g., correct inclusion of retailer policies and correct computation of the ad dissemination process, by sending challenges to the accountability contract. With the on-chain cryptographic authenticators, the accountability contract can efficiently and publicly verify the correctness of the challenged advertising activities without sacrificing user profile privacy. At the same time, the accountability contract can hold any advertising misconduct publicly accountable, i.e. confiscating cryptocurrency deposits of misbehaving parties. By doing so, *SANPA* achieves privacy-preserving accountability for *SAN*. Specifically, the contributions are summarized as follows:

- We propose a composite *SNARG* system from Quadratic Arithmetic Program (QAP)-based relations and multivariate linear relations in the discrete logarithm setting. The composite *SNARG* system is efficient for on-chain verifications of advertising activities and preserves user profile privacy while pursuing public accountability.
- We design an accountability contract that receives challenges for transparency explanations and enforces accountability on misbehaving parties. The contract implements the composite *SNARG* system and uses the cryptocurrencies as incentives to boosting honest advertising conducts and promote prompt on-chain responses.
- Through the security analysis, we formulate and achieve *privacy-preserving accountability* in *SANPA*. Extensive experiments are conducted to demonstrate the feasibility of *SANPA*. The experimental results present comprehensive benchmarks for both the off-chain and on-chain computation and storage overheads.

The paper is organized as follows. We review the related work in Section II. In Section III, we present the smart advertising model, security model, and design goals. In Section IV, we introduce the building blocks. In Section V, we propose *SANPA*, and provide the security analysis in Section VI. We evaluate the performance of *SANPA* in Section VII, and conclude this paper in Section VIII.

II. Related Work

A. Smart Advertising Transparency

Andreou *et al.* [5] investigated the transparency explanations of Facebook advertising, regarding how a user is labeled with attributes and why the user receives a specific ad. By collecting a large amount of explanation data from different users, the authors found out that the transparency explanations by the brokers are sometimes incomplete and vague. Parra *et al.* [6] developed a detection system that looks into users' web browser profile with a measurement criteria for user profile uniqueness, which enables a configurable and flexible transparency options for web users. Venkatadri *et al.* [10] utilized a third-party organization as a transparency explainer of the adverting network for users. Li *et al.* [9] used trusted hardware techniques (ARM trusted zone) to propose a verifiable advertisement click and display framework on mobile applications.

Since existing solutions mainly utilized a single entity, insufficient attentions were directed to the distributed transparency management for *SAN*. In contrast to the existing works, *SANPA* explored the blockchain as a distributed architecture to purse public awareness of advertising transparency and automatic enforcement if the advertising misconduct is identified.

B. Blockchain-based Accountability

Frankle *et al.* [21] investigated the accountability issues in a secret process of a court system, such as a surveillance warrant of a criminal target. The authors utilized the public ledger and multi-party computation techniques to achieve the system transparency and target privacy at the same time. Subsequently, Panwar *et al.* [22] leveraged the blockchain with zero-knowledge proofs and Merkle trees to design auditing framework for legal processes to preserve the real identities of the investigative targets. Li *et al.* [23] constructed a blockchain-based vehicular forensics framework that enforces accountability and fine-grained access control over the forensics data. The authors built a distributed Key Policy Attribute-based Encryption (KP-ABE) scheme that prevents malicious investigators to abuse the power. Neisse *et al.* [24] studied the General Data Protection Regulation (GDPR) for personal data usages and proposed a blockchain-based framework that realized personal data accountability and provenance tracking. Nguyen *et al.* [25] utilized the blockchain to construct a geo-marketplace for trading location data in a transparent and accountable manner.

Compared with the existing works, *SANPA* carefully investigated the characteristics of *SAN* and proposed a blockchain-based architecture for enhancing public accountability in *SAN*. *SANPA* also introduced an on/off chain model with an accountability contract that significantly increased the system efficiency for practical implementations.

C. SNARG System

In *SANPA*, we focus a research line of non-interactive *SNARG* systems in bilinear groups, that recognize the arithmetic circuit evaluations and the cryptographic algebra computations.

Gennaro *et al.* [26] exploited the *QAP* theory, that converts arithmetic circuit evaluations to divisibility checks for low-degree polynomials in bilinear groups. The proposed *SNARG* system [26] is non-interactive and succinct, which results in a notable storage and computation efficiency at the verifier. Subsequently, Parno *et al.* [27] adopted a more efficient *QAP* construction, that significantly reduced the degree of compiled programs, the key size and prover computation overhead. Ben-Sasson *et al.* [28] proposed a set of optimization techniques to improve the computation and storage efficiency of the Pinocchio framework [27]. Later, the proof size and verifier computation overhead of *QAP*-based *SNARG* system was further reduced in [29]. Agrawal *et al.* [30] observed that it is more efficient to combine *QAP*-based *SNARG* and traditional Sigma protocols for the instantiation of composite statements. Campanelli *et al.* [31] formalized a framework with modular utilizations of *SNARG* systems from *QAP*-based relations and algebra relations in bilinear groups.

It is a non-trivial task to design a composite *SNARG* system for *SANPA*. First, there are different privacy requirements for transparency explanations. Specifically, user profiles should be kept private while retailer policies are required to be transparent to the public. Second, different *SNARG* systems have different instantiations for different relations. A simple combination of traditional *SNARG* systems may not achieve desired properties of *succinctness*, *soundness*, and selective *privacy preservation*. As a result, the composite *SNARG* system requires careful decomposition of *SAN* functionalities and modular designs of a set of *SNARG* systems from different instantiations.

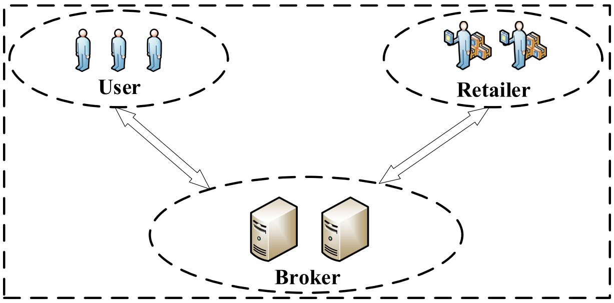


Fig. 1. Smart advertising model.

III. Problem Formulation

In this section, we first formulate the smart advertising model in terms of entities and the ad dissemination strategy. Then, we formalize the security model and the design goals.

A. Smart Advertising Model

We abstract the existing *SAN* model in Fig. 1, which consists of three entities: Broker, Retailer, and User.

- *User*: Users are equipped with multiple devices, e.g., mobile phones or tablets. They run a wide range of applications and can receive advertisements from multiple channels, e.g., web search or application push messages.
- *Retailer*: Retailers are shops or stores that wish to promote their products by advertisements. Retailers rely on the broker to manage their targeting policies and pay the broker for the ad dissemination services.
- *Broker*: Broker is a third-party advertising company (e.g., Google Ads). It manages user preference profiles and retailer targeting policies, and charges retailers based on per-view or per-click model.

In *SANPA*, we consider the ad dissemination with the popular keyword matching strategy between user profile and retailer targeting policies [32]. Specifically, a keyword dictionary $\mathcal{D} = \{\mathcal{W}_1, \mathcal{W}_2, ..., \mathcal{W}_n\}$ consists of n keywords. n is a few hundreds for a subcategory of the keyword space in Google Ads. The broker can assign each user a set of keywords from the user's interaction with the broker's applications, and constructs the user preference profile S_u. At the same time, the user can also access and modify her/his preference profile. Each retailer selects a set of keywords from $\mathcal{D}$ and constructs a targeting policy S_r. The broker measures the similarity between the user profile and retailer policies, and returns the user with advertisements that are most relevant to her preference profile.

B. Security Model

Users and retailers are both *rational*. That is, either users or retailers will only challenge the advertising system, if there are concerns on the advertising transparency. They will also accept transparency explanations if the explanations are

TABLE I
NOTATIONS

$\mathbb{G}$	Multiplicative groups
$\mathcal{R}$	Polynomial-time decidable relation
(x, w)	Statement x, witness w
$\mathbf{x}^n$	n-dimension vector
$\mathbf{X}^{m*n}$	$m * n$-dimension matrix
$[n]$	Integers from 1 to n
$\in_R$	Choose a random number
$\mathbf{Com}(\mathbf{x}, CK)$	Cryptographic commitment **Com** Input vector **x**, commitment key CK
$\mathbb{C} = (EK, VK)$	Common reference string $\mathbb{C}$ Evaluation key EK, verification key VK

publicly verifiable. The broker is a multi-sector enterprise, that may not always follow the pre-determined ad dissimilation strategy, due to profit considerations, slow internal processes, and the lack of public auditings. Under the security model, we define the security goal as *privacy-preserving accountability* and present its progressive meanings as follows:

Definition 1. Privacy-preserving Accountability

- Public Verifiability: Users and retailers can require the broker to provide advertising transparency explanations about the ad dissemination process and retailer policy management, the correctness of which should be publicly verifiable.
- Privacy: User preference profiles are concealed from the public view, even in a publicly verifiable transparency explanation.
- Accountability: Timely and automatic obligations enforcement on the broker should be achieved in case of any advertising misconduct.

C. Design Goals

SANPA should achieve the following design goals:

- Compatibility: *SANPA* should support the ad dissemination with the keyword matching strategy.
- Security: *SANPA* should achieve *privacy-preserving accountability* for the smart advertising network.
- Efficiency: *SANPA* should incur applicable overhead to the smart advertising network.

IV. PRELIMINARIES

In this section, we present the preliminaries in *SANPA*, including cryptographic commitment schemes, *SNARG* systems, and digital signature schemes. (1) The cryptographic commitment is utilized to securely digest targeting policies and user profiles into a succinct authenticator. (2) *SNARG* systems can achieve verifiable on-chain transparency explanations. (3) Digital signature is used to generate non-repudiable off-chain receipts of advertising activities. Notations are shown in Table I.

A. Notations

$\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ denote three cyclic multiplicative groups [33] with a prime order p and a bilinear pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \longrightarrow \mathbb{G}_T$. $\mathbb{Z}_p$ denotes a ring of integers modules p. $r \in_R \mathbb{Z}_p$ indicates r is randomly chosen from $\mathbb{Z}_p$. $\mathbb{F}$ denotes a finite field. $[n]$ denotes integers from 1 to n. A bold lower letter $\mathbf{x}^n \in \mathbb{F}^n$ denotes an n-dimension vector from $\mathbb{F}$. A bold capital letter $\mathbf{X}^{m*n} \in \mathbb{F}^{m*n}$ denotes an $m * n$ matrix from $\mathbb{F}$. We denote $\mathcal{R}$ as a polynomial-time decidable relation with a statement x and a witness w [31]. $(x, w) \in \mathcal{R}$ indicates that $\mathcal{R}$ holds on a pair (x, w), which can be efficiently decided by a non-interactive argument system [30].

B. Cryptographic Commitment

Cryptographic commitment schemes [34] allow a party to commit to a secret value, such as an integer or a vector of integers. The commitment can either be directly revealed to the public, or combined with the zero-knowledge proof technique [35] to demonstrate that the committed value satisfies a public relation without leaking the value. We define the cryptographic commitment as follows:

Definition 2. A cryptographic commitment of value x using a commitment key CK is denoted as $\mathbf{Com}(x, CK)$.

A typical example of commitment schemes in the cyclic multiplicative groups is Pedersen commitment. Given $x, r \in_R \mathbb{Z}_p^2$ where r is the randomness, and $CK = (g_1, g_2) \in \mathbb{G}_1^2$, a Pedersen commitment $\mathbf{Com}(x, CK) = g_1^x g_2^r$. Given a vector $\mathbf{x}_n$, $r \in_R \mathbb{Z}_p$, and $CK' = (g, g_1, g_2, ..., g_n) \in \mathbb{G}_1^{n+1}$, an extended Pedersen commitment $\mathbf{Com}(\mathbf{x}^n, CK') = g^r \prod_{i=1}^n g_i^{x_i}$.

C. Succinct Non-interactive ARGuments (SNARG)

The *SNARG* system allows a prover to demonstrate that a public relation $\mathcal{R}$ holds on a pair (x, w) to a verifier, which can be defined as follows:

Definition 3. A *SNARG* system $\sum$ for a relation $\mathcal{R}$ consists of three algorithms:

- $KeyGen(\mathcal{R}, pp) \rightarrow \mathbb{C} = (EK, VK)$
- $Prove(EK, x, w) \rightarrow \pi$
- $Verify(VK, x, \pi) \rightarrow (0, 1)$

$KeyGen$ takes as inputs the relation $\mathcal{R}$ and public system parameters pp, and outputs the common reference string $\mathbb{C}$ with EK and VK. *Prove* takes as inputs the EK, a statement x, and a witness w. *Prove* evaluates $\mathcal{R}$ on (x, w) and generates a proof π. $Verify$ takes VK, the statement x, and the proof π. It outputs 1 if $(x, w) \in \mathcal{R}$; it outputs 0, otherwise. We define four security notions of *SNARG* systems as follows:

- *Completeness*: A rational verifier will accept (x, π) if $(x, w) \in \mathcal{R}$ and π is correctly computed.
- *Soundness*: A computationally-bounded adversary cannot forge an invalid tuple (x', w', π'), such that $(x', w') \notin \mathcal{R}$ and $Verify(VK, x', \pi') \rightarrow 1$.
- *Succinctness*: The proof length is only determined by the system security parameter.
- *Privacy Preservation*: The verifier only learns whether $(x, w) \in \mathcal{R}$.

For different relations $\mathcal{R}$, *SNARG* systems can be categorized into subsets with different instantiations and security notions. In *SANPA*, we consider two categories: (1) Multivariate linear relations in the discrete logarithm (DLog) setting.

(2) Quadratic Arithmetic Program (*QAP*) based *SNARG* in bilinear groups.

1) Multivariate Linear Relations in the DLog: The *SNARG* system for multivariate linear relations can be efficiently recognized by discrete logarithms in multiplicative groups with a prime order. In *SANPA*, we focus on two specific *SNARG* systems in the DLog: (1) the equality test for two Pedersen vector commitments, and (2) the succinct openness for a subset of a Pedersen vector commitment [36], [37].

Definition 4. Two extended Pedersen commitments for the same vector $\mathbf{x}^n$ with different commitment keys CK_1, CK_2 are defined as follows:

$$\mathbf{Com}(\mathbf{x}^n, CK_1) = \prod_{i=1}^{n} g_i^{x_i},\ CK_1 = (g_1, ..., g_n) \in \mathbb{G}_1^n$$

$$\mathbf{Com}'(\mathbf{x}^n, CK_2) = \prod_{i=1}^{n} g_i'^{x_i},\ CK_2 = (g_1', ..., g_n') \in \mathbb{G}_1^n$$

A *SNARG* system $\sum_{\mathcal{V}}$ enables a prover with knowledge $\mathbf{x}^n$ to convince a verifier that $\mathbf{Com}$ and $\mathbf{Com}'$ open to the same vector $\mathbf{x}^n$.

Definition 5. Consider $\mathbf{R}^{m*n} \in \mathbb{Z}_P^{m*n}$, $\mathbf{y}^n \in \mathbb{Z}_P^n$ is a subset of $\mathbf{R}^{m*n}$ indexed by $I_S = \{(i_1, j_1), (i_2, j_2), ..., (i_n, j_n)\}$, a Pedersen commitment for $\mathbf{R}$ is defined as follows:

$$\mathbf{Com}(\mathbf{R}, CK) = \prod_{i=1}^{m} \prod_{j=1}^{n} g_{i,j}^{\mathbf{R}_{i,j}},\ CK = \{g_{i,j}\} \in \mathbb{G}_1^{m*n}$$

A *SNARG* system $\sum_{\mathcal{S}}$ enables a prover with knowledge $\mathbf{R}$ to convince a verifier: the subset of $\mathbf{Com}(\mathbf{R}, CK)$ indexed by I_S opens to $\mathbf{y}^n$.

We change the position of randomness r in the original Pedersen commitment to a position in $\mathbf{x}$ or $\mathbf{R}$. Both $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ achieve *completeness*, *soundness* and *Succinctness* for secure and efficient verifications. We will present the detailed designs of $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ in Section V.

2) QAP-based SNARG *in Bilinear Groups:* Evaluations of $\mathcal{R}$ on a pair (x, w) is equivalent to the circuit satisfiability evaluations with certain inputs. Gennaro et al. [26] proposed a technique to convert the evaluation of an arithmetic circuit $\mathcal{C}$ to the divisibility check of a Quadratic Arithmetic Program (*QAP*) $\mathcal{Q}$. In specific, $\mathcal{Q}$ consists of three sets of polynomials $V = \{v_k(x)\}$, $W = \{w_k(x)\}$, $Y = \{y_k(x)\}$, where $k \in [0, z]$ and z denotes the number of input, intermediate and output wires in $\mathcal{C}$. A target polynomial $t(x)$ is defined by picking a random root for each multiplication gate in $\mathcal{C}$. An input $(a_1, a_2, ..., a_o) \in \mathbb{F}^o$ and an output $(a_{z-p+1}, a_{z-p+2}, ..., a_z) \in \mathbb{F}^p$ are valid assignments of $\mathcal{C}$, iff $(a_{o+1}, a_{o+2}, ..., a_{o+q}) \in \mathbb{F}^q$ can be found such that $t(x)$ can divide $p(x)$, where $z = o + p + q$.

$$p(x) = (v_0(x) + \sum_{k=1}^{z} a_k v_k(x)) \times (w_0(x) + \sum_{k=1}^{z} a_k w_k(x)) - (y_0(x) + \sum_{k=1}^{z} a_k y_k(x)) \tag{1}$$

$(a_{o+1}, a_{o+2}, ..., a_{o+q})$ actually denote the assigned values of intermediate multiplication wires.

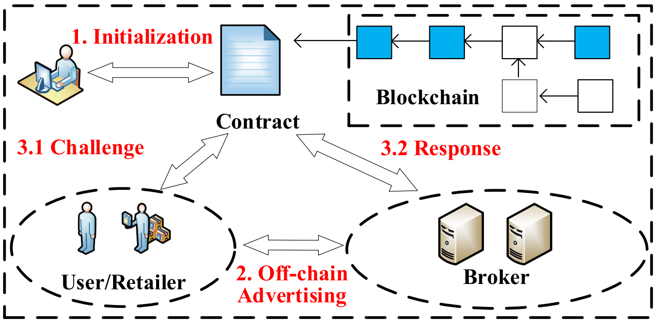


Fig. 2. SANPA workflow.

We define a relation $\mathcal{R}_{\mathcal{Q}}$ that decides on a pair $(x, \mathbf{w}^z)$ for an arithmetic circuit $\mathcal{C}$. $\mathbf{w}^z \in \mathbb{F}^z$ corresponds to input, output and intermediate multiplication wires of $\mathcal{C}$. Based on Equation 1, $\mathcal{R}_Q$ is represented by a linear combination of $\mathbf{w}^z$, which can be efficiently evaluated in pairing-friendly bilinear groups. Adopting techniques from [27], [29], [37], we obtain a *SNARG* system $\sum_{\mathcal{Q}}$ for $\mathcal{R}_{\mathcal{Q}}$.

D. Digital Signature

A digital signature scheme consists of three algorithms:

- $Gen(\mathbb{G}, 1^\lambda) \rightarrow (pk, sk)$
- $Sig(m)_{sk} \rightarrow \pi_s$
- $Veri(m, \pi_s)_{pk} \rightarrow (0, 1)$

$KeyGen$ takes into $\mathbb{G}$ and the security parameter λ, and outputs a public/private key pair (pk, sk). Sig takes into a message m and a secret key sk, and outputs a signature π_s. $Veri$ takes into a message and a signature. It outputs 1 if the verification passes; it outputs 0, otherwise. We utilize *ECDSA* signature [38] in *SANPA*, which is compatible in the Ethereum.

V. Smart Advertising Network with Privacy-Preserving Accountability

In this section, we first give an overview of *SANPA* including *System Model*, *Design Ideas*, and *Workflow*. Then, we present the details of *SANPA*, in terms of *Initialization*, *Off-chain Smart Advertising* and *On-chain Transparency Explanation*.

A. Overview

1) System Model: In *SANPA*, we introduce two additional entities to *SAN*: a distributed committee (DC) and a public blockchain in Fig. 2. (1) DC can be a set of independent supervising authorities running a Secure Multi-party Computation (SMC) protocol. For example, Zerocash has implemented an SMC protocol [39] to setup the blockchain system. (2) Blockchain is a public ledger, e.g., Ethereum. It is maintained by peer-to-peer blockchain miners and supports secure and automatic executions of smart contracts. We also assume that secure and authenticated off-chain channels are established among all entities.

2) Design Ideas: *SANPA* introduces an on/off chain computation model for the blockchain-based architecture: (1) User profile and retailer policy managements, and the ad dissemination are conducted in an off-chain manner by the ad broker. *SANPA* requires each advertising activity is non-repudiable by the broker, which can be achieved by using digital signatures. (2) Broker advertising activities are publicly audited with effective accountability enforcements in an on-chain manner, if required by users or retailers. (3) User profile privacy is guaranteed in the public transparency explanations with the design of a composite SNARG system. By doing so, *SANPA* achieves distributed, efficient, and privacy-preserving transparency explanations.

3) Workflow: In Fig. 2, *SANPA* consists of the following three phases:

(1) *Initialization.* DC sets up the advertising policies by initializing a set of *SNARG* systems $\sum_{\mathcal{Q}}$, $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$. $\sum_{\mathcal{S}}$ verifies that retailer policies are correctly managed by the ad broker. $\sum_{\mathcal{V}}$ verifies that the retailer policies are correctly entered into the ad dissemination process. $\sum_{\mathcal{Q}}$ verifies that the ad dissemination process follows the pre-determined keyword matching strategies. DC creates an accountability contract to store evaluation keys of the *SNARG* systems. More details are given in Section V-B.

(2) *Off-chain smart advertising.* Users and retailers register themselves at the broker. Users manage their preference profiles generated by the broker. Retailers set their targeting policies. The broker runs the ad dissemination process with the keyword matching strategy between the user preference profile and retailer targeting policies. The broker finds most relevant advertisements and sends them to the user.

(3) *On-chain transparency explanation.* Users and retailers can make challenges to the accountability contract and require transparency explanations on broker activities, e.g., the ad dissemination process and the retailer policy management. The broker must response to the challenges promptly, by updating correctness proofs of the advertising activities to the contract. The contract verifies the proofs from the broker and enforces obligations if any advertising misconduct is identified.

B. Initialization

DC chooses a system security parameter λ and a set of asymmetric multiplicative groups $\mathbb{G} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ with a prime order p and a bilinear pairing e. $g \in \mathbb{G}_1$ and $\tilde{g} \in \mathbb{G}_2$ are two random generators. DC sets the system public parameter $pp = (\mathbb{G}, p, e, g, \tilde{g})$. DC sets a composite relation as follows:

$$\begin{cases} (S_u, S_{\mathcal{R}}, S_o) \in \mathcal{R}_{\mathcal{Q}} \\ \wedge (D_{\mathcal{R}} = \mathbf{Com}(S_{\mathcal{R}}, CK_{\mathcal{R}}), D_{\bar{\mathcal{R}}} = \mathbf{Com}(S_{\mathcal{R}}, CK_{\bar{\mathcal{R}}})) \in \mathcal{R}_{\mathcal{V}} \\ \wedge (D_{\mathcal{R}}, S_r, I_S) \in \mathcal{R}_{\mathcal{S}} \end{cases} \tag{2}$$

DC abstracts the ad dissemination process in Algorithm 1 as a relation $\mathcal{R}_{\mathcal{Q}}$ on a pair $(S_u \in \mathbb{Z}_P^n, S_{\mathcal{R}} \in \mathbb{Z}_P^{m*n}, S_o \in \mathbb{Z}_P^k)$. S_u corresponds to an n-dimension user profile. $S_{\mathcal{R}}$ corresponds to m retailer targeting policies, each of which is also n-dimensional. The process outputs S_o, that consists of k retailer identifiers with the most relevant keywords to the user profile. $D_{\mathcal{R}}, D_{\bar{\mathcal{R}}}$ are commitments generated under different commitment keys $CK_{\mathcal{R}}, CK_{\bar{\mathcal{R}}}$. $S_r \in S_{\mathcal{R}}$ is the keyword set for an individual retailer indexed by I_S.

DC initializes the above relations with *SNARG* systems $\sum_{\mathcal{Q}}, \sum_{\mathcal{V}}$, and $\sum_{\mathcal{S}}$. The evaluation of the relation $\mathcal{R}_{\mathcal{Q}}$ is achieved by the design of the $\sum_{\mathcal{Q}}$. To preserve user profile privacy, the $Verify$ algorithm of $\sum_{\mathcal{Q}}$ takes a commitment $D_{\bar{u}\bar{o}}$ (a commitment of S_u and S_o) and a commitment $D_{\not{\mathcal{R}}}$ of retailer policies. However, the original commitment $D_{\not{\mathcal{R}}}$ in the $\sum_{\mathcal{Q}}$ does not support efficient verifications of retailer challenges. Therefore, a *SANRG* system $\sum_{\mathcal{S}}$ is designed to succinctly reveal the retailer policy from a well-structured external commitment $D_{\mathcal{R}}$. At the same time, *SANPA* proves that $D_{\bar{\mathcal{R}}}$ and $D_{\mathcal{R}}$ open to the same vector commitments with the design of the $\sum_{\mathcal{V}}$.

1) $\sum_{\mathcal{Q}}$ CRS Setup: DC instantiates $\sum_{\mathcal{Q}}$ as a Pinocchio *SNARG* system [27], with the following three algorithms:

- $KeyGen(\mathcal{R}_{\mathcal{Q}}, pp) \rightarrow \mathbb{C}_{\mathcal{Q}} = (EK_{\mathcal{Q}}, VK_{\mathcal{Q}})$
- $Prove(EK_{\mathcal{Q}}, (S_u, S_{\mathcal{R}}, S_o)) \rightarrow \pi_{\mathcal{Q}}$
- $Verify(VK_{\mathcal{Q}}, \pi_{\mathcal{Q}}) \rightarrow (0, 1)$

$KeyGen$ generates CRS $\mathbb{C}_{\mathcal{Q}}$ with an evaluation key $EK_{\mathcal{Q}}$ and a verification key $VK_{\mathcal{Q}}$. $Prove$ evaluates $(S_u, S_{\mathcal{R}}, S_o)$ and outputs a correctness proof $\pi_{\mathcal{Q}}$. $Verify$ outputs 1 if $(S_u, S_{\mathcal{R}}, S_o) \in \mathcal{R}_{\mathcal{Q}}$; Otherwise, it outputs 0. We omit the detailed constructions of the three algorithms, but note that the existence of Pedersen-like commitment keys $(CK_{\bar{u}} \in \mathbb{G}_1^n, CK_{\bar{\mathcal{R}}} \in \mathbb{G}_1^{m*n}, CK_{\bar{o}} \in \mathbb{G}_1^k) \in EK_{\mathcal{Q}}$ and Pedersen commitments $(D_{\bar{u}\bar{o}} \in \mathbb{G}_1, D_{\not{\mathcal{R}}} \in \mathbb{G}_1) \in \pi_{\mathcal{Q}}$.

2) $\sum_{\mathcal{V}}$ CRS Setup: DC chooses a set of random numbers $\mathbf{Z} \in_R \mathbb{Z}_P^{m*n}$, and computes $CK_{\mathcal{R}} = \{CK_{\mathcal{R}_{i,j}} = g^{\mathbf{Z}_{i,j}}\}_{i \in [m], j \in [n]} \in \mathbb{G}_1^{m*n}$. DC chooses random generators $R = \{R_{i,j}\}_{i \in [m], j \in [n]} \in_R \mathbb{G}_1^{m*n}$ and $\alpha, \beta, \gamma \in_R \mathbb{Z}_P^3$. DC computes $J = \tilde{g}^{\alpha}, K = \tilde{g}^{\beta}$, $L = \tilde{g}^{\gamma}$. DC computes $T_{i,j} = CK_{\mathcal{R}_{i,j}}^{\alpha} R_{i,j}^{\beta} CK_{\not{\mathcal{R}}_{i,j}}^{\gamma}$. DC sets $T = \{T_{i,j}\} \in \mathbb{G}_1^{m*n}$. The CRS of $\sum_{\mathcal{V}}$ is as follows:

$$EK_{\mathcal{V}} = (R, T), VK_{\mathcal{V}} = (J, K, L)$$

3) $\sum_{\mathcal{S}}$ CRS Setup: DC computes $\widetilde{CK}_{\mathcal{R}_{i,j}} = \tilde{g}^{\mathbf{Z}_{i,j}}, \forall i \in [m]$, $j \in [n]$ and sets $\widetilde{CK}_{\mathcal{R}} = \{\widetilde{CK}_{\mathcal{R}_{i,j}}\} \in \mathbb{G}_2^{m*n}$. DC computes $EK_{\mathcal{S}_{(i,j)(i',j')}} = g^{\mathbf{Z}_{i,j}\mathbf{Z}_{i'j'}}, \forall (i, i') \in [m],\ (j, j') \in [n], (i, j) \neq (i'$, $j')$. DC sets $EK_{\mathcal{S}} = \{EK_{\mathcal{S}_{(i,j)(i',j')}}\} \in \mathbb{G}_1^{m^2 n^2 - mn}$.

4) CRS Distribution: DC sends $\{pp, CK_{\mathcal{R}}, \widetilde{CK}_{\mathcal{R}}, CK_{\bar{u}}$, $CK_{\bar{o}}\}$ to users and retailers, and $\{pp, EK_{\mathcal{Q}}, EK_{\mathcal{V}}, EK_{\mathcal{S}}$, $CK_{\mathcal{R}}, \widetilde{CK}_{\mathcal{R}}\}$ to the broker.

For $\sum_{\mathcal{S}}$ and $\sum_{\mathcal{V}}$ from the DLog, the common reference string consists of sets of non-identical generators, whose well-formedness can be easily checked by the public. For $\sum_{\mathcal{Q}}$ from the *QAP* theorem, a trapdoor secret is used to generate the evaluation and verification keys, which can be securely computed by DC with a SMC protocol or must be securely destroyed if the secret is generated by a single authority.

C. Off-chain Smart Advertising

The off-chain smart advertising consists of *Registration*, *Retailer Policy Archiving* and *Ad Dissemination*. We assume secure and authenticated communication channels [40] are established among users, retailers and the broker.

1) Registration: The broker registers itself at DC with a public key $pk_{\mathcal{A}}$ of *ECDSA* signature and a blockchain address $addr_{\mathcal{A}}$. DC validates the identity of the broker and the well-formedness of $pk_{\mathcal{A}}$.

A user registers herself/himself at the broker with a universal ID ID_u and a public key pk_u of *ECDSA* signature. Similar to preference tag management in Google Ads, the user can obtain a set of keywords provided by the broker from the keyword dictionary $\mathcal{D} = \{\mathcal{W}_1, \mathcal{W}_2, ..., \mathcal{W}_n\}$. The user further sets a preference profile S_u as follows:

$$S_u = \{K_{u,i}\}_{i\in[n]} \begin{cases} K_{u,i} = 0, \ if \ \mathcal{W}_i \ is \ not \ selected \\ K_{u,i} = r_i, \ r_i \in_R \mathbb{Z}_P, \ otherwise \end{cases} \tag{3}$$

The user sends S_u to the broker. The broker sets $m_u = (ID_u, S_u, T_u)$, where T_u is a valid time stamp. The broker computes a signature as $Sig_{sk_{\mathcal{A}}}(m_u) \rightarrow \pi_u$ and sets the evidence of user preference profile as follows:

$$Evid_u = (m_u, \pi_u) \tag{4}$$

The broker sends $Evid_u$ to the user and stores (ID_u, S_u, pk_u) at its storage. The user checks S_u and $Veri(m_u, \pi_u)_{pk_{\mathcal{A}}} \rightarrow 1$ and sends back an acknowledgement to the broker if the checks pass, which ensures the correctness of the $Evid_u$.

A retailer registers herself/himself at the broker, with a universal ID ID_r, a public signature key pk_r and a targeting policy S_r as follows:

$$S_r = \{K_{r,i}\}_{i\in[n]} \begin{cases} K_{r,i} = 0, \ if \ \mathcal{W}_i \ is \ not \ selected \\ K_{r,i} = r_i, r_i \in_R \mathbb{Z}_P, \ otherwise \end{cases} \tag{5}$$

For representation simplicity, we assume that there are m retailers in *SANPA* that are sequentially indexed. We denote $r \in [m]$ as the index number of the retailer ID_r, which means that $ID_r = r$. The broker sets $VK_r = \{CK_{\mathcal{R}_{r,j}}\}_{j\in[n]} \in \mathbb{G}_1^n$, where $\widetilde{CK}_{\mathcal{R}_{r,j}}$ is the (r, j)-th item in $\widetilde{CK}_{\mathcal{R}}$. Similarly, the broker sets $\widetilde{VK}_r = \{\widetilde{CK}_{\mathcal{R}_{r,j}}\}_{j\in[n]} \in \mathbb{G}_2^n$. The broker sets $m_r = (ID_r, S_r, VK_r, \widetilde{VK}_r, T_r)$, where T_r is a time stamp. The broker computes a signature $\pi_r = Sig_{sk_{\mathcal{A}}}(m_r)$ and sets the evidence of the retailer policy $Evid_r$ as follows:

$$Evid_r = (m_r, \pi_r) \tag{6}$$

The broker sends the $Evid_r$ to the retailer and stores (ID_r, S_r, pk_r) at its local storage. The retailer checks that VK_r and $\widetilde{VK}_r$ are correctly chosen from $CK_{\mathcal{R}}$ and $\widetilde{CK}_{\mathcal{R}}$, and $Veri(m_r, \pi_r)_{pk_{\mathcal{A}}} \rightarrow 1$. The retailer sends an acknowledgement to the broker if all checks pass.

2) Retailer Policy Archiving: The broker collects targeting policies from m retailers as $S_{\mathcal{R}} = (S_1, S_2, ..., S_m)$. We denote $K_{i,j}$ as the j-th item in the keyword set of the i-th retailer S_i. The broker computes a cryptographic commitment $D_{\mathcal{R}}$ as follows:

$$D_{\mathcal{R}} = \prod_{i=1}^{m}\prod_{j=1}^{n} CK_{\mathcal{R}_{i,j}}^{K_{i,j}} \tag{7}$$

The broker uploads $D_{\mathcal{R}}$ to the accountability contract.

Algorithm 1: Ad Dissemination with Exact Keyword Matching

```
Input:User profile $S_u$, retailer policies $S_{\mathcal{R}}$
Output:Retailer Identifier Set $S_o$
Set $S_o$, $S_{temp}$ to be empty
for $S_i \in S_{\mathcal{R}}$do
  Set $flag$ to be 1
  for $K_{u,j} \in S_u$ do
    if $K_{u,j} \neq 0$ & $K_{i,j} = 0$
      Set $flag = 0$then
  if$flag = 1$ then
    Add the retailer identifier $ID_i$ to $S_{temp}$
Add $k$ identifiers of $S_{temp}$ to $S_o$
```

3) Ad Dissemination: The broker generates the non-repudiable evidence of ad disseminations for users with the following steps:

1) The user generates an ad request $m_{sid} = (sid, ID_u, T_{sid})$, where sid is a session id and T_{sid} is a time stamp. The user computes $\pi_{sid} = Sig(m_{sid})_{sk_u}$ and sends (m_{sid}, π_{sid}) to the broker.
2) The broker checks the time stamp, the user ID ID_u and the freshness of sid. If $Veri(m_{sid}, \pi_{sid})_{pk_u} \rightarrow 1$, the broker retrieves user preference profile S_u and conducts the ad dissemination in Algorithm 1.
3) The broker computes:

$$\begin{aligned} &D_{\bar{o}} = \prod_{i=1}^{k} CK_{\bar{o}_i}^{ID_i \in S_o}, \ D_{\bar{u}} = \prod_{i=1}^{n} CK_{\bar{u}_i}^{K_{u,i}} \\ &D_{\bar{u}\bar{o}} = D_{\bar{u}} D_{\bar{o}}, \ \pi'_{sid} = Sig_{sk_{\mathcal{A}}}(m'_{sid}) \end{aligned} \tag{8}$$

$m'_{sid} = (D_{\bar{u}\bar{o}}, m_{sid}, T'_{sid})$. The broker stores (ID_u, sid, S_o) at its storage and sends $(S_o, m'_{sid}, \pi'_{sid})$ to the user.
4) If $Veri(m'_{sid}, \pi'_{sid})_{pk_{\mathcal{A}}} \rightarrow 1$ and $D_{\bar{u}\bar{o}}$ is correctly computed with $S_u, S_o, CK_{\bar{u}}, CK_{\bar{o}}$, the user sets $Evid_{sid} = (m'_{sid}, \pi_{sid}, \pi'_{sid})$. The user also generates an acknowledgement for the broker to charge the retailer.

D. On-chain Transparency Explanation

In this section, we first define two types of challenges about the broker advertising activities. Second, we present an overview of the accountability contract. Third, we present the design details of the accountability contract.

1) Advertising Challenge: Two types of transparency challenges are designed in *SANPA*: (1) *Retailer Challenge*: A retailer can make a challenge with $Evid_r$ to the accountability contract to check whether the targeting policy S_r is correctly included in the on-chain authenticator $D_{\mathcal{R}}$. (2) *User Challenge*: A user can make a challenge with $Evid_{sid}$ to the accountability contract to check whether the ad dissemination process is correctly conducted.

2) Overview: The accountability contract consists of four phases: *Initialization*, *Challenge*, *Resolve* and *Claim*.

- *Initialization.* DC initializes system public parameters and creates the accountability smart contract.
- *Challenge.* Retailers or users send their evidence to the accountability contract to require transparency explanations for specific advertising activities.

- *Resolve.* The broker retrieves the challenges from the accountability contract and proves the correctness of the challenged advertising activities within a pre-determined threshold time. The broker uploads the correctness proof of the challenged activities to the contract and claims the deposits of the challenger if the proof passes the verification.
- *Claim.* If the broker does not provide the proof within the threshold time, users or retailers can send a request to the accountability contract and claim deposits of the broker.

3) Accountability Contract: We utilize the public Ethereum blockchain to implement the accountability contract. In *SANPA*, we require that DC is associated with an Ethereum blockchain address that is known to the public. In the following, we present the detailed designs of the above four phases:

Initialization. The broker registers its public key $pk_{\mathcal{A}}$ and blockchain address $addr_{\mathcal{A}}$ at DC. DC initializes the *SNARG* systems for the on-chain transparency explanations. For the retailer challenge, DC instantiates the *SNARG* system $\sum_{\mathcal{S}}$, that is to prove a retailer's targeting policy S_r is correctly included in $D_{\mathcal{R}}$. For the user challenge, *SANPA* designs a composite *SNARG* system with $\sum_{\mathcal{Q}}$ and $\sum_{\mathcal{V}}$. DC creates the accountability contract in Algorithm 2. The contract stores the broker's public key and blockchain address, the public system parameters, verification keys of *SNARG* systems, and the authenticator of retailer policies $D_{\mathcal{R}}$. The contract also stores a threshold time T_H and takes initial deposits $C_{\mathcal{A}}$ from the broker.

Challenge. Users and retailers make challenges of advertising activities to the accountability contract. *UserChallenge* function takes evidence and deposits C_u from users. *RetailerChallenge* function takes $Evid_r$ and deposits C_r from a retailer. Both of the functions will verify the signatures of the evidence, set the time stamp for the challenge, and record the valid evidence on the contract.

Resolve. We present the details in terms of user changes and retailer challenges.

For user challenges, the broker retrieves unprocessed user challenges and conducts off-chain processing via *Prove* algorithms of *SNARG* systems $\sum_{\mathcal{Q}}$ and $\sum_{\mathcal{V}}$ as follows:

- From Rec_U, the broker obtains ID_u and $Evid_{sid}$. The broker retrieves S_u with ID_u and $S_{\mathcal{R}}$ from its storage. S_u is the user preference profile and $S_{\mathcal{R}}$ is the keyword sets of all retailers.
- The broker runs $Prove(EK_{\mathcal{Q}}, (S_u, S_{\mathcal{R}}, S_o))$ to obtain a proof $\pi_{\mathcal{Q}}$ for $\mathcal{R}_{\mathcal{Q}}$. Note that, there are commitments $D_{\bar{\mathcal{R}}} = \mathbf{Com}(S_{\mathcal{R}}, CK_{\bar{\mathcal{R}}}), D_{\bar{u}\bar{o}} = \mathbf{Com}(S_u, CK_{\bar{u}}) \cdot \mathbf{Com}(S_o, CK_{\bar{o}}) \in \pi_{\mathcal{Q}}$.
- With $(T, R) \in EK_{\mathcal{V}}$, the broker runs $Prove$ function of $\sum_{\mathcal{V}}$ as follows:

$$X = \prod_{i=1}^{m}\prod_{j=1}^{n} T_{i,j}^{K_{i,j}}, Y = \prod_{i=1}^{m}\prod_{j=1}^{n} R_{i,j}^{K_{i,j}}, K_{i,j} \in S_{\mathcal{R}} \tag{9}$$

- The broker sets $\pi_{\mathcal{V}} = (X, Y)$.

For retailer challenges, the broker retrieves unprocessed retailer challenges from Rec_R by ID_r, and conducts $Prove$ algorithm of $\sum_{\mathcal{S}}$. Specifically, the broker obtains ID_r, S_r from $Evid_r$ and EK_S from its storage. We denote the index I_S for S_r as $\{(r,1),(r,2),...,(r,n)\}$ and $\mathbf{R}^{m*n}$ as $\{(1,1),(1,2),..., (m-1,n),(m,n)\}$. The broker computes a proof π_S as follows:

$$\pi_S = \prod_{(i,j)\in I_S} \prod_{(i',j')\in \mathbf{R}^{m*n}\setminus I_S} EK_{\mathcal{S}_{(i,j)(i',j')}}^{K_{i',j'}} \tag{10}$$

The broker uploads $(\pi_V, \pi_{\mathcal{Q}})$ or π_S to the accountability contract by calling *UChallengeResolve* or *RChallengeResolve* functions. Both functions retrieve unprocessed challenges from the blockchain storage, and check the correctness and freshness of the proofs. If the proof is correct, the deposits from users/retailers will be transferred to the broker; Otherwise, users/retailers will take deposits from the broker.

Claim. Users/retailers can use *UserClaim* or *RetailerClaim* to claim broker deposits, if they do not receive a timely response. The functions check the threshold time T_H and the processing status $flag$, and transfer the broker deposits to users/retailers if a response delay is identified.

VI. Security Analysis

First, we summarize the security properties of the *SNARG* systems in terms of *Completeness*, *Soundness*, *Succinctness* and *Privacy Preservation*. Then, we discuss the security and fairness of the accountability contract. Finally, based on the security properties of the *SNARG* systems and the smart contract, we give the security analysis of *privacy-preserving accountability*.

A. SNARG Security

The security properties of $\sum_{\mathcal{Q}}$ inherit from the QAP-based *SNARG* systems [27], [29]. *Completeness* is recognized by the *QAP* theorem and the correctness of the linear combination checks in the bilinear groups. Thus, a rational verifier will accept the proof if it is correctly generated. *Soundness* ensures the unforgeability of the proof $\pi_{\mathcal{Q}}$. If *q-PDH,d-PKE,2q-SDH* assumptions hold in elliptic curve-based groups of order q for a *QAP* of degree d where $q \geq 4d+4$ [29], a computationally-bonded broker without the knowledge of the trapdoor secret cannot forge a valid tuple $(S_u, S_{\mathcal{R}}, S_o) \notin \mathcal{R}_{\mathcal{Q}}$ to pass the $Verify$ function. The proof is *succinct* (a few group elements) due to the high expressiveness of $EK_{\mathcal{Q}}$. Users choose random numbers to construct S_u in the registration phase, which makes the corresponding commitments privacy-preserving even for two commitments with the same keyword set. Since only commitments of S_u is utilized in the $Verify$ function, $\sum_{\mathcal{Q}}$ achieves *privacy preservation* for the user profiles. $\sum_{\mathcal{V}}$ is *succinct* since there are only 2 group elements in the proof $\pi_{\mathcal{V}}$. $\sum_{\mathcal{V}}$ is complete and sound if the *SXDH* assumption holds in $\mathbb{G}$ [37]. $\sum_{\mathcal{S}}$ is *succinct* since there is only one group element in the proof $\pi_{\mathcal{S}}$. $\sum_{\mathcal{S}}$ is *complete* and *sound* if *CDH* assumption holds for computationally-bounded adversaries [36]. $\sum_{\mathcal{S}}$ is not *privacy preserving* as the prover directly opens the S_r to the verifier. This design is reasonable since $\sum_{\mathcal{S}}$ is used for retailer challenge, where the targeting policy S_r is required to be transparent to the public.

Algorithm 2: Accountability Contract

```
Require: $addr_{\mathcal{A}}, pk_{\mathcal{A}}, pp, VK_{\mathcal{Q}}, VK_{\mathcal{V}}, D_{\mathcal{R}}, T_H, C_{\mathcal{A}}$
Set $Rec_U, Rec_R$ to be $empty$
Function UserChallenge $Evid_{sid}$, deposit $C_u$
    Check $Veri(m_{sid}, \pi_{sid})_{pk_{\mathcal{A}}} \rightarrow 1$
    Check $Veri(m'_{sid}, \pi'_{sid})_{pk_{\mathcal{A}}} \rightarrow 1$
    Check $ID_u||sid \not\subseteq Rec_U$
    Set $addr_u = message\ sender,\ \ flag = 0$
    Set $T_{recv} = block.time$
    Add the following to $Rec_U$:
    $(ID_u||sid, addr_u, Evid_{sid}, T_{recv}, flag)$
Function RetailerChallenge $Evid_r$, deposit $C_r$
    Check $Veri(m_r, \pi_r)_{pk_{\mathcal{A}}} \rightarrow 1$
    Check $ID_r \not\subseteq Rec_R$
    Set $addr_r = message\ sender, flag = 0$
    Set $T_{recv} = block.time$
    Add $(ID_r, addr_r, Evid_r, T_{recv}, flag)$ to $Rec_R$
Function UserClaim $ID_u, sid$
    Retrieve tuples from $Rec_U$ by $ID_u||sid$
    Check $addr_u = message\ sender, flag = 0$
    if $block.time - T_{recv} > T_H$ then
        Transfer $C'_u$ to $addr_u$
Function RetailerClaim $ID_r$
    Retrieve tuples from $Rec_R$ by $ID_r$
    Check $addr_r = message\ sender, flag = 0$
    if $block.time - T_{recv} > T_H$ then
        Transfer $C'_r$ to $addr_r$
Function UChalResolve $ID_u, sid, \pi_{\mathcal{V}}, \pi_{\mathcal{Q}}$
    Check $message\ sender = addr_{\mathcal{A}}$
    Retrieve $Evid_{sid}, flag, T_{recv}$ from $Rec_U$
    Check $flag = 0$ and $block.time - T_{recv} < T_H$
    Check $(D_{\bar{uo}} \in Evid_{sid}) = (D_{\bar{uo}} \in \pi_{\mathcal{Q}})$
    Check $Verify(VK_{\mathcal{Q}}, \pi_{\mathcal{Q}}) \rightarrow 1$
    Check $e(X, \tilde{g}) = e(D_{\mathcal{R}}, J)e(Y, K)e(D_{\bar{\mathcal{R}}}, L)$
    if All checks pass then
        Transfer $C_u$ to $addr_{\mathcal{A}}$, set $flag = 1$
    else
        Transfer $C'_u$ to $addr_u$, set $flag = 1$
Fnnction RChalResolve $ID_r, \pi_S$
    Check $message\ sender = addr_{\mathcal{A}}$
    Retrieve $Evid_r, flag, T_{recv}$ from $Rec_R$
    Check $flag = 0$ and $block.time - T_{recv} < T_H$
    Check $e(\frac{D_{\mathcal{R}}}{\prod_{j\in[n]} CK_{\mathcal{R}_{r,j}}^{K_{r,j}}}, \prod_{j\in[n]} \widetilde{CK}_{\mathcal{R}_{r,j}}) = e(\pi_S, \tilde{g})$
    if All checks pass then
        Transfer $C_r$ to $addr_{\mathcal{A}}$, set $flag = 1$
    else
    Transfer $C'_r$ to $addr_r$, set $flag = 1$
```

B. Smart Contract Security

Smart contract security relies on the underlying Ehtereum blockchain. For a proof-of-work consensus protocol, the smart contract security is achieved if an adversary cannot control the most (51 percent) of the computing power in the blockchain network [16]. Informally, the smart contract in the Ethereum provides three security properties: (1) Data stored in the contract cannot be maliciously modified. (2) In a long term, honest blockchain nodes will agree on a consistent view of the smart contract states. (3) A correct contract function call will be verified and executed within a certain period of time, i.e. transaction confirmation time.

C. Privacy-preserving Accountability

1) Public Verifiability: The public verifiability [41] in *SANPA* consists of two parts: retailer policy management and the ad dissemination process. *SANPA* requires a trusted setup of the *SNARG* systems and the accountability contract, that stores verification keys of the *SNARG* systems and the cryptographic commitment of the $D_{\mathcal{R}}$ to receive challenges from users and retailers. Due to the first property of the smart contract, the on-chain storage is secure and cannot be maliciously modified.

For the retailer challenge, the broker generates the evidence $Evid_r$ for each retailer policy S_r. $Evid_r$ is non-repudiable due to the security of *ECDSA* signature. Any retailer can require the broker to generate a proof by sending a retailer challenge to the accountability contract. The broker proves that S_r is correctly included in $D_{\mathcal{R}}$ by running *Prove* function of $\sum_{\mathcal{S}}$, which cannot be forged due to the *soundness* of $\sum_{\mathcal{S}}$. For the user challenge, the broker generates the evidence $Evid_{sid}$ for each ad dissemination. The evidence is non-repudiable due to the security of *ECDSA* signature. When receiving a user challenge, the broker proves that the ad dissemination process is correctly conducted with $D_{\bar{uo}}$ in $Evid_{sid}$ and the on-chain authenticator $D_{\mathcal{R}}$. To do so, the broker runs *Prove* functions of $\sum_{\mathcal{Q}}$ and $\sum_{\mathcal{V}}$. Due to the *soundness* of $\sum_{\mathcal{Q}}$ and $\sum_{\mathcal{V}}$, the broker cannot forge an invalid proof that passes $Verify$ functions in the *UChalResolve*.

With the *completeness* of the *SNARG* systems, a rational user or retailer will accept the proof if it is correctly computed. Since the accountability contract is implemented over the Ethereum blockchian, anyone in the public can verify correctness of the transparency challenges. In summary, *public verifiability* is achieved in *SANPA*.

2) Privacy Preservation: For the user challenge, users upload the evidence $Evid_{sid}$ to the accountability contract. $Evid_{sid}$ contains the commitment $D_{\bar{uo}}$ of S_u, S_o. The accountability contract uses $Verify$ functions of $\sum_{\mathcal{Q}}$ and $\sum_{\mathcal{V}}$ for public verifications of the proof $\pi_{\mathcal{Q}}$ and $\pi_{\mathcal{V}}$. Due to the randomness in the user profile, the verifications on the contract will not leak the plaintext of $D_{\bar{uo}}$, other than if $(S_u, S_{\mathcal{R}}, S_o) \in \mathcal{R}_{\mathcal{Q}}$. That is, *privacy preservation* for the user profile is achieved in *SANPA*.

3) Obligation Enforcement: *SANPA* utilizes the smart contract to enforce timely obligations on misbehaving parties. Specifically, the accountability contract takes initial deposits from the broker and will transfer the deposits if any advertising misconduct is identified. With the *liveness* of the smart contract, an advertising challenge cannot be maliciously delayed or ignored. Moreover, to promote timely responses from the broker, the accountability contract utilizes the Ethereum block time and sets a threshold T_H for each advertising challenge. That is, obligation enforcement is achieved in *SANPA*.

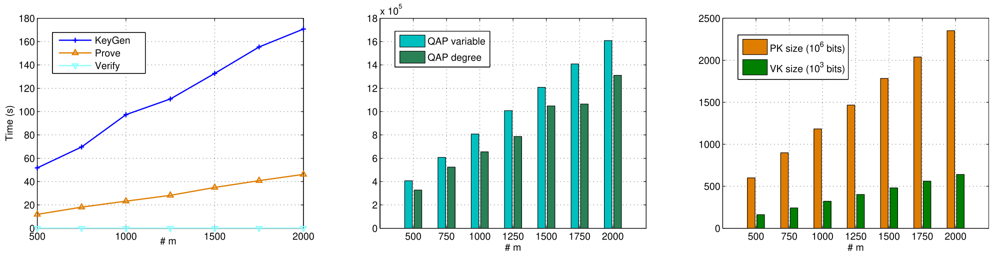
VII. Performance Evaluation

Since *SANPA* is an add-on component of the existing *SAN*, we mainly evaluate the additional overhead of *SANPA* in terms of

TABLE II
SNARG COMPLEXITY

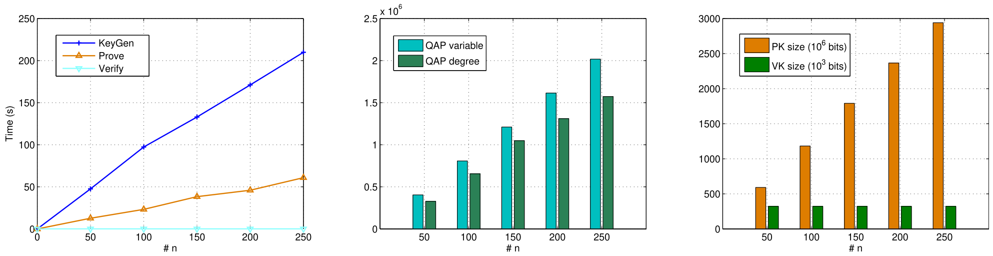
	EK	VK	Proof	KeyGen	Prove	Verify
$\sum_{\mathcal{V}}$	$2mn\lvert\mathbb{G}_1\rvert$	$3\lvert\mathbb{G}_2\rvert$	$2\lvert\mathbb{G}_1\rvert$	$4mnE_1+3E_2$	$2mnE_1$	$4P$
$\sum_{\mathcal{S}}$	$(m^2n^2-mn)\mathbb{G}_1$	$n(\mathbb{G}_1+\mathbb{G}_2)$	$\mathbb{G}_1$	$(m^2n^2-mn)E_1+mnE_2$	$n(mn-n)E_1$	nE_1+2P

* $|\mathbb{G}_1|, |\mathbb{G}_2|, |\mathbb{Z}_P|$, size of a group element in $\mathbb{G}_1, \mathbb{G}_2, \mathbb{Z}_P$; E_1, E_2, exponentiation operations in $\mathbb{G}_1, \mathbb{G}_2$; P, paring operation in $\mathbb{G}$; m, number of retailers; n, dimension of the keyword dictionary.



(a) Computation (b) QAP Complexity (c) Key Size

Fig. 3. Overheads vs $m, n = 100, k = m$.



(a) Computation (b) QAP Complexity (c) Key Size

Fig. 4. Overheads vs $n, m = k = 1000$.

the *SNARG* systems and the accountability contract. We utilize a single entity to implement DC in the experiments for illustrative purposes. First, we present the theoretical and experimental analysis of *SNARG* systems $\sum_{\mathcal{Q}}$, $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$. Second, we analyze the accountability contract in terms of the on-chain gas cost. All experiments are conducted on a Linux System with 2.4 GHz processor and 8 GB memory.

A. SNARG *Systems*

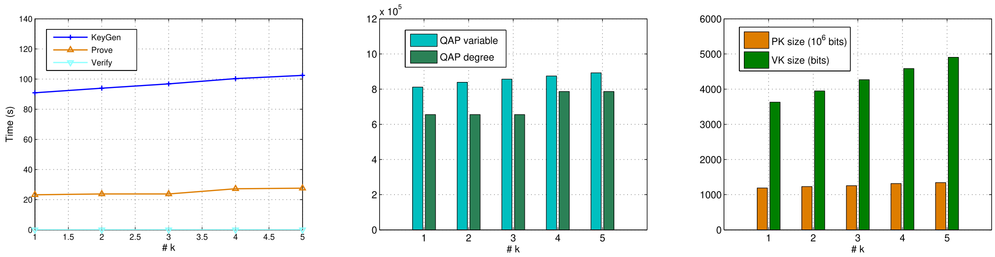
1) $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ Complexity: $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ are instantiated based on *alt-bn128* curve in the *libff* library [42]. In Table II, we summarize the storage and computation complexity of $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ with the following observations:

- Storage cost of proof and computation cost of $Verify$ remain *succinct* regardless of the input size m. This is critical since the verification is conducted on the accountability contract with expensive on-chain computation and storage costs.
- $Prove$ and $KeyGen$ consists of multi exponentiations E_1 and E_2, which are optimized with a table of intermediate powers.

Therefore, $\sum_{\mathcal{V}}$ and $\sum_{\mathcal{S}}$ are practical for both the on-chain implementations of the $Verify$ function and off-chain implementations of $Prove$ and $KeyGen$ at the broker with powerful computing resources.

2) $\sum_{\mathcal{Q}}$ Complexity: We write the ad dissemination process of Algorithm 1 in C. The python interface of Pinocchio is adopted to translate the C codes to circuits. We re-compile the circuit-to-SNARG interface [43]–[45] in *libsnark*, by instantiating the *R1CS ppzkSNARK* with *alt-bn128* curve, which is supported by pairing operations in the Ethereum [16]. We choose three tunable system parameters: the dimension of the keyword dictionary n, the number of retailers m, and the number of returned results k. We also identify three sets of performance indicators. First, we measure the time costs for the three functions: $KeyGen$, $Prove$ and $Verify$. Second, we measure the *QAP* complexity by the number of variables and degrees in the compiled *QAP* program. Third, we measure the size of *PK* and *VK* in bits.

In Fig. 3(a), 4(a) and 5(a), the time cost for $Prove$ and $KeyGen$ increases with m, n, k. At the same time, the time cost for $Verify$ remains the same: around 0.27s in all experiments. This is because $\sum_{\mathcal{Q}}$ achieves *succinct* verification cost with only 12 pairings. Although $Keygen$ operation is much more expensive compared with $Prove$ and $Verify$, it is acceptable since it is an one-time setup and the entity can remain offline after the setup. Similar properties for the *QAP* complexity are found in Fig. 3(b), 4(b) and 5(b). In Fig. 3(c), 4(c) and 5(c), the *PK* size is much larger than the *VK* size. Since the original S_o instead of its commitment is used in the experiment, the *VK* size



(a) Computation (b) QAP Complexity (c) Key Size

Fig. 5. Overheads vs $k, m = 1000, n = 100$.

TABLE III
FUNCTION COMPLEXITY & GAS COST

Function	Storage	Computation	Approximate Gas Cost
UserChallenge	$2\pi_E + 4 * W_{32} + \lvert\mathbb{G}_1\rvert$	$2ER$	20,000
RetailerChallenge	$\pi_E + 2 * W_{32} + n(\lvert\mathbb{G}_1\rvert + \lvert\mathbb{G}_2\rvert + \lvert\mathbb{Z}_P\rvert)$	ER	14,000 n +9,000
UChalResolve	$2 * W_{32} + 13\lvert\mathbb{G}_1\rvert + \lvert\mathbb{G}_2\rvert$	18P	719,000
RChalResolve	$W_{32} + \lvert\mathbb{G}_1\rvert$	$2P + nE_1$	6,000n +119,000

* $|\mathbb{G}_1|, |\mathbb{G}_2|, |\mathbb{Z}_P|$, size of a group element in $\mathbb{G}_1, \mathbb{G}_2, \mathbb{Z}_P$; E_1, exponentiation operation in $\mathbb{G}_1$; P, paring operation in $\mathbb{G}$; π_E, *ECDSA* signature; ER, *ECDSA* verification operation; W_i, an i-bit word; n, dimension of the keyword dictionary.

linearly grows with the number of outputs S_o. The *VK* size remains the same in Fig. 4(c) as the number of outputs is fixed at 1000. In Fig. 5(c), the *VK* size is reduced to a few thousand bits if $\sum_Q$ only outputs a few results.

B. Accountability Contract

We mainly estimate the gas cost of storing data and conducting cryptographic operations since algebraic operations and read/store operations are negligible [46] in the contract. For example, it costs 2,000 gas to store a 256-bit word in the contract storage and 3,000 gas to verify an *ECDSA* signature. For *alt-bn128* group operations, we take the estimations of optimized precompiled contracts [47].

In Table III, we summarize the storage, computation, and gas cost for the four complex functions of the accountability contract. For the estimation, we take the theoretical results of *alt-bn128* curve, where $|\mathbb{Z}_p|$ is 256 bit, $|\mathbb{G}_1|$ is 512 bit, and $|\mathbb{G}_2|$ is 1024 bit. We regard the user/retailer ID and the time stamp as a 32-bit word. The combination of $\sum_Q$ and $\sum_\mathcal{V}$ increases the size and verification cost of π_Q compared with the adopted *libsnark* implementations in our experiments. Theoretically, two additional elements $D_{\bar{\mathcal{R}}}, D_{\bar{u}\bar{o}} \in \mathbb{G}_1$ are introduced. Two more elements in $\mathbb{G}_1$ for appropriate span checks of $D_{\bar{\mathcal{R}}}, D_{\bar{u}\bar{o}}$ are also added. Two more elements in $\mathbb{G}_1$ and two more pairings are also introduced to check $D_{\bar{\mathcal{R}}}, D_{\bar{u}\bar{o}}$ are well formed. In Table III, the retailer challenge is more expensive than the user challenge. As a result, we can increase the amount of broker deposits, such that the broker loses more cryptocurrencies if the misconduct of retailer policy management is identified.

We summarize some insights into the experimental results. (1) Linear off-chain complexity at the prover is mainly caused by the arithmetic computations. Specifically, dynamic subscript assignment for arrays and loop breaks are not supported in the implementation. (2) The efficiency at the on-chain verifier is achieved due to the efficient verifications of the *QAP* theorem, which significantly reduces the implementation cost as on-chain storage and computation are expensive.

VIII. CONCLUSION

In this paper, we have proposed a blockchain-based smart advertising network with privacy-preserving accountability (*SANPA*). *SANPA* can increase the public awareness of the advertising transparency with privacy-preserving accountability enforcements on advertising misconduct. We have conducted extensive experiments to demonstrate that *SANPA* is feasible for real-world implementations. The research may shed light on the future research and practice of a more trustworthy advertising network. For our future work, we will investigate the smart advertising network under recent privacy regulations.

REFERENCES

[1] J. Chen, K. Hu, Q. Wang, Y. Sun, Z. Shi, and S. He, "Narrowband internet of things: Implementations and applications," *IEEE Internet of Things J.*, vol. 4, no. 6, pp. 2309–2314, Dec. 2017.

[2] L. Zhu, J. Zhang, Z. Xiao, X. Cao, D. O. Wu, and X.-G. Xia, "Joint tx-rx beamforming and power allocation for 5g millimeter-wave non-orthogonal multiple access networks," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 5114–5125, Jul. 2019.

[3] Advertising Industry Statistics. Accessed Mar. 2020. [Online]. Available: https://www.emarketer.com/content/mobile-advertising-is-expected-to-surpass-tv-ad-spending

[4] A. Andreou, M. Silva, F. Benevenuto, O. Goga, P. Loiseau, and A. Mislove, "Measuring the facebook advertising ecosystem," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2019.

[5] A. Andreou, G. Venkatadri, O. Goga, K. Gummadi, P. Loiseau, and A. Mislove, "Investigating ad transparency mechanisms in social media: A case study of facebook's explanations," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2018.

[6] J. Parra-Arnau, J. P. Achara, and C. Castelluccia, "Myadchoices: Bringing transparency and control to online advertising," *ACM Trans. Web*, vol. 11, no. 1, pp. 1–47, 2017.

[7] M. Ali, P. Sapiezynski, M. Bogen, A. Korolova, A. Mislove, and A. Rieke, "Discrimination through optimization: How facebook's ad delivery can lead to skewed outcomes," 2019, *arXiv:1904.02095*.

[8] J. Gui, M. Nagappan, and W. G. Halfond, "What aspects of mobile ads do users care about? an empirical study of mobile in-app ad reviews," 2017, *arXiv:1702.07681*.

[9] W. Li, H. Li, H. Chen, and Y. Xia, "Adattester: Secure online mobile advertisement attestation using trustzone," in *Prof. MobiSys*., 2015, pp. 75–88.

[10] G. Venkatadri, A. Mislove, and K. P. Gummadi, "Treads: Transparency-enhancing ads," in *Proc. ACM Workshop Hot Topics Netw.*, 2018, pp. 169–175.

[11] Google, Microsoft and Amazon pay to get around ad blocking tool. Accessed Mar. 2020. [Online]. Available: https://www.ft.com/content/80a8ce54-a61d-11e4-9bd3-00144feab7de

[12] D. Liu, A. Alahmadi, J. Ni, X. Lin, and X. Shen, "Anonymous reputation system for iiot-enabled retail marketing atop pos blockchain," *IEEE Trans. Ind. Inform.*, vol. 15, no. 6, pp. 3527–3537, Jun. 2019.

[13] H. Liu and J. Chen, "Distributed privacy-aware fast selection algorithm for large-scale data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 2, pp. 365–376, Feb. 2017.

[14] J. Misic, V. B. Misic, X. Chang, S. G. Motlagh, and Z. M. Ali, "Modeling of bitcoin's blockchain delivery network," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 3, pp. 1368-1381, Jul.-Sep. 2020, doi: 10.1109/TNSE.2019.2928716.

[15] Y. Xu, C. Zhang, Q. Zeng, G. Wang, J. Ren, and Y. Zhang, "Blockchain-enabled accountability mechanism against information leakage in vertical industry services," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1202–1213, Apr.-Jun. 2021.

[16] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger byzantium version," *Ethereum Project Yellow Paper*, pp. 1–39, 5, Jun. 2018.

[17] M. Li, J. Weng, A. Yang, J.-n. Liu, and X. Lin, "Towards blockchain-based fair and anonymous ad dissemination in vehicular networks," *IEEE. Trans. Veh. Technol.*, vol. 68, no. 11, pp. 11248–11259, Nov. 2019.

[18] J. Eberhardt and S. Tai, "Zokrates-scalable privacy-preserving off-chain computations," in *Proc. IEEE Int. Conf. Blockchain*, 2018, pp. 1084-1091.

[19] D. Liu, J. Ni, C. Huang, X. Lin, and X. Shen, "Secure and efficient distributed network provenance for IoT: A blockchain-based approach," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7564-7574, Aug. 2020, doi: 10.1109/JIOT.2020.2988481.

[20] W. Tang, J. Ren, and Y. Zhang, "Enabling trusted and privacy-preserving healthcare services in social media health networks," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 579–590, Mar. 2019.

[21] J. Frankle, S. Park, D. Shaar, S. Goldwasser, and D. Weitzner, "Practical accountability of secret processes," in *Proc. USENIX Secur.*, 2018, pp. 657–674.

[22] G. Panwar, R. Vishwanathan, S. Misra, and A. Bos, "Sampl: Scalable auditability of monitoring processes using public ledgers," in *Proc. ACM CCS*, 2019, pp. 2249–2266.

[23] M. Li, J. Weng, J.-N. Liu, X. Lin, and C. Obimbo, "BB-VDF: Enabling accountability and fine-grained access control for vehicular digital forensics through blockchain," Cryptology ePrint Archive, Report 2020/011, 2020. [Online]. Available: https://eprint.iacr.org/2020/011/20200106:083447

[24] R. Neisse, G. Steri, and I. Nai-Fovino, "A blockchain-based approach for data accountability and provenance tracking," in *Proc. Int. Conf. Availability, Rel. Secur.*, 2017, pp. 1–14.

[25] K. Nguyen, G. Ghinita, M. Naveed, and C. Shahabi, "A privacy-preserving, accountable and spam-resilient geo-marketplace," in *Proc. Int. Conf. Advances Geographic Inf. Syst. ACM SIGSPATIAL*, 2019, pp. 299–308.

[26] R. Gennaro, C. Gentry, B. Parno, and M. Raykova, "Quadratic span programs and succinct nizks without pcps," in *Proc. EUROCRYPT*, 2013, pp. 626–645.

[27] B. Parno, J. Howell, C. Gentry, and M. Raykova, "Pinocchio: Nearly practical verifiable computation," in *Proc. IEEE S&P*, 2013, pp. 238–252.

[28] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Succinct non-interactive zero knowledge for a von Neumann architecture," in *Proc. USENIX Secur.*, 2014, pp. 781–796.

[29] J. Groth, "On the size of pairing-based non-interactive arguments," in *Proc. EUROCRYPT*, 2016, pp. 305–326.

[30] S. Agrawal, C. Ganesh, and P. Mohassel, "Non-interactive zero-knowledge proofs for composite statements," in *Proc. CRYPTO*, 2018, pp. 643–673.

[31] M. Campanelli, D. Fiore, and A. Querol, "Legosnark: Modular design and composition of succinct zero-knowledge proofs," in *Proc. ACM CCS*, 2019.

[32] J. Jiang, Y. Zheng, Z. Shi, X. Yuan, X. Gui, and C. Wang, "A practical system for privacy-aware targeted mobile advertising services," *IEEE Trans. Serv. Comput.*, vol. 13, no. 3, pp. 410-424, 1 May-Jun. 2020, doi: 10.1109/TSC.2017.2697385.

[33] J. Ni, K. Zhang, Q. Xia, X. Lin, and X. Shen, "Enabling strong privacy preservation and accurate task allocation for mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 6, pp. 1317-1331, 1 Jun. 2020, doi: 10.1109/TMC.2019.2908638.

[34] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proc. Annu. Int. Cryptology Conf.*, 1991, pp. 129–140.

[35] J. Camenisch, M. Drijvers, and M. Dubovitskaya, "Practical uc-secure delegatable credentials with attributes and their application to blockchain," in *Proc. ACM CCS*, 2017, pp. 683–699.

[36] R. W. Lai and G. Malavolta, "Subvector commitments with application to succinct arguments," in *Proc. CRYPTO*, 2019, pp. 530–560.

[37] D. Fiore, C. Fournet, E. Ghosh, M. Kohlweiss, O. Ohrimenko, and B. Parno, "Hash first, argue later: Adaptive verifiable computations on outsourced data," in *Proc. ACM CCS*, 2016, pp. 1304–1316.

[38] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ecdsa)," *Int. J. Inf. Secur.*, vol. 1, no. 1, pp. 36–63, 2001.

[39] S. Bowe, A. Gabizon, and M. D. Green, "A multi-party protocol for constructing the public parameters of the Pinocchio zk-SNARK," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, 2018, pp. 64–77.

[40] D. Chen, N. Zhang, R. Lu, N. Cheng, K. Zhang, and Z. Qin, "Channel precoding based message authentication in wireless networks: Challenges and solutions," *IEEE Netw.*, vol. 33, no. 1, pp. 99–105, Jan.-Feb. 2018.

[41] J. Shao, R. Lu, Y. Guan, and G. Wei, "Achieve efficient and verifiable conjunctive and fuzzy queries over encrypted data in cloud," *IEEE Trans. Serv. Comput.*, to be published, doi: 10.1109/TSC.2019.2924372.

[42] C++ library for Finite Fields and Elliptic Curves. Accessed Jan. 2020. [Online]. Available: https://github.com/scipr-lab/libff

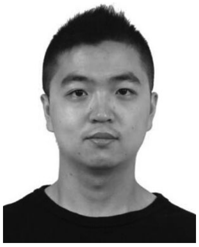
[43] libsnark: A C++ library for zkSNARK proofs. Accessed Jan. 2020. [Online]. Available: https://github.com/scipr-lab/libsnark

[44] JSnark. Accessed Jan. 2020. [Online]. Available: https://github.com/akosba/jsnark

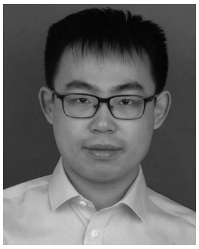
[45] A. Kosba, C. Papamanthou, and E. Shi, "xjsnark: a framework for efficient verifiable computation," in *Proc. IEEE S&P*, 2018, pp. 944–961.

[46] Ethereum Gas Table. Accessed Mar. 2020. [Online]. Available: https://ethgastable.info

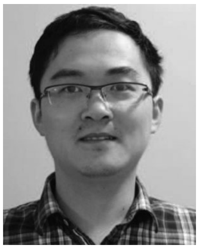
[47] EIP 1108. Reduce alt-bn128 precompile gas costs. Accessed Feb. 2020. [Online]. Available: https://eips.ethereum.org/EIPS/eip-1108



Dongxiao Liu (Member, IEEE) received the B.S. and M.S. degrees from the School of Computer Science and Engineering, University of Electronic Science and Technology of China, China in 2013 and 2016, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Canada, in 2020. His research interests include applied cryptography and privacy enhancing technologies for blockchain.



Cheng Huang (Member, IEEE) received the B.Eng. and M.Eng. degrees in information security from Xidian University, China, in 2013 and 2016, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, ON, Canada, in 2020. He was a Project Officer with the INFINITUS Laboratory, School of Electrical and Electronic Engineering, Nanyang Technological University, till July 2016. His research interests include applied cryptography, cyber security, and privacy in the mobile network.



Jianbing Ni (Member, IEEE) received the B.E. and M.S. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2011 and 2014, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, Canada, in 2018. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON, Canada. His current research interests include applied cryptography and network security, with a focus on cloud computing, smart grid, mobile crowdsensing, and the Internet of Things.

Xiaodong Lin (Fellow, IEEE) received the Ph.D. degree in information engineering from the Beijing University of Posts and Telecommunications, China, and the Ph.D. degree (with Outstanding Achievement in Graduate Studies Award) in electrical and computer engineering from the University of Waterloo, Canada. He is currently an Associate Professor with the School of Computer Science, University of Guelph, Canada. His research interests include computer and network security, privacy protection, applied cryptography, computer forensics, and software security.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular ad hoc and sensor networks. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.

He was the recipient of the R.A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and the Education Award in 2017 from the IEEE Communications Society, and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He was also the recipient of the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He was the Technical Program Committee Chair/Co-Chair for the IEEE Globecom'16, the IEEE Infocom'14, the IEEE VTC'10 Fall, the IEEE Globecom'07, the Symposia Chair for the IEEE ICC'10, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the elected IEEE Communications Society Vice President for Technical & Educational Activities, Vice President for Publications, Member-at-Large on the Board of Governors, Chair of the Distinguished Lecturer Selection Committee, Member of IEEE Fellow Selection Committee. He was/is the Editor-in-Chief of the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, *IET Communications*, and *Peer-to-Peer Networking and Applications*.