An Efficient Technique for Protecting Location Privacy of Cooperative Spectrum Sensing Users

Mohamed Grissa[⋆], Attila Yavuz[⋆], and Bechir Hamdaoui[⋆]
[⋆]Oregon State University, grissam,yavuza,hamdaoub@onid.oregonstate.edu

***Abstract*—Cooperative spectrum sensing, despite its effectiveness in enabling dynamic spectrum access, suffers from location privacy threats, merely because Secondary Users (SUs)' sensing reports that need to be shared with a fusion center to make spectrum availability decisions are highly correlated to the users' locations. It is therefore important that cooperative spectrum sensing schemes be empowered with privacy preserving capabilities so as to provide SUs with incentives for participating in the sensing task. In this paper, we propose an efficient privacy preserving protocol that uses an additional architectural entity and makes use of various cryptographic mechanisms to preserve the location privacy of SUs while performing reliable and efficient spectrum sensing. We show that not only is our proposed scheme secure and more efficient than existing alternatives, but also achieves fault tolerance and is robust against sporadic network topological changes.**

***Index Terms*—Location privacy, secure cooperative spectrum sensing, order preserving encryption, cognitive radio networks.**

I. Introduction

Cooperative spectrum sensing is a key component of cognitive radio networks (CRNs) essential for enabling dynamic and opportunistic spectrum access [1]. It consists of having secondary users (SUs) sense the licensed channels on a regular basis and collaboratively decide whether a channel is available prior to using it so as to avoid harming primary users (PUs). One of the most popular spectrum sensing techniques is energy detection, thanks to its simplicity and ease of implementation, which essentially detects the presence of PU signal by measuring and relying on the energy strength of the sensed signal, commonly known as the received signal strength (RSS) [2].

Broadly speaking, cooperative spectrum sensing techniques can be classified into two categories: Centralized and distributed [1]. In centralized techniques, a central entity called fusion center (FC) orchestrates the sensing operations as follows. It selects one channel for sensing and, through a control channel, requests that each SU perform local sensing on that channel and send its sensing report (e.g., the observed RSS value) back it to. It then combines the received sensing reports, makes a decision about the channel availability, and diffuses the decision back to the SUs. In distributed sensing techniques, SUs do not rely on a FC for making channel availability decisions. They instead exchange sensing information among one another to come to a unified decision [1]. This requirement makes distributed sensing techniques highly complex with respect to their centralized counterparts. Hence, centralized sensing techniques are considered more practical for real-life applications.

Despite its usefulness and effectiveness in promoting dynamic spectrum access, cooperative spectrum sensing suffers from serious security and privacy threats. One big threat to SUs, which we tackle in this work, is location privacy, which can easily be leaked due to the wireless nature of the signals communicated by SUs during the cooperative sensing process. In fact, it has been shown that RSS values of SUs are highly correlated to their physical locations [3], thus making it easy to compromise the location privacy of SUs when sending out their sensing reports. The fine-grained location, when combined with other publicly available information, could easily be exploited to infer private information about users [4]. Examples of such private information are shopping patterns, user preferences, and user beliefs, just to name a few [4]. With such privacy threats and concerns, SUs may refuse to participate in the cooperative sensing tasks. It is therefore imperative that cooperative sensing schemes be enabled with privacy preserving capabilities that protect the location privacy of SUs, thereby encouraging them to participate in such a key CRN function, the spectrum sensing.

In this paper, we propose an efficient privacy-preserving scheme for cooperative spectrum sensing that exploits various cryptographic mechanisms to preserve the location privacy of SUs while performing the cooperative sensing task reliably and efficiently. We show that our proposed scheme is secure and more efficient than its existing counterparts, and is robust against sporadic topological changes and network dynamism.

A. Related Work

Security and privacy in CRNs have gained some attention recently. Yan et al. [5] discussed security issues in fully distributed cooperative sensing. Qin et al. [6] proposed a privacy-preserving protocol for CRN transactions using a commitment scheme and zero-knowledge proof. Wang et al. [7] proposed a privacy preserving framework, *PrimCos*, for collaborative spectrum sensing in the context of multiple service providers.

Location privacy, though well studied in the context of location-based services (LBS) [8], [9], has received little attention in the context of CRNs [3], [10]–[12]. Some works focused on location privacy but not in the context of cooperative spectrum sensing (e.g., database-driven spectrum sensing [10], [11] and dynamic spectrum auction [12]) and are skipped here since they are not within this paper's scope.

In the context of cooperative spectrum sensing, Shuai et al. [3] showed that SUs' locations can easily be inferred from their RSS reports, and called this the SRLP (single report location privacy) attack. They also identified the DLP (differential location privacy) attack, where a malicious entity can estimate the RSS (and hence the location) of a leaving/joining user from the variations in the final aggregated

RSS measurements before and after user's joining/leaving of the network. They finally proposed $PPSS$, a protocol for cooperative spectrum sensing, to address these two attacks. Despite its merits, $PPSS$ has several limitations: (i) It needs to collect all the sensing reports to decode the aggregated result. This is not fault tolerant, since some reports may be missing due, for e.g., to the unreliable nature of wireless channels. (ii) It cannot handle dynamism if multiple users join or leave the network simultaneously. (iii) The pairwise secret sharing requirement incurs extra communication overhead and delay. (iv) The underlying encryption scheme requires solving the *Discrete Logarithm Problem* [13], which is possible only for very small plaintext space and can be extremely costly (see Table I). Chen et al. [14] proposed $PDAFT$, a fault-tolerant and privacy-preserving data aggregation scheme for smart grid communications.

$PDAFT$, though proposed in the context of smart grids, is suitable for cooperative sensing schemes. But unlike $PPSS$, $PDAFT$ relies on an additional semi-trusted entity, called gateway, and like other aggregation based methods, is prone to the DLP attack. In our previous work [15] we proposed an efficient scheme called $LPOS$ to overcome the limitations that existent approaches suffer from. $LPOS$ combines order preserving encryption and yao's millionaire protocol to provide a high location privacy to the users while enabling an efficient sensing performance.

B. Our Contribution

In this paper, we propose a new location privacy-preserving scheme that we call *LP-3PSS* (location privacy for 3-party spectrum sensing architecture), which harnesses various cryptographic primitives (e.g., order preserving encryption) in innovative ways along with an additional architectural entity (i.e., a gateway) to achieve high location privacy with a low overhead. That is, our proposed *LP-3PSS* scheme offers the following desirable properties:

- Location privacy of secondary users while performing the cooperative spectrum sensing effectively and reliably.
- Fault tolerance and robustness against network dynamism (e.g., multiple SUs join/leave the network) and failures (e.g., missed sensing reports).
- Reliability and resiliency against malicious users via an efficient reputation mechanism.
- Accurate spectrum availability decisions via half-voting rules while incurring minimum communication and computation overhead.

Note that for simplicity we use energy detection through RSS measurement for spectrum sensing in our scheme. However, our scheme can be applied with any other spectrum detection technique whose sensing reports may leak information about the location of the users.

II. Preliminaries

We consider a cooperative spectrum sensing architecture that consists of a FC and a set of SUs, where each SU is assumed to be capable of measuring RSS on any channel by means of an energy detection method [2]. In this cooperative sensing architecture, the FC combines the sensing observations collected from the SUs, decides about the spectrum availability, and broadcasts the decision back to the SUs through a control channel. This could typically be done via either *hard* or *soft* decision rules. The most common soft decision rule is aggregation, where FC collects the RSS values from the SUs and compares their average to a predefined threshold, τ, to decide on the channel availability.

In hard decision rules, e.g. voting, FC combines votes instead of RSS values. Here, each SU compares its RSS value with τ, makes a local decision (available or not), and then sends to the FC its one-bit local decision/vote instead of sending its RSS value. FC applies then a voting rule on the collected votes to make a channel availability decision. However, for security reasons to be discussed shortly, it may not be desirable to share τ with SUs. In this case, FC can instead collect the RSS values from the SUs, make a vote for each SU separately, and then combine all votes to decide about the availability of the channel.

In this work, we opted for the voting-based decision rule, with τ is not to be shared with the SUs, over the aggregation-based rule. There are two reasons for choosing voting-based decision rule over the aggregation-based decision rule: (i) Aggregation methods are more prone to sensing errors; for example, receiving some erroneous measurements that are far off from the average of the RSS values can skew the computed RSS average, thus leading to wrong decision. (ii) Voting does not expose users to the DLP attack [3] (which is identified earlier in Section I-A). We chose not to share τ with the SUs because doing so limits the action scope of malicious users that may want to report falsified RSS values for malicious and/or selfish purposes.

In this paper we investigate a 3-party cooperative sensing architecture, where a third entity, called gateway (GW), is incorporated along with the FC and SUs to cooperate with them in performing the sensing task. As will be shown later, this additional gateway allows to achieve higher privacy and lesser computational overhead, but of course at its cost.

A. Security Threat Model and Objectives

We consider a *semi-honest* threat model, where all the network parties (i.e., SUs, FC, and GW) are assumed to be *honest but curious* in that they execute the protocol honestly but show interest in learning information about the other parties. This means that none of these entities is trusted. More specifically, we make the following assumptions:

Security Assumption 1. *No party in the system modifies maliciously (or nonmaliciously) the integrity of its input. That is, (i) FC does not maliciously inject false τ; and (ii) the SUs do not maliciously change their RSS values.*

Security Assumption 2. *No party in the system colludes with any of the other parties. That is, (i) FC does not collude with SUs; (ii) SUs do not collude with one another; and (iii) GW does not collude with SUs or FC.*

As mentioned before, RSS values are shown to be highly correlated to the SUs' locations [3]. Therefore, if the confidentiality of the RSS values is not protected, then nor is the location privacy of the SUs. With this in mind, the security objectives of the proposed schemes are then:

Security Objective 1. *Keep the RSS value of each SU confidential to the SU only by hiding it from all other parties. This should hold during all sensing periods and for any network membership change.*

Also, since SUs may rely on the threshold τ to maliciously manipulate their RSSs, our second objective is then to:

Security Objective 2. *Keep τ confidential to the FC only by hiding it from all other parties. This should hold during all sensing periods and for any network membership change.*

B. Half-Voting Availability Decision Rule

Our proposed scheme uses the *half-voting decision rule*, shown to be optimal in [16], and for completeness, we here highlight its main idea. Details can be found in [16].

Let h_0 and h_1 be the spectrum sensing hypothesis that PU is absent and present, respectively. Let P_f, P_d and P_m denote the probabilities of false alarm, detection, and missed detection, respectively, of one SU; i.e., $P_f = Pr(RSS > \tau \mid h_0)$, $P_d = Pr(RSS > \tau \mid h_1)$, and $P_m = 1 - P_d$. FC collects the 1-bit decision D_i from each SU U_i and fuses them together according to the following fusion rule [16]:

$$dec = \begin{cases} \mathcal{H}_1, & \sum_{i=1}^{n} D_i \geq \lambda \\ \mathcal{H}_0, & \sum_{i=1}^{n} D_i < \lambda \end{cases} \tag{1}$$

Note that FC infers that PU is present when at least λ SUs are inferring h_1. Otherwise, FC decides that PU is absent, i.e. $\mathcal{H}_0$. Note here that the OR fusion rule corresponds to the case where $\lambda = 1$ and the AND fusion rule corresponds to the case where $\lambda = n$. The cooperative spectrum sensing false alarm probability, Q_f, and missed detection probability, Q_m, are: $Q_f = Pr(\mathcal{H}_1 \mid h_0)$ and $Q_m = Pr(\mathcal{H}_0 \mid h_1)$. Letting n be the number of SUs, the optimal value of λ that minimizes $Q_f + Q_m$ is $\lambda_{opt} = \min(n, \lceil n/(1+\alpha) \rceil)$, where $\alpha = \ln(\frac{P_f}{1-P_m}) / \ln(\frac{P_m}{1-P_f})$ and $\lceil \cdot \rceil$ denotes the ceiling function. For simplicity, λ_{opt} is denoted as λ throughout this paper.

C. Reputation Mechanism

To make the voting rule more reliable, we incorporate a reputation mechanism that allows FC to progressively eliminate faulty and malicious SUs. It does so by updating and maintaining a reputation score for each SU to reflect the level of reliability the SU has. Our proposed schemes incorporate the *Beta Reputation* mechanism, proposed and shown to be robust by Arshad et al. [17]. For completeness, we highlight its key features next; more details can be found in [17].

At the end of each sensing period t, FC obtains a decision vector, $\boldsymbol{b}(t) = [b_1(t), b_2(t), \ldots, b_n(t)]^T$ with $b_i(t) \in \{0, 1\}$, where $b_i(t) = 0$ (resp. $b_i(t) = 1$) means that the spectrum is reported to be free (resp. busy) by SU U_i. FC then makes a global decision using the fusion rule f as follows:

$$dec(t) = f(\boldsymbol{w}(t), \boldsymbol{b}(t)) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i(t) b_i(t) \geq \lambda \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\boldsymbol{w}(t) = [w_1(t), w_2(t) \ldots, w_n(t)]^T$ is the weight vector calculated by FC based on the credibility score of each user, as will be shown shortly, and λ is the voting threshold determined by the Half-voting rule [16], as presented in Section II-B.

For each SU U_i, FC maintains positive and negative rating coefficients, $\varrho_i(t)$ and $\eta_i(t)$, that are updated every sensing period t as: $\varrho_i(t) = \varrho_i(t-1) + \nu_1(t)$ and $\eta_i(t) = \eta_i(t-1) + \nu_2(t)$, where $\nu_1(t)$ and $\nu_2(t)$ are calculated as

$$\nu_1(t) = \begin{cases} 1 & b_i(t) = dec(t) \\ 0 & \text{otherwise} \end{cases} \qquad \nu_2(t) = \begin{cases} 1 & b_i(t) \neq dec(t) \\ 0 & \text{otherwise} \end{cases}$$

Here, $\varrho_i(t)$ (resp. $\eta_i(t)$) reflects the number of times U_i's observation, $b_i(t)$, agrees (resp. disagrees) with the FC's global decision, dec(t).

FC computes then U_i's credibility score, φ_i(t), and contribution weight, w_i(t), at sensing period t as:

$$\varphi_i(t) = \frac{\varrho_i(t) + 1}{\varrho_i(t) + \eta_i(t) + 2} \quad (3) \qquad w_i(t) = \varphi_i(t) / \sum_{j=1}^{n} \varphi_j(t) \quad (4)$$

D. Cryptographic Building Blocks

Our scheme uses a well known cryptographic building block, which we define next before using it in the next section when describing our scheme so as to ease the presentation.

Definition 1. ***Order Preserving Encryption* (OPE)*:*** *is a deterministic symmetric encryption scheme whose encryption preserves the numerical ordering of the plaintexts, i.e. for any two messages m_1 and m_2 s.t. $m_1 \leq m_2$, we have $c_1 \leftarrow OPE.\mathcal{E}_K(m_1) \leq c_2 \leftarrow OPE.\mathcal{E}_K(m_2)$ [18], with $c \leftarrow OPE.\mathcal{E}_K(m)$ is order preserving encryption of a message $m \in \{0, 1\}^d$ under key K, where d is the block size of OPE.*

Note that communications are made over a secure (authenticated) channel maintained with a symmetric key (e.g., via SSL/TLS as in Algorithm 1) to ensure confidentiality and authentication. For the sake of brevity, we will only write encryptions but not the authentication tags (e.g., Message Authentication Codes [19]) for the rest of the paper.

III. *LP-3PSS*

We now present our proposed scheme that we call *LP-3PSS* (location privacy for 3-party spectrum sensing architecture), which offers high location privacy and low overhead, and uses an additional entity in the network, referred to as Gateway (GW) (thus ”3P” refers to the 3 parties: SUs, FC, and GW). GW enables a higher privacy by preventing FC from even learning the order of encrypted RSS values of SUs which was allowed in $LPOS$ [15]. GW also learns nothing but secure comparison outcome of RSS values and τ, as in YM but only using OPE. Thus, no entity learns any information on RSS or τ beyond a pairwise secure comparison, which is the minimum information required for a voting-based decision.

- *Intuition*: The main idea behind *LP-3PSS* is simple yet very powerful: We enable GW to privately compare n distinct OPE encryptions of τ and RSS values, which were computed under n pairwise keys established between FC and SUs. These OPE encrypted pairs permit GW to learn the comparison outcomes without deducing any other information. GW then sends these comparison results to FC to make the final decision. FC learns no information on RSS values and SUs cannot obtain the value of τ, which complies with our Security Objectives 1 and 2. Note that *LP-3PSS* relies *only on symmetric cryptography* to guarantee the location privacy of SUs. Hence, it is the *most computationally efficient and compact* scheme among all alternatives (see Section V), but with an additional entity in the system.

LP-3PSS is described in Algorithm 1 and outlined below.

Algorithm 1 *LP-3PSS* Algorithm

```
    Initialization: Executed only once.
1:  $FC$ sets energy sensing, optimal voting thresholds $\tau$, $\lambda$,
    and weights vector $\boldsymbol{w} \leftarrow \mathbf{1}$, respectively.
2:  Entities establish private pairwise keys and maintain authenticated secure channels (e.g., via SSL/TLS) as follows:
    • $k_{FC,i}$ between $FC$ and each user $U_i$, $i = 1, \ldots, n$.
    • $k_{GW,i}$ between $GW$ and each user $U_i$, $i = 1, \ldots, n$.
    • $k_{FC,GW}$ between $FC$ and $GW$.
3:  $FC$ computes $c_i \leftarrow \mathcal{E}_{k_{FC,GW}}(OPE.\mathcal{E}_{k_{FC,i}}(\tau))$, $i = 1, \ldots, n$ and sends $\{c_i\}_{i=1}^{n}$ to $GW$.

    Private Sensing: Executed every sensing period $t_w$
4:  $U_i$ computes $\varsigma_i \leftarrow \mathcal{E}_{k_{GW,i}}(OPE.\mathcal{E}_{k_{FC,i}}(RSS_i))$, $i = 1, \ldots, n$ and sends $\{\varsigma_i\}_{i=1}^{n}$ to $GW$.
5:  $GW$ obtains $OPE.\mathcal{E}_{k_{FC,i}}(\tau) \leftarrow \mathcal{D}_{k_{FC,GW}}(c_i)$ and $OPE.\mathcal{E}_{k_{FC,i}}(RSS_i) \leftarrow \mathcal{D}_{k_{GW,i}}(\varsigma_i)$, $i = 1, \ldots, n$.
6:  for $i = 1, \ldots, n$ do
7:      if $OPE.\mathcal{E}_{k_{FC,i}}(RSS_i) < OPE.\mathcal{E}_{k_{FC,i}}(\tau)$ then $b_i \leftarrow 0$
8:      else $b_i \leftarrow 1$
9:  $GW$ computes $\zeta \leftarrow \mathcal{E}_{k_{FC,GW}}(\{b_i\}_{i=1}^{n})$ and sends $\zeta$ to $FC$.
10: $FC$ decrypts $\zeta$ and computes $v \leftarrow \sum_{i=1}^{n} w_i \times b_i$
11: if $v \geq \lambda$ then $dec \leftarrow$ Channel busy
12: else $dec \leftarrow$ Channel free
13: $FC$ updates the credibility score $\varphi_i$ and weight $w_i$ of each
    user $U_i$ as in equations 3 and 4 for $i = 1, \ldots, n$
        return $dec$

    Update after $\mathcal{G}$ Membership Changes or Breakdown:
14: If a user joins the network, it needs to establish a pairwise secret key with $FC$ and $GW$. If $SU$(s) join/leave or breakdown, $\lambda$ is updated as $\lambda$'.
15: Follow the private sensing steps with new $\lambda$'.
```

- *Initialization:* Let $(\mathcal{E}, \mathcal{D})$ be IND-CPA secure [19] block cipher (e.g. AES) encryption/decryption operations. FC establishes a secret key with each SU and GW. GW establishes a secret key with each SU. FC encrypts τ with OPE using $k_{FC,i}$, $i = 1 \ldots n$. FC then encrypts OPE ciphertexts with $\mathcal{E}$ using $k_{FC,GW}$ and sends these c_is to GW, $i = 1 \ldots n$. Since these encryptions are done offline at the beginning of the protocol, they do not impact the online private sensing phase. FC may also pre-compute a few extra encrypted values in the case of new users joining the sensing.

- *Private Sensing:* Each U_i encrypts RSS_i with OPE using $k_{FC,i}$, which was used by FC to OPE encrypt τ value. U_i then encrypts this ciphertext with $\mathcal{E}$ using key $k_{GW,i}$, and sends the final ciphertext ς_i to GW. GW decrypts $2n$ ciphertexts c_is and ς_is with $\mathcal{D}$ using $k_{FC,GW}$ and $k_{GW,i}$, which yields OPE encrypted values. GW then compares each OPE encryption of RSS with its corresponding OPE encryption of τ. Since both were encrypted with the same key, GW can compare them and conclude which one is greater as in Step 7. GW stores the outcome of each comparison in a binary vector $\boldsymbol{b}$, encrpyts and sends it to FC. Finally, FC compares the summation of votes v to the optimal voting threshold λ to make the final decision about spectrum availability and updates the reputation scores of the users.

- *Update after $\mathcal{G}$ Membership Changes or Breakdown:* Each new user joining the sensing just establishes a pairwise secret key with FC and GW. This has no impact on existing users. If some users leave the network, FC and GW remove their secret keys, which also has no impact on existing users. In both cases, and also in the case of a breakdown or failure, λ must be updated accordingly.

IV. Security Analysis

We first describe the underlying security primitives, on which our schemes rely, and then precisely quantify the information leakage of our schemes, which we prove to achieve our Security Objectives 1 and 2.

Fact 1. *An OPE is* indistinguishable under ordered chosen-plaintext attack (IND-OCPA) *[18] if it has no leakage, except the order of ciphertexts (e.g. [21], [22]).*

Let $\mathcal{E}$ and $OPE.\mathcal{E}$ be *IND-CPA secure* [19] and *IND-OCPA secure* symmetric ciphers, respectively. $(\{RSS_i^j\}_{i=1,j=1}^{n,l}, \tau)$ are RSS values and τ of each U_i and FC for sensing periods $j = 1, \ldots, l$ in a group $\mathcal{G}$. $(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3)$ are history lists, which include all values learned by entities U_i, FC and GW, respectively, during the execution of the protocol for all sensing periods and membership status of $\mathcal{G}$. Vector $\vec{V}$ is a list of IND-CPA secure values transmitted over secure (authenticated) channels. $\vec{V}$ may be publicly observed by all entities including external attacker $\mathcal{A}$. Hence, $\vec{V}$ is a part of all lists $(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3)$. Values (jointly) generated by an entity such as cryptographic keys or variables stored only by the entity itself (e.g., λ, π) are not included in history lists for the sake of brevity.

Theorem 1. *Under Security Assumptions 1 and 2,* LP-3PSS *leaks no information on* $(\{RSS_i^j\}_{i=1,j=1}^{n,l}, \tau)$ *beyond*

TABLE I: Computational overhead comparison

Scheme	**Computation**		
	FC	***SU***	***GW***
LP-3PSS	$\mathcal{D} + \beta \cdot (\mathcal{E} + OPE_E)$	$OPE_E + \mathcal{E}$	$n \cdot \mathcal{D} + \mathcal{E}$
$LPOS$	$1/2 \cdot (2 + log\ n) \cdot \gamma \cdot \lvert p \rvert \cdot Mulp$	$(2\gamma \cdot \lvert p \rvert + 2\gamma) \cdot Mulp + OPE + 2\mu \cdot log\ n \cdot PMulQ$	-
$PPSS$	$H + (n+2) \cdot Mulp + (2^{\gamma-1} \cdot n + 2) \cdot Expp$	$H + 2Expp + Mulp$	-
$PDAFT$	$2ExpN^2 + InvN^2 + y \cdot MulN^2$	$2ExpN^2 + MulN^2$	$n \cdot MulN^2$

(i) Variables: κ security parameter, N: modulus in Paillier, p: modulus of El Gamal, H: cryptographic hash operation, K: secret group key of OPE. $Expu$ and $Mulu$ denote a modular exponentiation and a modular multiplication over modulus u respectively, where $u \in \{N, N^2, p\}$. $InvN^2$: modular inversion over N^2, $PMulQ$: point multiplication of order Q, $PAddQ$: point addition of order Q. y: number of servers needed for decryption in $PDAFT$. **(ii) Parameters size:** For a security parameter $\kappa = 80$, suggested parameter sizes by *NIST 2012* are given by : $|N| = 1024$, $|p| = 1024$, $|Q| = 192$ as indicated in [20]. **(iii) OPE**: we rely on OPE scheme proposed by Boldyreva [18] for our evaluation because of its popularity and public implementation but our schemes can use *any secure* OPE scheme (e.g., [18], [21], [22]) as a building block. **(v)** $\mathcal{E}$: We rely on AES [23][1] as our ($\mathcal{E}$,$\mathcal{D}$) for our cost analysis.

IND-CPA secure $\{\vec{V}^j\}_{j=1}^{l}$, *IND-OCPA secure pairwise order* $\{OPE.\mathcal{E}_{k_{FC,i}}(RSS_i^j),\ OPE.\mathcal{E}_{k_{FC,i}}(\tau)\}_{i=1,j=1}^{n,l}$ *to* GW *and* $\{b_i^j\}_{i=1,j=1}^{n,l}$ *to FC.*

Proof: $\vec{V}^j = \{c_i^j, \varsigma_i^j, \zeta^j\}_{i=1,j=1}^{n,l}$, where $\{c_i^j\}_{i=1,j=1}^{n,l}$ and $\{\varsigma_i^j, \zeta^j\}_{i=1,j=1}^{n,l}$ are generated at the initialization and privacy sensing in Algorithm 1, respectively. History lists are as follows for each sensing period $j = 1, \ldots, l$:

$$\begin{aligned} \mathcal{L}_1 &= \vec{V}^j, \quad \mathcal{L}_2 = (\{b_i^j\}_{i=1,j=1}^{n,l}, \vec{V}^j), \\ \mathcal{L}_3 &= (\{OPE.\mathcal{E}_{k_{FC,i}}(RSS_i^j), OPE.\mathcal{E}_{k_{FC,i}}(\tau)\}_{i=1,j=1}^{n,l}, \vec{V}^j, \\ &\quad \{b_i^j\}_{i=1,j=1}^{n,l}) \end{aligned}$$

Variables in $(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3)$ are IND-CPA secure and IND-OCPA secure, and therefore leak no information beyond the pairwise order of ciphertexts to GW by Fact 1.

Any membership status update on $\mathcal{G}$ requires an authenticated channel establishment or removal for joining or leaving members, whose private keys are independent from each other. Hence, history lists $(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3)$ are computed identically as described above for the new membership status of $\mathcal{G}$, which are IND-CPA secure and IND-OCPA secure. □

Corollary 1. *Theorem 1 guarantees that in our scheme, RSS values and* τ *are IND-OCPA secure for all sensing periods and membership changes. Hence, our scheme achieves Objectives 1 and 2 as required.*

V. Performance Evaluation

We now evaluate our proposed scheme, *LP-3PSS*, by comparing it to existent approaches that we briefly explain below.

A. Existing Approaches

$PPSS$ [3] uses secret sharing and the Privacy Preserving Aggregation (PPA) process proposed in [24] to hide the content of specific sensing reports and uses dummy report injections to cope with the DLP attack. $LPOS$ [15] also uses OPE but in a completely different way than how we use it in this paper. Users OPE encrypt their RSS values, send them to FC which, based on the order of the encrypted RSSs, performs at worst a logarithmic number of yao's millionaires secure comparisons [25] between τ and RSSs and then makes a final decision about spectrum availability. $PDAFT$ [14] combines Paillier cryptosystem [26] with Shamir's secret sharing [27], where a set of smart meters sense the consumption of different households, encrypt their reports using Paillier, then send them to a gateway. The gateway multiplies these reports and forwards the result to the control center, which selects a number of servers (among all servers) to cooperate in order to decrypt the aggregated result. $PDAFT$ requires a dedicated gateway, just like *LP-3PSS*, to collect the encrypted data, and a minimum number of working servers in the control center to decrypt the aggregated result.

B. Performance Analysis and Comparison

We focus on communication and computational overheads. We consider the overhead incurred during the sensing operations but not that related to system initialization (e.g. key establishment), where most of the computation and communication is done offline. We model the membership change events in the network as a random process R that takes on 0 and 1, and whose average is μ. $R = 0$ means that no change occurred in the network and $R = 1$ means that some users left/joined the sensing task. Let β(t) be a function that models the average number of users that join the sensing at the current sensing period t, where

$$\beta(t) = \begin{cases} n(t) - n(t-1), & \text{if } n(t) - n(t-1) > 0 \ \&\ R(t) = 1 \\ 0, & \text{otherwise} \end{cases}$$

The execution times of the different primitives and protocols were measured on a laptop running Ubuntu 14.10 with 8GB of RAM and a core M 1.3 GHz Intel processor, with cryptographic libraries MIRACL [28], Crypto++ [29] and *Louismullie*'s Ruby implementation of OPE [30].

Computational Overhead: Table I provides an analytical computational overhead comparison including the details of variables, parameters and the overhead of building blocks.

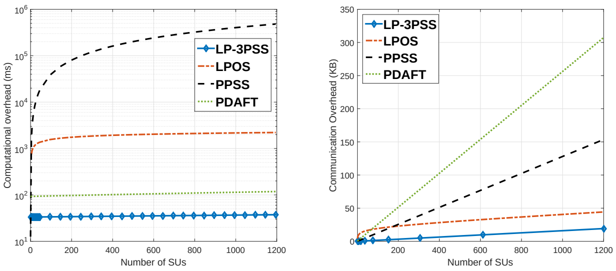
In *LP-3PSS*, FC requires only a small constant number of $(\mathcal{D}, \mathcal{E}, OPE)$ operations. An SU requires one OPE and $\mathcal{E}$ encryptions of its RSS. Finally, GW requires one $\mathcal{D}$ operation per user and one $\mathcal{E}$ of vector $\boldsymbol{b}$. All computations in *LP-3PSS* rely on only symmetric cryptography, which makes it *the most computationally efficient scheme among all alternatives*.

For illustration purpose, we plot in Fig. 1(a) the system end-to-end computational overhead of the different schemes.

TABLE II: Communication overhead comparison

Scheme	**Communication**
LP-3PSS	$(n+1) \cdot \epsilon_{\mathcal{E}}$
$LPOS$	$2\gamma \cdot \lvert p \rvert \cdot (2 + log\ n) + n \cdot \epsilon_{OPE} + \mu \cdot \lvert Q \rvert \cdot log\ n$
$PPSS$	$\lvert p \rvert \cdot n + \beta \cdot \mu \cdot \lvert p \rvert \cdot n$
$PDAFT$	$\lvert N \rvert \cdot (2(n+1) + \beta)$

$\epsilon_{OPE} = 128\, bits$: maximum ciphertext size obtained under OPE encryption, $\epsilon_{\mathcal{E}}$: size of ciphertext under $\mathcal{E}$.



(a) Computational overhead

(b) Communication overhead

Fig. 1: Performance compariosn, $\kappa = 80$, $\beta = 5$, $\mu = 20\%$

Fig. 1(a) shows that *LP-3PSS* is several order of magnitudes faster than the other schemes including $LPOS$, that we proposed in a previous work, for any number of users.

Communication Overhead: Table II provides the analytical communication overhead comparison. *LP-3PSS* requires $(n+1)$ $\mathcal{E}$ ciphertexts and single ζ, which are significantly smaller than the ciphertexts transmitted in the other schemes.

We further compare our scheme with its counterparts in terms of communication overhead in Fig. 1(b). Fig. 1(b) shows that *LP-3PSS* has the smallest communication overhead since, again, it relies on symmetric cryptography only. $PPSS$ and $PDAFT$ have a very high communication overhead due to the use of expensive public key encryptions (e.g., Pailler [26]).

Overall, our performance analysis indicates that *LP-3PSS* is significantly more efficient than all other counterpart schemes in terms of computation and communication overhead, even for increased values of the security parameters, but with the cost of including an additional entity.

VI. Conclusion

We developed an efficient scheme for cooperative spectrum sensing that protects the location privacy of SUs with a low cryptographic overhead while guaranteeing an efficient spectrum sensing. Our scheme is secure and robust against users dynamism, failures, and user maliciousness. Our performance analysis indicates that our scheme outperforms existing alternatives in various metrics.

References

[1] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, “Cooperative spectrum sensing in cognitive radio networks: A survey,” *Physical Communication*, vol. 4, pp. 40–62, 2011.

[2] O. Fatemieh, A. Farhadi, R. Chandra, and C. A. Gunter, “Using classification to protect the integrity of spectrum measurements in white space networks.” in *NDSS*, 2011.

[3] S. Li, H. Zhu, Z. Gao, X. Guan, K. Xing, and X. Shen, “Location privacy preservation in collaborative spectrum sensing,” in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 729–737.

[4] S. B. Wicker, “The loss of location privacy in the cellular age,” *Communications of the ACM*, vol. 55, no. 8, pp. 60–68, 2012.

[5] Q. Yan, M. Li, T. Jiang, W. Lou, and Y. Hou, “Vulnerability and protection for distributed consensus-based spectrum sensing in cognitive radio networks,” in *INFOCOM, 2012 Proc. IEEE*, March 2012.

[6] Z. Qin, S. Yi, Q. Li, and D. Zamkov, “Preserving secondary users’ privacy in cognitive radio networks,” in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 772–780.

[7] W. Wang and Q. Zhang, “Privacy-preserving collaborative spectrum sensing with multiple service providers,” *Wireless Communications, IEEE Transactions on*, vol. 14, no. 2, pp. 1011–1019, 2015.

[8] D. Yang, X. Fang, and G. Xue, “Truthful incentive mechanisms for k-anonymity location privacy,” in *INFOCOM, 2013 Proceedings IEEE*, April 2013, pp. 2994–3002.

[9] X. Zhao, L. Li, and G. Xue, “Checking in without worries: Location privacy in location based social networks,” in *INFOCOM, 2013 Proceedings IEEE*, April 2013, pp. 3003–3011.

[10] Z. Gao, H. Zhu, Y. Liu, M. Li, and Z. Cao, “Location privacy in database-driven cognitive radio networks: Attacks and countermeasures,” in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 2751–2759.

[11] E. Troja and S. Bakiras, “Leveraging p2p interactions for efficient location privacy in database-driven dynamic spectrum access,” in *Proc. of the 22nd ACM SIGSPATIAL International Conf. on Advances in Geographic Information Systems*. ACM, 2014, pp. 453–456.

[12] S. Liu, H. Zhu, R. Du, C. Chen, and X. Guan, “Location privacy preserving dynamic spectrum auction in cognitive radio network,” in *Distributed Computing Systems (ICDCS), 2013 IEEE 33rd International Conference on*. IEEE, 2013, pp. 256–265.

[13] K. S. McCurley, “The discrete logarithm problem,” in *Proc. of Symp. in Applied Math*, vol. 42, 1990, pp. 49–74.

[14] L. Chen, R. Lu, and Z. Cao, “PDAFT: A privacy-preserving data aggregation scheme with fault tolerance for smart grid communications,” *Peer-to-Peer Networking and Applications*, pp. 1–11, 2014.

[15] M. Grissa, A. A. Yavuz, and B. Hamdaoui, “Lpos: Location privacy for optimal sensing in cognitive radio networks,” in *Global Communications Conference (GLOBECOM), 2015 IEEE*. IEEE, 2015.

[16] W. Zhang, R. K. Mallik, and K. Letaief, “Cooperative spectrum sensing optimization in cognitive radio networks,” in *Communications, 2008. ICC’08. IEEE International Conf. on*. IEEE, 2008, pp. 3411–3415.

[17] K. Arshad and K. Moessner, “Robust collaborative spectrum sensing based on beta reputation system,” in *Future Network & Mobile Summit (FutureNetw), 2011*. IEEE, 2011, pp. 1–8.

[18] A. Boldyreva, N. Chenette, Y. Lee, and A. O´ neill, “Order-preserving symmetric encryption,” in *Advances in Cryptology-EUROCRYPT 2009*. Springer, 2009, pp. 224–241.

[19] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*. Chapman & Hall/CRC, 2007.

[20] “Cryptographic key length recommendation,” http://www.keylength.com/en/compare/#Biblio6.

[21] R. A. Popa, F. H. Li, and N. Zeldovich, “An ideal-security protocol for order-preserving encoding,” in *Security and Privacy (SP), IEEE Symposium on*. IEEE, 2013, pp. 463–477.

[22] F. Kerschbaum and A. Schroepfer, “Optimal average-complexity ideal-security order-preserving encryption,” in *Proc. of the SIGSAC Conf. on Computer and Comm. Security*. ACM, 2014, pp. 275–286.

[23] J. Daemen and V. Rijmen, “Aes proposal: Rijndael,” 1999.

[24] E. Shi, T.-H. H. Chan, E. G. Rieffel, R. Chow, and D. Song, “Privacy-preserving aggregation of time-series data.” in *NDSS*, vol. 2, no. 3, 2011.

[25] H.-Y. Lin and W.-G. Tzeng, “An efficient solution to the millionaires’ problem based on homomorphic encryption,” in *Applied Cryptography and Network Security*. Springer, 2005, pp. 456–466.

[26] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Advances in cryptology-EUROCRYPT99*. Springer, 1999, pp. 223–238.

[27] A. Shamir, “How to share a secret,” *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[28] “Miracl library,” http://www.certivox.com/miracl.

[29] “Crypto++ library,” http://www.cryptopp.com/.

[30] “Ruby ope implementation,” https://github.com/louismullie/ope-rb.