Preserving Incumbent Users' Privacy in Server-Driven Dynamic Spectrum Access Systems

Yanzhi Dou*, He Li*, Kexiong (Curtis) Zeng*, Jinshan Liu*, Yaling Yang*, Bo Gao† and Kui Ren‡
* Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA
† Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
‡ Department of Computer Science and Engineering, University at Buffalo, State University of New York, Buffalo, USA
Email: {yzdou, heli, kexiong6, jinshan, yyang8}@vt.edu, gaobo@ict.ac.cn, kuiren@buffalo.edu

***Abstract*—Dynamic spectrum access (DSA) technique has emerged as a fundamental approach in improving spectrum utilization to mitigate the spectrum scarcity problem. As a key form of DSA, government is proposing to release more federal spectrum for sharing with commercial wireless users. However, the flourish of federal-commercial sharing hinges upon how federal privacy issues are managed. In current DSA proposals, the sensitive operation information of federal incumbent users (IUs) needs to be shared with a dynamic spectrum access system (SAS) to realize spectrum allocation. However, SAS is not necessarily trust-worthy for holding such sensitive IU data, especially considering that FCC allows some industry third parties (e.g., Google) to operate SAS for better efficiency and scalability. Therefore, the current proposals dissatisfy the IUs' privacy requirement. To address the privacy issues, this paper presents an IU-privacy-preserving SAS (IP-SAS) design, which realizes the spectrum allocation process through secure computation over ciphertext based on homomorphic encryption so that none of the IU operation information is exposed to SAS.**

***Index Terms*—Dynamic spectrum access; privacy; homomorphic encryption;**

I. System Design

A. Problem Statement

We consider a SAS involving three parties: IUs, SUs, and SAS Server. SAS Server refers to a cloud-based spectrum management infrastructure that allocates spectrum resources while considering incumbent and secondary operation protection according to some interference management method. In this paper, we focus on the exclusion zone (E-Zone) method for interference management. The E-Zone method defines a spatial region around an IU where SUs are forbidden to operate to avoid mutual interference with the IU. The privacy issue of the protection-zone method, which considers interference aggregation from multiple IUs and SUs, has been addressed in our previous work [1]. In a typical scenario of E-Zone-based SAS systems, IUs first compute their E-Zones and send the E-Zone data to SAS in the initialization phase. When an SU wants to access the spectrum, it needs to provide its operation parameters and geolocation to SAS. SAS checks whether the SU is within the E-Zone of any IU. For a given spectrum, if the answer is yes (no), SAS denies (permits) the SU's spectrum access to this spectrum.

We assume SAS Server is semi-honest, which means it exactly follows the protocol as described above, but attempts to infer private operation data of IUs from the information communicated to it. Our goal is to design a privacy-preserving SAS that can correctly realize spectrum allocation without exposing any information that can potentially lead to IU privacy violation to the semi-honest SAS Server.

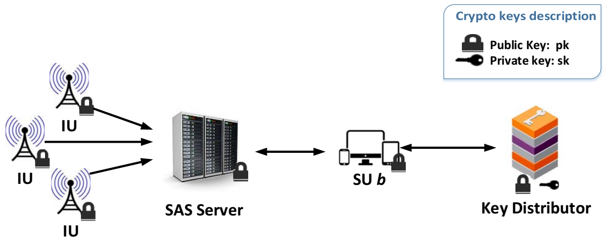

Fig. 1. IP-SAS overview

Our design is based on Paillier cryptosystem, which is an additive-homomorphic encryption scheme $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Add})$ that allows addition operation Add on ciphertexts and generates an encrypted result, when decrypted, corresponding to the sum of the plaintexts. Mathematically, $\mathsf{Dec}_{\mathsf{sk}}\Big(\mathsf{Add}\big(\mathsf{Enc}_{\mathsf{pk}}(m_1), \mathsf{Enc}_{\mathsf{pk}}(m_2)\big)\Big) = m_1 + m_2$, where m_1 and m_2 are two plaintext messages, $(\mathsf{pk}, \mathsf{sk})$ is a key pair of Paillier cryptosystem generated by KeyGen. In the remainder of this paper, we use $\widehat{m}$ to denote the Paillier encryption of a plaintext message m.

B. Design Overview

As shown in Figure 1, IP-SAS involves four parties: (1) a SAS Server $\mathcal{S}$ for spectrum allocation, (2) IUs, (3) SUs, and (4) a Key Distributor $\mathcal{K}$. $\mathcal{K}$ creates a Paillier public/private key pair $(\mathsf{pk}, \mathsf{sk})$ and is trusted for keeping sk a secret only known to itself. In the real world, the role of $\mathcal{K}$ can be played by some authorities such as FCC and NTIA. The protocol of IP-SAS is shown in Table I. In the following, we describe the details of each step.

C. E-Zone Information Generation & Representation

To reduce unnecessary wastes of spectrum resources, IUs have to compute E-Zone boundaries accurately. Following the recommendations in [2], we assume that the E-Zones computed by an IU can have multiple tiers. Each tier corresponds to SUs with a specific operation parameter setting. An SU operation parameter setting is a tuple $(f_s, h_s, p_{ts}, g_{rs}, i_s)$. Similarly, an IU operation parameter setting is a tuple

1063-6927/16 $31.00 © 2016 IEEE
DOI 10.1109/ICDCS.2016.40

TABLE I
THE PROTOCOL OF IP-SAS

I. Initialization Phase:
$\mathcal{K}$:
(1) $\mathcal{K}$ runs KeyGen and generates a Paillier key pair $(\mathsf{pk}, \mathsf{sk})$.
pk is distributed to $\mathcal{S}$ and IUs, and sk is kept secret.
IUs (numbered as $1, 2, ..., k, ..., K$):
(2) IU k calculates its E-Zone map $\mathbf{T}_k$.
(3) IU k encrypts $\mathbf{T}_k$ with pk and gets $\widehat{\mathbf{T}}_k$.
(4) IU k uploads $\widehat{\mathbf{T}}_k$ to $\mathcal{S}$.
$\mathcal{S}$:
(5) Upon all IUs having uploaded their E-Zone map, $\mathcal{S}$ aggregates the E-Zone map of all IUs and generates $\widehat{\mathbf{M}}$.
II. Spectrum Computation Phase:
SU b:
(6) SU b submits spectrum request containing its operation parameters $(h_s, p_{ts}, g_{rs}, i_s)$ and location l to $\mathcal{S}$.
$\mathcal{S}$:
(7) $\mathcal{S}$ retrieves the corresponding entry in the global E-Zone map $\widehat{\mathbf{M}}$ and obtains $\widehat{\mathbf{X}}_b$.
(8) $\mathcal{S}$ adds random blinding factor $\widehat{\boldsymbol{\beta}}$ to $\widehat{\mathbf{X}}_b$ to generate $\widehat{\mathbf{Y}}_b$.
(9) $\mathcal{S}$ returns $\widehat{\mathbf{Y}}_b$ and $\boldsymbol{\beta}$ to SU b.
III. Recovery Phase:
SU b:
(10) SU b relays $\widehat{\mathbf{Y}}_b$ to $\mathcal{K}$ for decryption.
$\mathcal{K}$:
(11) $\mathcal{K}$ decrypts $\widehat{\mathbf{Y}}_b$ with sk and returns $\mathbf{Y}_b$ to SU b.
SU b:
(12) SU b recovers $\mathbf{X}_b$ by removing the blinding factor $\boldsymbol{\beta}$ from $\mathbf{Y}_b$.

$(f_i, h_i, p_{ti}, g_{ri}, i_i)$. f_s and f_i denote operation frequency, h_s and h_i denote antenna height, p_{ts} and p_{ti} denote transmitter effective radiated power, g_{rs} and g_{ri} denote receiver antenna gain, i_s and i_i denote receiver interference tolerance threshold.

Plug SU and IU's operation parameter settings into a sophisticated radio propagation model that incorporates terrain details, IU can accurately compute its E-Zone for a specific SU operation setting, which is denoted by $EZ(f_s, h_s, p_{ts}, g_{rs}, i_s)$. To reduce computation complexity, we discretize the operation parameters so that each IU k in IP-SAS captures its multi-tier E-Zone information using a multidimensional E-Zone map matrix $\mathbf{T}_k := \{T_k(l, f, h_s, p_{ts}, g_{rs}, i_s)\}$, where

$$T_k(l, f, h_s, p_{ts}, g_{rs}, i_s) := \begin{cases} \epsilon, & \text{grid } l \in EZ(f, h_s, p_{ts}, g_{rs}, i_s) \\ 0, & \text{grid } l \notin EZ(f, h_s, p_{ts}, g_{rs}, i_s) \end{cases} \tag{1}$$

ϵ is a positive random number used to denote that SU is in the E-Zone and should not be allowed to operate, and 0 means SU is out of the E-Zone and can be allowed to operate.

To protect IU privacy against semi-honest $\mathcal{S}$, each entry of $\mathbf{T}_k$ is encrypted by pk before sending to $\mathcal{S}$. Therefore, IU k transmits $\widehat{\mathbf{T}}_k := \{\widehat{T_k}(l, f, h_s, p_{ts}, g_{rs}, i_s)\}$ when updating $\mathcal{S}$ with its E-Zone map.

D. E-Zone Map Aggregation in $\mathcal{S}$

Assume that there are altogether K IUs registered in $\mathcal{S}$ and all of them have sent in their E-Zone map. The first step of $\mathcal{S}$ is to aggregate all the IUs' E-Zone maps to create a global E-Zone map $\widehat{\mathbf{M}} := \{\widehat{M}(l, f, h_s, p_{ts}, g_{rs}, i_s)\}$ by

$$\widehat{\mathbf{M}} := \oplus_{k \in \{1,2,...,K\}} \widehat{\mathbf{T}}_k, \tag{2}$$

where $\oplus$ is the homomorphic version of summation symbol $\sum$. From formula (1), it is easy to see that for an SU at location l with operation parameter setting $(h_s, p_{ts}, g_{rs}, i_s)$, if $M(l, f, h_s, p_{ts}, g_{rs}, i_s) = 0$, the grid l is out of E-Zones of all IUs and spectrum f is available for the SU; If $M(l, f, h_s, p_{ts}, g_{rs}, i_s) > 0$, the grid l is within the E-Zone of at least one IU and spectrum f is unavailable for the SU.

E. Spectrum Computation Phase & Recovery Phase

When an SU b needs to access the spectrum, it submits a spectrum request containing its operation parameters $(h_s, p_{ts}, g_{rs}, i_s)$ and location l in plaintext to $\mathcal{S}$. $\mathcal{S}$ retrieves the corresponding entries in the global E-Zone map $\widehat{\mathbf{M}}$ and generates $\widehat{\mathbf{X}}_b := \{\widehat{X}_b(f)\}$, where

$$\widehat{X}_b(f) := \widehat{M}(l, f, h_s, p_{ts}, g_{rs}, i_s). \tag{3}$$

Essentially, $\widehat{\mathbf{X}}_b$ holds the information of spectrum availability for SU b. $\mathcal{S}$ then homomorphically adds some random blinding factors $\widehat{\boldsymbol{\beta}} := \{\widehat{\beta}(f)\}$ to obfuscate the results as follows:

$$\widehat{Y}_b(f) := \mathsf{Add}(\widehat{X}_b(f), \widehat{\beta}(f)), \tag{4}$$

where the plaintext of the blinding factor $\beta(f)$ is a one-time random number. $\widehat{\mathbf{Y}}_b$ and the plaintext $\boldsymbol{\beta}$ are both sent back to SU b. SU b needs to decrypt $\widehat{\mathbf{Y}}_b$ for the spectrum availability information, so it sends $\widehat{\mathbf{Y}}_b$ to $\mathcal{K}$ for decryption. $\mathcal{K}$ decrypts every entry of $\widehat{\mathbf{Y}}_b$ using sk and gets $\mathbf{Y}_b$. $\mathcal{K}$ cannot infer the spectrum availability information from $\mathbf{Y}_b$ since it does not have the values of the blinding factors $\boldsymbol{\beta}$ to recover $\mathbf{X}_b$. $\mathcal{K}$ sends $\mathbf{Y}_b$ to SU b, which uses $\boldsymbol{\beta}$ to recover the correct spectrum computation results $\mathbf{X}_b$ by

$$X_b(f) = Y_b(f) - \beta(f). \tag{5}$$

II. PRELIMINARY RESULTS

We evaluate IP-SAS on a 154.82 km^2 area in Washington DC. We employ the Longley-Rice model provided by SPLAT! [3] to generate the E-Zone maps in this area. High resolution terrain data SRTM3 [4] is fed to SPLAT!. We build IP-SAS on a Paillier cryptosystem of 112-bit security level. Evaluation results show that the total time to process an SU spectrum request in SAS Server is around 1.25 seconds, and the total communication overhead for SU is 17.75 KB. The evaluation results show that IP-SAS's performance is good enough even for mobile SUs in highly changeable environment.

ACKNOWLEDGMENT

This work is supported by US National Science Foundation under grant CNS-1547366.

REFERENCES

[1] Y. Dou, K. C. Zeng, H. Li, Y. Yang, B. Gao, C. Guan, and S. Li, "P2-SAS: Preserving Users' Privacy in Centralized Dynamic Spectrum Access Systems," *to appear in Proceedings of the 17th ACM MobiHoc*, 2016.

[2] A. Ullah, S. Bhattarai, J.-M. Park, J. Reed, D. Gurney, and B. Bahrak, "Multi-Tier Exclusion Zones for Dynamic Spectrum Sharing," in *IEEE ICC*, 2015.

[3] http://www.qsl.net/kd2bd/splat.html.

[4] http://dds.cr.usgs.gov/srtm/version2_1/SRTM3/.