Privacy-Preserving and Secure Spectrum Sharing for Database-Driven Cognitive Radio Networks

Saleh Darzi[a,*], Gökcan Cantali[b,c,2], Attila Altay Yavuz[a,3] and Gürkan Gür[b,4]

[a] Bellini College of Artificial Intelligence, Cybersecurity and Computing, University of South Florida, Tampa, 33620, Florida, USA
[b] Zurich University of Applied Sciences (ZHAW), Winterthur, 8401, Switzerland
[c] Department of Informatics, University of Zurich (UZH), Zurich, 8450, Switzerland

ARTICLE INFO

Keywords:
Database-Driven Cognitive Radio Networks
Location Privacy
Anonymous Credentials
Location Proof
Counter-DoS

ABSTRACT

Database-driven cognitive radio networks (DB-CRNs) enable dynamic spectrum sharing through geolocation databases but introduce critical security and privacy challenges, including mandatory location disclosure, susceptibility to location spoofing, and denial-of-service (DoS) attacks on centralized services. Existing approaches address these issues in isolation and lack a unified, regulation-compliant solution under realistic adversarial conditions. In this work, we present a unified security framework for DB-CRNs that simultaneously provides location privacy, user anonymity, verifiable location, and DoS resilience. Our framework, denoted as `SLAPX`, enables privacy-preserving spectrum queries using delegatable anonymous credentials, supports adaptive location verification without revealing precise user location, and mitigates DoS attacks through verifiable delay functions (VDFs) combined with RLRS-based rate limiting. Extensive cryptographic benchmarking and network simulations demonstrate that `SLAPX` achieves significantly lower latency and communication overhead than existing solutions while effectively resisting location spoofing and DoS attacks. These results show that `SLAPX` is practical and well-suited for secure next-generation DB-CRN deployments.

1. Introduction

The rapid growth of wireless technologies, driven by mobile devices, Internet-of-Things (IoT) deployments, and emerging spectrum-hungry applications, has intensified pressure on traditionally static spectrum allocation policies [4, 11, 25]. To address spectrum underutilization, database-driven cognitive radio networks (DB-CRNs) and dynamic spectrum sharing have been introduced, enabling Secondary Users (SUs) to opportunistically access licensed spectrum owned by Primary Users (PUs) through regulatory-approved geolocation databases (DBs). A prominent example is the U.S. Citizens Broadband Radio Service (CBRS), which operates in the 3.5 GHz band originally allocated to federal and satellite services [4, 24]. Moreover, dynamic spectrum sharing is expected to be a foundational element in future networks such as 6G systems [36].

While this paradigm significantly improves spectrum efficiency, it also introduces a wide range of security and privacy challenges. In particular, regulatory authorities (e.g., FCC and ITU) require users to continuously report sensitive operational information, including precise location, device identity, and transmission parameters, to centralized DBs. This requirement raises serious concerns regarding user privacy, anonymity, and long-term profiling [27, 22]. Moreover, the location-centric design of DB-CRNs makes them inherently vulnerable to location spoofing attacks, in which malicious or compromised users falsify their positions to obtain unauthorized spectrum access [47]. At the same time, the centralized reliance on spectrum databases, the broadcast nature of wireless communication, and the proliferation of low-cost devices further expose DB-CRNs to denial-of-service (DoS) attacks that can disrupt spectrum coordination and degrade network availability [14].

Existing solutions for DB-CRNs largely address individual challenges in isolation and do not provide a unified, regulation-compliant framework that jointly ensures location privacy, anonymity, verifiable location, and resilience against DoS attacks under realistic adversarial conditions. In this work, to address those research gaps, our contributions are threefold:

1. We develop and evaluate a novel and efficient framework, namely `SLAPX`, that leverages advanced cryptographic primitives to simultaneously address multiple, often conflicting, objectives in DB-CRNs. The proposed design achieves user location privacy and anonymity during spectrum querying and access while complying with enforced regulations. This paper extends our `SLAP` framework[1], incorporating substantial new material and the following key enhancements and additional features. Designed for realistic adversarial settings, `SLAPX` remains robust against location spoofing and DoS attacks by malicious or compromised users. The key properties of `SLAPX` are outlined as follows: *(i) Efficient Location Privacy-Preserving and Anonymous Spectrum Query:* `SLAPX` provides strong anonymity and location privacy throughout the spectrum access workflow while remaining fully compliant with protocol and regulatory requirements. *(ii) Adaptive Location Proof and Spoofing Resistance for DB-CRN:* `SLAPX` comprises an adaptive location proof algorithm with dual scenario support and

✉ salehdarzi@usf.edu (S. Darzi); canl@zhaw.ch (G. Cantali); attilaayavuz@usf.edu (A.A. Yavuz); gurkan.gur@zhaw.ch (G. Gür)
ORCID(s):

[1] The preliminary version of our approach, namely SLAP, was presented and published at the *IEEE SVCC 2025 Conference* [15].

architectural flexibility that operates under realistic network assumptions with malicious or compromised users. *(iii) DoS Mitigation with DB-CRN Architecture Compliance:* SLAPX provides comprehensive DoS mitigation through a proactive defense mechanism that combines Verifiable Delay Functions (VDFs) with rate limiting enforced via the linkability properties of Revocable-iff-linked Linkable Ring Signatures (RLRSs).

2. We present a system model that integrates SLAPX into a DB-CRN architecture comprising user equipment (UEs), access networks, spectrum databases, and regulatory authorities. This model provides a practical baseline for implementing and benchmarking DB-CRN security mechanisms, and serves as a foundation for evaluating future enhancements, including quantum-safe extensions.
3. We provide extensive simulation-based experiments and numerical analysis, which provide a blueprint for realistic evaluation and benchmarking for further efforts regarding the investigated research questions on privacy-preserving and secure spectrum sharing.

Our paper is organized as follows: in the next section, we present the technical preliminaries, system model, and cryptographic building blocks for our proposed SLAPX framework. In Section 3, we construct threat and security models and attack scenarios relevant to SLAPX. Subsequently, we describe SLAPX's core operations and the detailed algorithmic specification of the framework in Section 4. Then, in Section 5, a comprehensive performance evaluation with simulations and numerical experiments is presented. We discuss the related work to position our contributions in the current body of work and concisely highlight the research gaps addressed by our work to elaborate on our contributions in Section 6. Finally, we conclude with key takeaways and future work.

2. Preliminaries and Building Blocks

This section presents the notations, network architecture, and cryptographic building blocks of our framework.

Notations: The operators $||$, $|x|$, and $\{0,1\}^k$ denote string concatenation, the bit-length of a value x, and the set of k-bit binary strings, respectively, while $\oplus$ represents the bitwise XOR operation. The set of natural numbers is denoted by $\mathbb{N}$. For a sequence $\{x_i\}_{i=1}^{\ell}$, we write $(x_1, \dots, x_\ell)$, and $\overset{\$}{\longleftarrow} \mathcal{S}$ denotes uniform random sampling from a set $\mathcal{S}$. Let $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ be cyclic groups of prime order p, and let $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ denote a bilinear pairing satisfying bilinearity and non-degeneracy. The i-th element of a vector $\mathbf{m}$ is denoted by $\mathbf{m}[i]$, and $h(\cdot)$ represents a cryptographically secure hash function. The security parameter is denoted by λ and p denotes a large prime. Finally, $\mathtt{sk}$, $\mathtt{PK}$, and $\mathtt{ID}$ denote a secret key, a public key, and a user identifier (e.g., a MAC address), respectively, while $\mathtt{SGN.Sign}(\cdot)$ and $\mathtt{SGN.Verify}(\cdot)$ represent standard digital signature generation and verification algorithms.

2.1. System Architecture: DB-CRN

Network Components: The DB-CRN is a spectrum-sharing architecture in which unlicensed devices query trusted geolocation databases to obtain channel availability at their current location [36]. This model was deployed in real FCC TV White Space (TVWS) systems and adopted in the Microsoft and Google database trials [10, 16]. The DB-CRN architecture consists of five key entities [35]:

- ***Spectrum Regulator*:** It is the regulatory authority that defines technical rules for DB-CRN operation, like incumbent protection requirements, device reporting and registration procedures, and transmit parameters [4]. This entity authorizes and oversees geolocation database administrators and provides authoritative incumbent datasets and regulatory constraints. Throughout this manuscript, the FCC is used interchangeably to denote this spectrum regulatory authority for convenience.
- ***Servers*:** They represent external network or cloud services (e.g., CRN, cloud, edge nodes) that users access after obtaining spectrum authorization [22]. They do not participate in spectrum assignment; instead, they simply carry user traffic over the channels approved by the PSD.
- ***Private Spectrum Databases (PSDs)*:** PSDs are FCC-certified, cloud-based geolocation databases (e.g., Google, Microsoft, Spectrum Bridge) responsible for maintaining incumbent information, propagation constraints, and channel availability maps [12]. PSDs process spectrum-availability queries submitted by users and return permissible channels and power limits, synchronizing regularly according to FCC mandates [24].
- ***User Equipments (UEs)*:** UEs are unlicensed devices of users (e.g., smartphones, sensors, and vehicular radios) that seek opportunistic access to licensed spectrum bands [22]. Acting as SUs, they determine their location, submit channel-availability queries to PSDs, and operate strictly within the constraints specified in PSD responses to avoid interfering with PUs. Throughout this paper, the terms client and UE are used interchangeably.
- ***Access Points (APs)*:** APs consist of WiFi access points or cellular base stations (LTE/5G macro or small cells) that provide the IP connectivity required for UEs to reach PSDs and servers over the Internet [15]. APs do not manage spectrum availability but offer backhaul connectivity and, in some advanced architectures, assist with localization or timing support.

Communication Flow and Protocol Stack: In a DB-CRN, UEs communicate with PSDs over standard Internet paths to obtain spectrum availability, while servers and APs provide network connectivity and application-layer services. Communication exchanges follow an IP-based client–server model and use secure, lightweight protocols consistent with real FCC TVWS deployments and the IETF PAWS specification [30]. The communication workflow consists of three primary interactions:

- *UE-to-AP Communication (Access Connectivity):* UEs attach to a nearby AP to obtain Internet connectivity. This step uses standard link-layer and access-network protocols, such as IEEE 802.11 or 3GPP RRC/NAS for LTE/5G [40]. APs forward UE traffic to the Internet and do not participate in the decision-making process [30].

- *UE-to-PSD Communication (Spectrum Querying):* UEs submit spectrum-availability queries directly to PSDs using HTTPS over TLS to ensure confidentiality and integrity of location and device parameters. The request and response formats follow the JSON-based geolocation DB access protocol standardized by the IETF PAWS specification (RFC-7545) [30]. A typical UE-to-PSD exchange includes (i) a registration or device-descriptor message, (ii) a location-based *AVAIL_CHANNELS* query, and (iii) a PSD response containing permissible channels, power limits, and validity intervals. PSDs process each request using incumbent datasets and FCC rules before sending their response [40].
- *UE-to-Server Communication (Application Traffic):* After obtaining spectrum authorization, UEs communicate with application or cloud Servers using standard IP protocols. These Servers do not participate in spectrum allocation; they merely receive traffic over the channels authorized by the PSD. Servers may verify spectrum-access credentials or tokens in enhanced architectures, but this is not required in the baseline DB-CRN model.

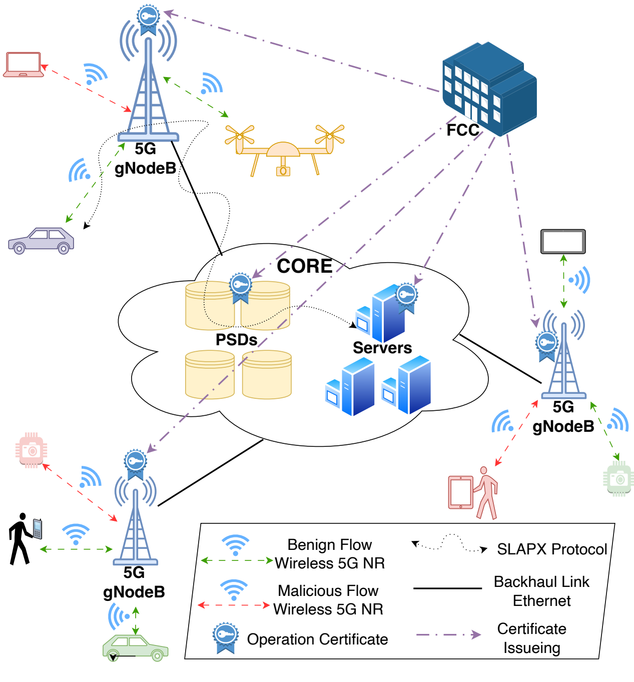


Figure 1: Overview of DB-CRN system architecture

The communication stack follows a standard Internet protocol hierarchy, ensuring seamless interoperability with existing devices and networks. This design enables secure, fine-grained, and location-aware spectrum access without requiring any modifications to underlying access infrastructures or access points. The overall system architecture of DB-CRN is illustrated in Figure 1.

2.2. Cryptographic Building Blocks

$\texttt{SLAPX}$ harnesses various cryptographic components to achieve its privacy-preserving and secure operation goals:

- **Delegatable Anonymous Credentials (*DAC*):** We use an attribute-based DAC scheme [32] that supports multi-level attribute issuance, controlled delegation, and anonymity. The main algorithms are summarized below (see [32]).

Definition 1. A Delegatable Anonymous Credential ($\texttt{DAC}$) scheme is defined by the following algorithms:

- $(pp, \texttt{sk}_{\boldsymbol{RI}}, \texttt{PK}_{\boldsymbol{RI}}) \leftarrow \texttt{Setup}(1^\lambda, 1^t, 1^\eta)$: Given the security parameter λ, a bound t on the committed attribute set size, and a delegation depth parameter $\eta > 1$, it initializes the system by producing public parameters pp and generating the root issuer's key pairs $(\texttt{sk}_{\boldsymbol{RI}}, \texttt{PK}_{\boldsymbol{RI}})$ for each level in η. pp is an implicit input to other algorithms.
- $(\texttt{PK}, \texttt{sk}) \leftarrow \texttt{KeyGen}(pp)$: Generates the user's key pair $(\texttt{PK}, \texttt{sk})$, which $\texttt{PK}$ acts as the user's initial pseudonym.
- $(\texttt{nym}, \texttt{aux}) \leftarrow \texttt{NymGen}(\texttt{PK})$: From the user's $\texttt{PK}$, outputs a fresh pseudonym $\texttt{nym}$ and auxiliary randomness $\texttt{aux}$ to maintain unlinkability in subsequent interactions.
- $\texttt{CreateCred}(L', A, \texttt{sk}_{\boldsymbol{RI}}) \leftrightarrow \texttt{GetCred}(\texttt{PK}_{\boldsymbol{RI}}, \texttt{sk}_u, A) \rightarrow (\texttt{cred}, (\vec{C}, \vec{O}), \texttt{dk}_{L'})$: An interactive protocol between the root issuer (RI) and a user identified by $\texttt{nym}_u$. For the attribute set A, the RI forms a set commitment C and signs it using commitment to create a delegatable credential at $\texttt{PK}_{\boldsymbol{RI}}$. The user receives the credential $\texttt{cred}$, the opening information $\vec{O}$, and a delegation key $\texttt{dk}_{L'}$ for level L'.
- $\texttt{IssueCred}(\texttt{PK}_{\boldsymbol{RI}}, \texttt{dk}_{L'}, \texttt{sk}_u, \texttt{cred}_u, A_l, L'') \leftrightarrow \texttt{ReceiveCred}(\texttt{PK}_{\boldsymbol{RI}}, \texttt{sk}_r, A_l) \rightarrow (\texttt{cred}_r, \texttt{dk}'_{L''})$: It enables controlled delegation from a delegator $\texttt{nym}_i$ to a recipient $\texttt{nym}_r$. Using their secret key $\texttt{sk}_i$, public key of the RI $\texttt{PK}_{\boldsymbol{RI}}$, credential $\texttt{cred}_i$, delegation key $\texttt{dk}_{L'}$, and attribute extension A_l, the delegator generates a new credential $\texttt{cred}_r$ for the recipient. The delegatee, using the secret key $\texttt{sk}_r$, obtains the credential $\texttt{cred}_r$ with the updated attribute set $A' = (A, A_l)$ and delegation depth $L'' \leq L'$, along with a new delegation key $\texttt{dk}'_{L''}$ that supports further delegation if allowed.
- $\texttt{CredProve}(\texttt{PK}_{\boldsymbol{RI}}, \texttt{sk}_p, \texttt{nym}_p, \texttt{aux}_p, \texttt{cred}_p, \boldsymbol{D}) \leftrightarrow \texttt{CredVerify}(\texttt{PK}_{\boldsymbol{RI}}, \texttt{nym}_p, \boldsymbol{D}) \rightarrow \{0, 1\}$: A Zero-Knowledge (ZK) presentation protocol that allows a user to anonymously prove possession of a valid credential. The prover constructs a proof using their secret key $\texttt{sk}_p$, pseudonym $\texttt{nym}_p$, randomness $\texttt{aux}_p$, and credential $\texttt{cred}_p$, revealing only the disclosed attributes $\boldsymbol{D}$. The verifier checks the proof under $\texttt{PK}_{\boldsymbol{RI}}$ and outputs 1 if the credential is valid, and 0 otherwise.

- **Distance Bounding Protocol (DBP):** A DBP is a two-party protocol that verifies the physical proximity of two entities by measuring round-trip delays in a fast challenge–response exchange. We employ a PK-DBP [28] based on a one-pass authenticated key-agreement protocol using nonce Diffie–Hellman to derive a session key between the prover and verifier. This key is then used within the OTDB symmetric-DBP [43].

Definition 2. A public key DBP is defined as follows:

- $(\mathtt{sk}, \mathtt{PK}) \leftarrow \mathtt{KeyGen}(1^{\lambda})$: Generates a public/secret key pair for security parameter λ.
- $ss \leftarrow \mathtt{AKA}(\mathtt{sk}, \mathtt{PK}, \mathtt{PK}')$: The prover and verifier derive a shared session key using their own key pairs and the other party's public key ($\mathtt{PK}'$).
- $\{0, 1\} \leftarrow \mathtt{SymDBP}(ss, th)$: An interactive proximity-test protocol using session key ss and distance threshold th, consisting of: *(i)* `Initialization`: The verifier samples $m \in \{0, 1\}^{2n}$ and sends it to the prover, who computes $a = ss \oplus m$. *(ii)* `Rapid Bit Exchange`: For $i = 1, 2, ..., n$, the verifier sends a challenge $c_i \in \{0, 1\}$ and the prover responds with $r_i = a_{2i+c_i-1}$, while the verifier records the round-trip time $\mathtt{RTT}_i$. *(iii)* `Authentication`: Using $a = ss \oplus m$, the measured delays, and the distance bound, the verifier checks $\mathtt{RTT}_i \leq 2 \times th$ and $r_i = a_{2i+c_i-1}$. It outputs 1 if the prover is within the threshold; otherwise 0.

• **Revocable-iff-linked Linkable Ring Signature (RLRS):** Ring signatures allow a user to anonymously sign on behalf of a group. An RLRS augments this by embedding an event identifier into each signature, enabling detection of multiple signatures produced by the same signer within that context and supporting controlled anonymity revocation when necessary [6].

Definition 3. A Revocable-iff-linked Linkable Ring Signature ($\mathtt{RLRS}$) scheme is a 6-tuple algorithm as shown below:

- $(\mathtt{msk}, pp_{\mathtt{RLRS}}) \leftarrow \mathtt{RLRS.Setup}(1^{\lambda})$: Given λ, generate the master secret key $\mathtt{msk}$ and the public params $pp_{\mathtt{RLRS}}$.
- $(\mathtt{sk}_u) \leftarrow \mathtt{RLRS.Extract}(pp, \mathtt{ID}_u)$: An interactive key-extraction protocol that, given $pp_{\mathtt{RLRS}}$ and a user identifier $\mathtt{ID}_u \in \{0, 1\}^*$, outputs the user secret key $\mathtt{sk}_u$.
- $(\sigma, \tau) \leftarrow \mathtt{RLRS.Sign}(\mathtt{sk}_u, m, \mathcal{L}_{\mathtt{ID}}, e_{\mathtt{ID}})$: To sign a message m under identifier $e_{\mathtt{ID}}$ with ring $\mathcal{L}_{\mathtt{ID}} = \{\mathtt{ID}_1, \dots, \mathtt{ID}_n\}$ ($n \leq t_{\max}$), the signer uses a non-interactive ZK proof to produce a signature σ and link tag τ.
- $\{0, 1\} \leftarrow \mathtt{RLRS.Verify}(\mathcal{L}_{\mathtt{ID}}, m, e_{\mathtt{ID}}, \sigma, \tau)$: Given the ring $\mathcal{L}_{\mathtt{ID}}$, message m, identifier $e_{\mathtt{ID}}$, and signature-tag pair (σ, τ), output 1 if the signature is valid, or else 0.
- $\{0, 1\} \leftarrow \mathtt{RLRS.Link}(\mathcal{L}_{\mathtt{ID}}, e_{\mathtt{ID}}, m, \sigma, \tau, m', \sigma', \tau')$: Determine if two signature-tag pairs (σ, τ) and (σ', τ') on messages m and m' (under event $e_{\mathtt{ID}}$ and ring $\mathcal{L}_{\mathtt{ID}}$) were produced by the same signer. Return 1 if linked, or else 0.
- $(\perp, \mathtt{ID}_u) \leftarrow \mathtt{RLRS.Revoke}()$: Given two linked signatures from events $e_{\mathtt{ID}}$ and rings $\mathcal{L}_{\mathtt{ID}}$ and $\mathcal{L}'_{\mathtt{ID}}$, output the signer's identity $\mathtt{ID}_u \in \mathcal{L}_{\mathtt{ID}} \cap \mathcal{L}'_{\mathtt{ID}}$ if the signatures originate from the same user; output $\perp$ otherwise.

• **Verifiable Delay Function (VDF):** A VDF [8] enforces a prescribed delay by requiring sequential computation, while enabling a fast verification of the result. We use an RSA-based VDF [46], which provides compact proofs and constant-time verification.

Definition 4. A Verifiable Delay Function ($\mathtt{VDF}$) scheme is a 3-tuple algorithm as shown below:

- $(N, H, H_p) \leftarrow \mathtt{VDF.Setup}(1^{\lambda}, \kappa)$: Given λ and difficulty level κ, generate an RSA modulus N of size λ, a hash function $H : \{0, 1\}^* \rightarrow \{0, 1\}^{2\kappa}$, and a prime-deriving hash $H_p(m) = \mathtt{next\text{-}prime}(H(m))$, which returns the closest prime number greater than or equal to $H(m)$.
- $(\ell, \pi) \leftarrow \mathtt{VDF.Eval}(m, \tau)$: On input $m \in \{0, 1\}^*$ and delay parameter $\tau \in \mathbb{N}$, compute $x = H(m)$ and the sequentially run $y = x^{2^{\tau}} \mod N$. Then compute the proof using $\ell = H_p(x + y)$ and $\pi = x^{\lfloor 2^{\tau}/\ell \rfloor}$.
- $\{0, 1\} \leftarrow \mathtt{VDF.Verify}(x, \tau, \ell, \pi)$: Given x, τ, ℓ, π, compute $x = H(m)$, $r = 2^{\tau} \mod \ell$, and $y = \pi^{\ell} \times x^{r} \mod N$. The verifier outputs 1 if $\ell = H_p(x + y)$, or else 0.

3. Threat and Security Models and Attack Scenarios

This section outlines (i) the threat model and scope of our solution, defining the adversary, its capabilities, and the considered attack vectors; (ii) the security objectives and security model that underpin the formal security proof of `SLAPX` as a primitive; and (iii) representative attack scenarios that motivate and guide our network-level simulations when our proposed cryptographic techniques are integrated into the network architecture.

3.1. Threat Model and Scope

Our threat model considers a probabilistic polynomial-time (PPT) adversary with control over the wireless communication channel. This adversary seeks to violate user privacy by targeting confidentiality, identity protection, message authenticity, and integrity during spectrum access, while also attempting to disrupt service through DoS attacks against PSDs and backend servers. In addition, the adversary may exploit the system by submitting fraudulent or manipulated location information. PSDs, and service servers are assumed to be honest-but-curious: they follow the prescribed protocols correctly but may attempt to infer sensitive information, such as user identity or precise location. Under this model, the adversary can mount three attack vectors commonly observed in DB-CRNs, which are captured in our threat assumptions and described in detail below:

- *Client Privacy and Anonymity*: Adversaries such as PSDs, service servers, or external observers may attempt to deduce a user's real-world identity, precise location coordinates, or device attributes by analyzing protocol messages exchanged during spectrum queries and access requests. These efforts can include passive monitoring, active interaction, or correlating metadata to uncover sensitive user-specific information.
- *Location Spoofing Attacks*: Malicious users may attempt to bypass access restrictions by reporting incorrect location information to obtain spectrum resources or services beyond their authorized area. Such misuse can also arise from compromised or spoofed devices. Relevant threats include falsified coordinates, replayed location proofs, collusion with adversarial entities, distance fraud, relay or mafia fraud, and distance hijacking [34].
- *Denial-of-Service Attacks*: Malicious and compromised users or external attackers may attempt to disrupt DB-CRN operations by overwhelming PSDs or service servers during spectrum queries or service requests. Such attacks can involve flooding the system with invalid, replayed,

or resource-intensive requests to exhaust computational capacity, increase response latency, or deny service to legitimate users. These threats are particularly harmful in distributed and delay-sensitive spectrum access environments, where timely and reliable availability is essential.

Scope of Our Solution: The SLAPX framework is designed to preserve user anonymity and location privacy during spectrum access. It protects SUs while they query spectrum availability, obtain cryptographic puzzles, and access network services, mitigating threats such as identity exposure, location spoofing, and DoS attacks. Our work concentrates exclusively on the spectrum query and access phases. Privacy risks arising during user registration or from spectrum usage patterns (e.g., utilization inference) fall outside the scope of this study. Although the framework is tailored for SUs, it can be extended to support PUs interacting with PSD servers; however, PU-related protections are not considered here. Moreover, SLAPX does not address location leakage during ongoing spectrum use, user mobility or handover processes [21], nor does it cover timing-based attacks, side-channel leakage, or signal-based localization methods such as triangulation [7].

3.2. Security Model

Building on the system and threat models, SLAPX is evaluated under the following security definitions:

Definition 5 (Credential Unforgeability). *A DAC scheme cred is* unforgeable *if no PPT adversary can cause an honest verifier to accept a credential proof for attributes it is not entitled to. Technically,* $\Pr[\mathtt{CredVerify}(\mathtt{PK}_{\boldsymbol{RI}}, \mathtt{nym}^{\star}, \boldsymbol{D}^{\star}) = 1 \wedge \boldsymbol{D}^{\star} \nsubseteq A_i] \leq negl(\lambda)$*, where $\mathcal{A}$ may obtain credentials via the CreateCred and IssueCred protocols, $(\mathtt{nym}^{\star}, \boldsymbol{D}^{\star})$ is the adversary's output in a showing attempt, and A_i denotes any attribute set for which $\mathcal{A}$ has legitimately obtained a credential (including delegation).*

Definition 6 (Credential Anonymity). *A DAC scheme cred provides* anonymity *if no PPT adversary can distinguish which of two legitimate credential holders produced a valid credential proof. Technically,* $\Pr[\mathcal{A}^{\mathtt{CredProve}(\mathtt{PK}_{\boldsymbol{RI}})}(i_0, i_1) = 1] \leq \frac{1}{2} + negl(\lambda)$*, where $\mathcal{A}$ is given two distinct user indices (i_0, i_1) corresponding to honest users holding valid credentials, interacts with a verification oracle that executes CredProve and CredVerify on behalf of one of the two users chosen uniformly at random, and attempts to guess which user generated the accepted proof.*

Definition 7 (Credential Unlinkability). *A DAC scheme cred provides* unlinkability *if no PPT adversary can determine whether two valid credential showings originate from the same credential holder. Technically,* $\Pr[\mathcal{A}(\mathtt{cred}_0, \mathtt{cred}_1) = 1] \leq \frac{1}{2} + negl(\lambda)$*, where $\mathtt{cred}_0$ and $\mathtt{cred}_1$ are two accepted protocol credentials generated via CredProve and CredVerify, and $\mathcal{A}$ is given oracle access to credential issuance and verification and attempts to distinguish whether $\mathtt{cred}_0$ and $\mathtt{cred}_1$ were produced by the same user or by two distinct users holding valid credentials.*

Definition 8 (Correctness and Soundness of Location Verification). *A location verification scheme satisfies* correctness *if any honest user physically located at (l_x, l_y) can generate a proof of location PoL that is accepted by the verifier with overwhelming probability, i.e.,* $\Pr[\mathtt{Verify}(\mathtt{PoL}(l_x, l_y)) = 1] \geq 1 - negl(\lambda)$*. The scheme satisfies* soundness *if no PPT adversary $\mathcal{A}$ can produce a proof $\mathtt{PoL}'$ that is accepted as valid for a location $l' \neq l$, except with negligible probability, unless $\mathcal{A}$ is physically present at l', i.e.,* $\Pr[\mathtt{Verify}(\mathtt{PoL}') = 1 \ \wedge \ \mathtt{Loc}(\mathtt{PoL}') = l'] \leq negl(\lambda)$*. Valid proofs are cryptographically bound to their spatio-temporal context, ensuring non-transferability and replay resistance, and thereby preventing relay, distance-fraud, mafia-fraud, and location-hijacking attacks.*

Definition 9 (Counter-DoS). *A system is* DoS-resilient *if no PPT adversary $\mathcal{A}$ issuing at most $q(\lambda)$ requests can cause service unavailability—defined as delaying any honest request beyond a fixed bound Δ—without incurring a proportional computational cost. Technically,* $\Pr[\mathtt{Delay}(\mathcal{A}) > \Delta \ \wedge \ \mathtt{Cost}(\mathcal{A}) < q(\lambda) \cdot \tau] \leq negl(\lambda)$*, where τ denotes the enforced per-request cost determined by VDF evaluation, rate-limiting, and authenticated queries.*

3.3. Attack Scenarios

In addition to the primitive-level security analysis, we consider attack scenarios for DoS and location spoofing at the network-level. These scenarios motivate and guide the network-level security simulations of the SLAPX.

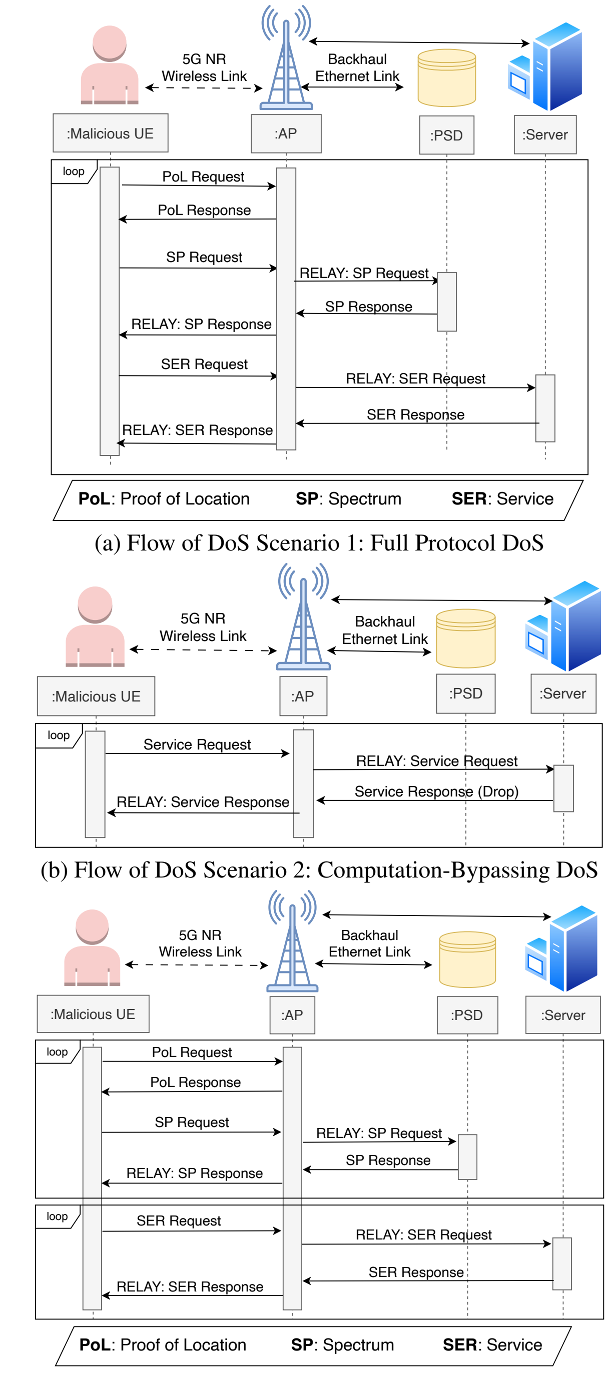
3.3.1. DoS Attack Scenarios

We consider three representative DoS attack scenarios to capture protocol-compliant and protocol-abusive behaviors, as well as high-cost and low-cost adversarial strategies under realistic threat conditions. All scenarios are summarized in Figure 2 and are analyzed in Section 5.2.

1) Full Protocol DoS: The malicious users repeatedly execute the SLAPX algorithms with the goal of exhausting server-side computational and networking resources. While an honest user typically executes the protocol only once to obtain service access, the adversary continuously restarts the protocol to keep the servers busy.

2) Computation-Bypassing DoS: The malicious users flood the server with service requests without executing the prerequisite steps ofSLAPX. As a result, these requests lack valid cryptographic proofs and are discarded upon verification. However, by bypassing expensive client-side computation, adversaries can generate a significantly higher request rate, enabling large-scale flooding attacks that stress server-side filtering and rate-limiting mechanisms.

3) Proof Precomputation DoS: The malicious users execute SLAPX up to the location proof phase and precompute a batch of valid proofs within a limited time window. Since each proof embeds a timestamp and remains valid only for a bounded duration, the attacker must carefully balance the number of precomputed proofs to maximize server load while minimizing proof expiration. This scenario captures a more strategic adversary that exploits temporal validity to amplify attack impact.



(a) Flow of DoS Scenario 1: Full Protocol DoS

(b) Flow of DoS Scenario 2: Computation-Bypassing DoS

(c) Flow of DoS Scenario 3: Proof Precomputation DoS

Figure 2: SLAPX DoS attack scenarios

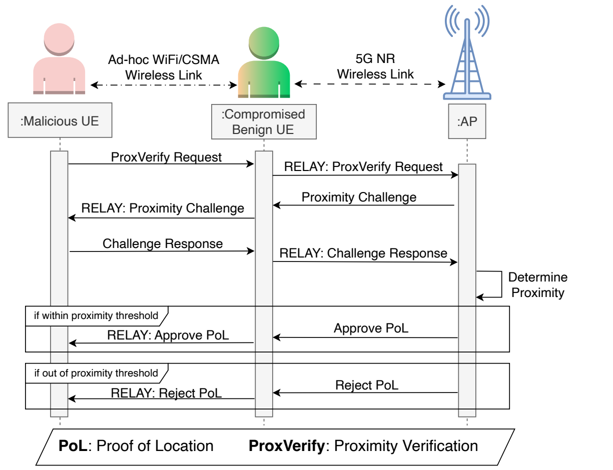


Figure 3: Distance hijacking flow using benign UE as a relay

3.3.2. *Location Spoofing Attack Scenarios*

We consider two representative location spoofing scenarios derived from our location verification algorithms: (i) Distance Hijacking and (ii) Distance Fraud. They capture relay-based and protocol-level proximity violations and are evaluated in Section 5.2.

1) Distance Hijacking: A malicious UE that is physically far from the gNodeB attempts to impersonate a nearby device by relaying messages through a compromised benign UE within legitimate proximity. The benign UE is assumed to be honest but compromised, and thus serves as a relay for all protocol messages exchanged between the malicious UE and the gNodeB. By leveraging the strong received signal strength (RSS) of the compromised UE, the attacker aims to deceive the gNodeB into inferring a shorter distance than the malicious UE's true location. However, because the relayed communication introduces additional propagation delay, the measured round-trip time (RTT) reflects the true distance between the malicious UE and the gNodeB. This scenario allows us to study the relative effectiveness of RSS- versus RTT-based distance estimation and assess the system's resistance to relay-based proximity attacks. The message flow of this attack is shown in Figure 3.

2) Distance Fraud: No access point or trusted infrastructure is available in the vicinity, and proximity verification relies on device-to-device DBPs with nearby nodes. A malicious UE that is physically distant attempts to authenticate itself as nearby by manipulating the DBP execution. Since the DBP enforces proximity using both RTT constraints and challenge–response correctness, the adversary cannot reduce propagation delay below physical limits. Instead, the malicious UE attempts to guess the verifier's challenges in advance and transmit corresponding responses prematurely, with the goal of appearing closer than it actually is. This scenario captures a protocol-level spoofing attack that does not rely on relaying, but instead exploits the probabilistic nature of challenge-response mechanisms under strict timing constraints.

4. SLAPX: Design and Instantiation

This section outlines the key enhancements of SLAPX over the original SLAP protocol, and then presents the system setup, core operations, and detailed algorithmic specification of the SLAPX framework.

Improvements over SLAP Framework and the SOTA: SLAPX extends our SLAP framework [15], incorporating substantial new material and the following key enhancements and additional features: *(i) Comprehensive DoS countermeasures:* We significantly strengthen DoS resilience by introducing on-demand, device-specific, and non-parallelizable VDFs, authenticated using lightweight standard signatures (e.g., ECDSA). This defense is further reinforced by a rate-limiting mechanism based on the linkability of event-oriented RLRSs, enabling effective DoS mitigation for both geolocation databases and service servers. *(ii) Enhanced Proof of Location:* We enhance the location proof mechanism by leveraging RLRS-based event linkability to enforce rate limiting on proof issuance and enable revocation of APs that collude with malicious users. This provides accountability and supports more realistic adversarial network settings. *(iii) Comprehensive performance evaluation and network simulations:* We substantially expand the evaluation by including detailed cryptographic benchmarking, comparisons with SOTA, and extensive network simulations that model real-world DoS and location spoofing scenarios under practical deployment conditions.

SLAPX advances the state of the art in DB-CRN security by providing the first unified, regulation-compliant framework that simultaneously achieves anonymous and location-private spectrum access, verifiable location assurance, and proactive DoS resilience under realistic adversarial conditions. Unlike prior solutions that address these challenges in isolation or rely on heavy cryptography, multiple non-colluding databases, or trusted infrastructure, SLAPX integrates delegatable anonymous credentials, adaptive location proofs, VDF-enforced rate control, and RLRS-based accountability into a single lightweight architecture compatible with existing DB-CRN deployments. Compared to existing schemes, SLAPX operates with a single geolocation database, incurs orders-of-magnitude lower communication overhead, and achieves substantially lower end-to-end latency, while uniquely preserving anonymity even against access points and databases.

4.1. SLAPX Framework Initial Setup

DB Structure: The DBs are organized in compliance with FCC spectrum-sharing regulations and maintain synchronized views of spectrum availability as required by the FCC. Each DB entry is indexed by location coordinates at a predetermined spatial resolution (e.g., 50 m), along with device characteristics (e.g., transmit power, device type) and access constraints (e.g., bandwidth/duration). This structured indexing ensures consistency across synchronized databases. Each record includes spectrum availability data, a validity time window, and operational parameters such as the maximum allowable transmission power (EIRP).

Access Point Setup: Access Points (APs) within a geographical region jointly form a ring ($\mathcal{L}_{\mathtt{ID}}$) that contains the $\mathtt{ID}$ of all participating APs. Each AP is provisioned with a cryptographic key pair $(\mathtt{sk}_{\mathtt{AP}}, \mathtt{PK}_{\mathtt{AP}})$, generated by the FCC using the $\mathtt{RLRS.Extract}(pp, \mathtt{ID}_{\mathtt{AP}})$ algorithm (Definition 3). APs periodically broadcast time-synchronized beacons $\beta_{\mathtt{TS}}$ within predefined time windows $\mathtt{TS}$ to enable nearby device discovery. To verify UE proximity, each AP performs signal-based ranging using physical-layer measurements, including received signal strength ($\mathtt{RSS}$) and round-trip time ($\mathtt{RTT}$) [3]. These measurements are processed via $\Delta \leftarrow \mathtt{ProxVerify}(\mathtt{RSS}, \mathtt{RTT}, env_{\text{params}})$, which incorporates environment-specific factors such as path-loss models, multipath effects, and channel conditions to estimate the UE–AP distance and associated confidence bounds [31]. The resulting estimate Δ determines whether the UE lies within the AP's trusted coverage region, providing a confidence interval that underpins subsequent authenticated access decisions. Additional details are provided in [38], with simulation results reported in Section 5.

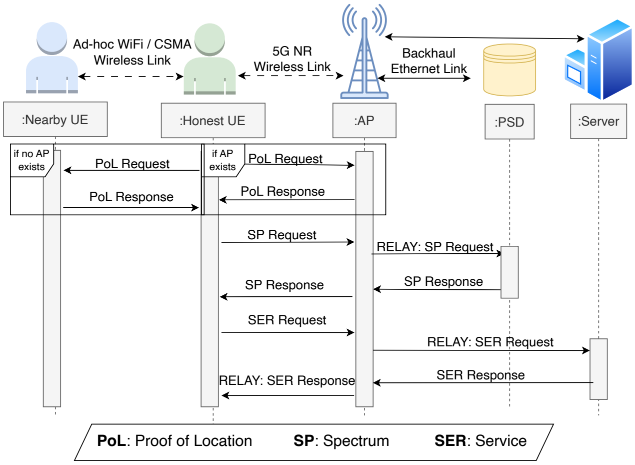


Figure 4: High-level flow of the SLAPX protocol

System Setup: In the system initialization, the FCC, acting as the core network authority, serves as the root issuer for all participants. For each registered UE, the FCC certifies a set of device attributes (e.g., identifier, transmit power) and access policies (e.g., validity period) by issuing a level-1 root credential via $\mathtt{CreateCred}(L', A, \mathtt{sk}_{\mathtt{FCC}})$. A UE, represented by a pseudonym $\mathtt{nym}_c$, retrieves it through $\mathtt{GetCred}(\mathtt{PK}_{\mathtt{FCC}}, \mathtt{sk}_c, A)$, obtaining $(\mathtt{cred}, (\vec{C}, \vec{O}), \mathtt{dk}_{L'})$. The issued credential $\mathtt{cred}$ binds the attribute commitment $\vec{C}$ to the FCC's public key $\mathtt{PK}_{\mathtt{FCC}}$, includes the corresponding opening information $\vec{O}$, and embeds a delegation key $\mathtt{dk}_{L'}$ that permits controlled delegation up to level L'. This credential allows UEs to refresh their pseudonyms or delegate them to another party by transitioning to a new public key and, if needed, augmenting the attribute set to A'. Credential usage requires the UE to prove knowledge of the associated secret key $\mathtt{sk}_c$ and to generate a randomized signature over the relevant attributes.

4.2. SLAPX Framework Main Operations

The high-level flow of the SLAPX framework is depicted in Figure 4. Moreover, a more detailed look at the internal steps of the protocol is provided in Algorithms 1, 2, and 3.

4.2.1. Proof of Location Acquisition and Validation

Before querying PSDs or accessing network services, a UE must acquire a valid proof of location ($\mathtt{PoL}$). This phase establishes a $\mathtt{PoL}$ bound to a specific geographic region and time window. The process accommodates two complementary scenarios: infrastructure-rich settings with an available nearby AP, and resource-constrained or rural environments where no AP is present.

(i) AP-Assisted $\mathtt{PoL}$ Generation: When a nearby AP is available, the client obtains a $\mathtt{PoL}$ following Algorithm 1. After receiving periodic beacons ($\beta_{\mathtt{TS}}$) from surrounding

APs, the client selects the AP with the strongest signal and requests a PoL for the current time window TS. The client discloses the received beacon, its location coordinates, and the current timestamp as credential attributes and proves them anonymously to the AP. Upon validating the client's credentials ($\texttt{nym}_c, \texttt{cred}_c$) under the FCC's public key, the AP assesses the client's physical proximity using signal strength and RTT measurements via $\texttt{ProxVerify}(\cdot)$. If the measured distance satisfies the predefined threshold, the AP binds the credential to the disclosed attributes, derives an event identifier, and generates an RLRS signature over the client's attributes and credential (Steps 6–10). The signed tuple is returned to the client, who verifies the signature and accepts it as a valid proof of location Φ (Steps 11–12).

```
Algorithm 1 $\Phi \leftarrow \texttt{PoL.AP}(\texttt{TS}, (l_x, l_y), \texttt{cred}_c, \beta_{\texttt{TS}})$

❶ $D_{\texttt{TS}} \leftarrow \texttt{Client.PoL.Request}(\beta_{\texttt{TS}})$:
 1: Given the latest beacon $\beta_{\texttt{TS}}$
 2: Set $D_{\texttt{TS}} \leftarrow (\beta_{\texttt{TS}}, (l_x, l_y), \texttt{TS})$
 3: Perform $\texttt{CredProve}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, \texttt{nym}_c, \texttt{aux}_c, \texttt{cred}_c, D_{\texttt{TS}})$
❷ $(\mathbf{m}, \sigma_{\texttt{AP}}, \tau_{\texttt{AP}}) \leftarrow \texttt{AP.PoL.Respond}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, \texttt{cred}_c, D_{\texttt{TS}})$:
 4: if $1 \leftarrow \texttt{AP.CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, D_{\texttt{TS}})$, then
 5:     $\Delta_c \leftarrow \texttt{AP.ProxVerify}(\texttt{RSS}, \texttt{RTT}, env_{param})$
 6:     if $(l_x, l_y) \in \Delta_c$, then
 7:         Set $\mathbf{m} \leftarrow \{D_{\texttt{TS}}, \texttt{nym}_c, \texttt{cred}_c\}$
 8:         Set $e_{\texttt{ID}} \leftarrow \{(l_x, l_y), \texttt{TS}, \beta_{\texttt{TS}}\}$
 9:         $(\sigma_{\texttt{AP}}, \tau_{\texttt{AP}}) \leftarrow \texttt{RLRS.Sign}(\texttt{sk}_{\texttt{AP}}, \mathbf{m}, \mathcal{L}_{\texttt{ID}}, e_{\texttt{ID}})$
10:         Send $(\mathbf{m}, \sigma_{\texttt{AP}}, \tau_{\texttt{AP}})$ to the Client.
❸ $\Phi \leftarrow \texttt{Client.PoL.Verify}(\mathbf{m}, \sigma_{\texttt{AP}}, \tau_{\texttt{AP}})$:
11: if $\{0, 1\} \leftarrow \texttt{RLRS.Verify}(\mathcal{L}_{\texttt{ID}}, \mathbf{m}, e_{\texttt{ID}}, \sigma_{\texttt{AP}}, \tau_{\texttt{AP}})$, then
12:     $\Phi \leftarrow (\mathbf{m}, \sigma_{\texttt{AP}}, \tau_{\texttt{AP}})$
```

(ii) Device-Assisted PoL Generation: In environments with limited infrastructure, such as rural areas without nearby WiFi APs or cellular coverage, the client relies on neighboring devices (NDs) to obtain a PoL and anonymous credentials, as described in Algorithm 2. The client initiates this process by broadcasting a PoL request that includes the current time and claimed location (Step 1). Upon receiving replies, the client designates its location coordinates and timestamp as the attributes to be selectively disclosed and proves them anonymously to the responding ND. Using the FCC's public key, the ND verifies the client's credential and, upon successful validation, establishes a shared session key ss through an interactive authenticated key agreement. The ND then executes a symmetric-DBP to confirm the client's proximity within a predefined threshold th (Steps 4–6). If the proximity check succeeds, the ND augments its attribute set with the client's location (l_x, l_y) and time window TS, and issues a delegated credential with restricted delegation rights (Steps 7–10). Finally, the client retrieves the delegated credential and corresponding location proof, authenticated under the FCC's public key. Thereby, obtaining a valid PoL embedded within the extended attributes (Steps 11–13).

```
Algorithm 2 $(\texttt{cred}'_c, A') \leftarrow \texttt{PoL.ND}(\texttt{TS}, (l_x, l_y), \texttt{cred}_c)$

❶ $D_{\texttt{TS}} \leftarrow \texttt{Client.PoL.Request}(\texttt{nym}_c, \texttt{cred}_c)$:
 1: Request PoL and a delegated credential $\texttt{cred}'_c$ from an ND
 2: Set $D_{\texttt{TS}} \leftarrow ((l_x, l_y), \texttt{TS})$
 3: Perform $\texttt{CredProve}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, \texttt{nym}_c, \texttt{aux}_c, \texttt{cred}_c, D_{\texttt{TS}})$
❷ $D_{\texttt{TS}} \leftarrow \texttt{ND.PoL.Respond}(\beta_{\texttt{TS}})$:
 4: if $1 \leftarrow \texttt{ND.CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, D_{\texttt{TS}})$, then
 5:     Perform $ss \leftarrow \texttt{AKA}(\texttt{sk}_{\texttt{ND}}, \texttt{PK}_{\texttt{ND}}, \texttt{PK}_c)$
 6:     if $1 \leftarrow \texttt{SymDBP}(ss, th)$ and $(l_x, l_y) \in th$, then
 7:         Set $A_l \leftarrow D = ((l_x, l_y), \texttt{TS})$
 8:         Set $dk'_{L''} := \perp$
 9:         $\texttt{IssueCred}(\texttt{PK}_{\texttt{FCC}}, \texttt{dk}_{L'}, \texttt{sk}_{\texttt{ND}}, \texttt{cred}_{\texttt{ND}}, A_l, L'')$
10: Send $(\texttt{cred}'_c, \texttt{dk}'_{L''})$ to the client.
❸ $D_{\texttt{TS}} \leftarrow \texttt{Client.PoL.Verify}(\beta_{\texttt{TS}})$:
11: $(\texttt{cred}'_c, \texttt{dk}'_{L''}) \leftarrow \texttt{ReceiveCred}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, A_l)$
12: Set $A' \leftarrow (A, \Phi)$ where $\Phi \leftarrow A_l$
13: $(\texttt{cred}'_c, A')$
```

4.2.2. *Querying Spectrum Availability and Services*

Algorithm 3 describes the procedure for requesting spectrum availability and related services. Each query must be preceded by obtaining a valid PoL bound to the client's coordinates and the current timestamp. In the *AP-assisted* PoL case, the client initiates the request by anonymously proving its credential while disclosing its location, time window TS, and the obtained location proof Φ, and then submits a spectrum query for the surrounding area (Steps 1–5). The PSD first validates the client's credential and then verifies the RLRS signature issued by the AP. Using the RLRS-based PoL within the current time interval, the AP checks the linkability of the proof against other signatures to detect misuse such as DoS and replay (Steps 12–14). Upon successful verification and based on the disclosed device attributes (e.g., transmit power and device type), the PSD generates a VDF with an appropriate difficulty level and signs it using SGN (Steps 10-14). In the *device-assisted* PoL case, the client relies on a delegated credential that embeds the location, timestamp TS, and Φ as disclosed attributes. The client proves this credential and submits a spectrum query in the same manner. After validating the credential, the PSD follows an identical process by issuing a VDF with the corresponding difficulty and signing it via SGN, then returns the spectrum availability information together with the VDF and its signature to the client.

4.2.3. *Notifying Spectrum Usage and Service Requests*

To submit spectrum usage reports or request network services, the client first evaluates the previously issued VDF through sequential computation and derives a communication token bound to the message m, which encodes either usage information or a service request (Steps 19–20). The client then presents this token and anonymously proves possession of a valid credential to the server (or equivalently to the PSD) to initiate the request. Upon successful verification of the credential and validation of the VDF, the server (or PSD) checks the associated proof of location according to whether it was generated via the AP-assisted

Algorithm 3 SLAPX Scheme

```
❶ $\rho_c \leftarrow \texttt{Client.Query}(\texttt{cred}_c, \Phi_c, (l_x, l_y), \texttt{TS}, freq)$:
Given $(l_x, l_y)$ and TS, client request PoL:
If $\Phi_c \leftarrow \texttt{Client.PoL.AP}((l_x, l_y), \texttt{TS}, \texttt{cred}_c, \beta_{\texttt{TS}})$
    Set $D_c \leftarrow ((l_x, l_y), \texttt{TV}, ch, \Phi)$
    Perform $\texttt{CredProve}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, \texttt{nym}_c, \texttt{aux}_c, \texttt{cred}_c, \boldsymbol{D}_c)$
    Query a PSD for $\rho_c \leftarrow ((l_x, l_y), ch, \texttt{TV})$
elseif $(\texttt{cred}'_c, A') \leftarrow \texttt{Client.PoL.ND}(\texttt{cred}_c, (l_x, l_y), \texttt{TS})$
    Given $\texttt{cred}_c \leftarrow \texttt{cred}'_c$ with $A \leftarrow (A, \Phi)$ for $((l_x, l_y), \texttt{TV})$
    Perform $\texttt{CredProve}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, \texttt{nym}_c, \texttt{aux}_c, \texttt{cred}_c, A')$
    Query a PSD for $\rho_c \leftarrow ((l_x, l_y), ch, \texttt{TV})$

❷ $R_{\texttt{PSD}} \leftarrow \texttt{PSD.Respond}((l_x, l_y), ch, \texttt{TV})$:
If $1 \leftarrow \texttt{CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, \boldsymbol{D})$
    If $1 \leftarrow \texttt{RLRS.Verify}(\texttt{vk}, \mathbf{m} = (\boldsymbol{D}, \texttt{nym}_c, \texttt{cred}_c), \sigma_{\texttt{AP}})$
        for all $(m', \sigma', \tau') \in \texttt{TS}$ do
            If $1 \leftarrow \texttt{RLRS.Link}(\mathcal{L}_{\texttt{ID}}, e_{\texttt{ID}}, m, \sigma, \tau, m', \sigma', \tau')$
                Return $\perp$
            else
                Set $\Pi \leftarrow \texttt{VDF.Setup}(1^\lambda, \kappa)$ accordingly
                $\sigma_{\pi_\theta} \leftarrow \texttt{SGN.Sign}(\texttt{sk}_{\texttt{PSD}}, \pi_\theta)$
                Return $R_{\texttt{PSD}} \leftarrow (\Pi, \sigma_{\pi_\theta}, freq)$
elseif $1 \leftarrow \texttt{CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, A')$
    Set $\Pi \leftarrow \texttt{VDF.Setup}(1^\lambda, \kappa)$ accordingly
        $\sigma_{\pi_\theta} \leftarrow \texttt{SGN.Sign}(\texttt{sk}_{\texttt{PSD}}, \pi_\theta)$
    Return $R_{\texttt{PSD}} \leftarrow (\Pi, \sigma_{\pi_\theta}, freq)$

❸ $\{0, 1\} \leftarrow \texttt{Client.Service.Request}(\Phi, (l_x, l_y), \texttt{TS})$:
Notifying spectrum usage or requesting services to Servers:
Given the request/notification message as $m$
$(\ell, \psi) \leftarrow \texttt{VDF.Eval}(m, \tau)$
Performs $\texttt{CredProve}(\texttt{PK}_{\texttt{FCC}}, \texttt{sk}_c, \texttt{nym}_c, \texttt{aux}_c, \texttt{cred}_c, \boldsymbol{D}_c)$
Send $(m, \psi)$ to PSD/Server

❹ $\{0, 1\} \leftarrow \texttt{Server.Respond}(m, \psi, )$:
If $1 \leftarrow \texttt{CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, \boldsymbol{D})$
    If $1 \leftarrow \texttt{SGN.Verify}(\pi_\theta, \sigma_{\pi_\theta})$
        If $1 \leftarrow \texttt{RLRS.Verify}(\texttt{vk}, \mathbf{m} = (\boldsymbol{D}, \texttt{nym}_c, \texttt{cred}_c), \sigma_{\texttt{AP}})$
            PSD/Server Return 1, and Grant Services.
elseif $1 \leftarrow \texttt{CredVerify}(\texttt{PK}_{\texttt{FCC}}, \texttt{nym}_c, A')$
    If $1 \leftarrow \texttt{SGN.Verify}(\pi_\theta, \sigma_{\pi_\theta})$
PSD/Server Return 1, and Grant Services.
```

or device-assisted PoL process. If all checks pass, the PSD records the reported spectrum usage, or the server authorizes access to the requested services. Distinct from prior approaches, spectrum usage reporting in SLAPX is also protected by anonymous credentials with attribute binding, enabling higher-quality usage information while remaining compliant with FCC coexistence requirements.

4.3. Security Analysis

Based on the established threat model, we formalize the security model and present the corresponding security analysis. We state the main security theorem for SLAPX, while the complete security model, formal definitions, and detailed proofs are provided in Appendix A.1.

Theorem 1. *The SLAPX framework achieves the following security guarantees: (i) anonymous user authentication, ensured by the anonymity, soundness, and unforgeability of the underlying ZKPoK and SPSEQ-UC signature schemes; (ii) location privacy, provided by the unlinkability of credential-based signatures; (iii) verifiable user location, enforced through either the unforgeability of RLRS combined with signal-based proximity measurements or the security of public-key distance-bounding protocols and anonymous credential delegation; and (iv) resilience to DoS attacks, enabled by VDF and rate-limiting mechanisms.*

Additionally, Section 5.2 presents a network-level security analysis of SLAPX, including simulation-based evaluations of its resilience to various DoS and location spoofing attack scenarios in wireless network settings.

5. Performance Evaluation

This section provides a comprehensive evaluation of the SLAPX framework, covering both cryptographic primitive-level and network-level performance analysis.

5.1. Primitive-Level Evaluation

We evaluate the primitive-level performance of SLAPX by first describing the evaluation metrics and implementation setup, followed by an empirical analysis of the framework and a comparison with state-of-the-art schemes.

5.1.1. Configuration and Experimental Setup

Hardware: We evaluate the efficiency of SLAPX on a standard desktop platform equipped with an Intel Core *i9-11900K* @3.50 GHz, 64 GiB RAM, 1 TB SSD running Ubuntu 22.04.4 LTS.

Libraries: The implementation spans multiple languages (including C, Python, and Java) and leverages established cryptographic libraries and tools such as *DAC-from-EQS*[2], *Wesolowski-VDF*[3], *RELIC*[4], and *OpenSSL*[5], to realize the core components of SLAPX and its underlying primitives (e.g., hashing, modular arithmetic, and exponentiation).

Configuration and Parameter Selection: We exclude signal transmission and network communication delays from our analysis, as these are typically constant-time operations and occur at the microsecond scale. Our configuration employs SHA-256 for hashing, structure-preserving set commitments [20], SPSEQ-UC signatures [32], and the BN-256 elliptic curve (EC) for binding and EC operations (with $n = 2048$). Anonymous authentication is realized via Schnorr-type zero-knowledge proofs augmented with Damgård's technique, following the DAC construction [32]. Non-interactive proofs are instantiated via the Fiat–Shamir transform, yielding an effective security level of approximately 100 bits. The SGN digital signature is instantiated using ECDSA, a widely adopted and computationally efficient standard signature scheme.

Evaluation Metrics and Rationale: We empirically analyze the SLAPX framework by measuring cryptographic, computational, and communication overhead across all operational phases and underlying primitives, including DAC,

[2] https://github.com/mir-omid/DAC-from-EQS
[3] https://github.com/futexor/Wesolowski-VDF
[4] https://github.com/relic-toolkit/relic
[5] https://www.openssl.org/

Table 1
Computational and communication overhead of `SLAPX`

Phase	Entity	Computational Cost (*ms*)	Communication Overhead (*B*)
PoL.AP	*Client*	32.2	2456
	AP	75.79	
PoL.ND	*Client*	31.8	1944
	ND	77.05	
Spectrum Query	*Client*	17.23	3016
	PSD	74.34	
Service Request	*Client*	$\kappa \times S_q + 17.23$	2712
	Server	77.68	

Note: S_q and κ represent the repeated squaring time to solve a puzzle and puzzle difficulty, respectively.

RLRS, DBP, and VDF. Since no existing solution provides an equivalent combination of functionalities, direct performance comparisons are not applicable. Instead, we present a comprehensive performance breakdown of `SLAPX` along key metrics to assess its practicality and deployability. We further include a qualitative and analytical comparison with representative schemes that address partial aspects of these functionalities in the DB-CRN setting. The evaluation proceeds as outlined below.

5.1.2. Experimental Results

The empirical evaluation of cryptographic overhead, computational costs, and communication overhead for each phase of the `SLAPX` framework is summarized in Table 1 and detailed below:

Cryptographic Overhead: To demonstrate credential ownership, the user re-randomizes both the credential `cred` and pseudonym `nym`, and executes a ZKPoK over the secret key `sk` and auxiliary randomness `aux`. This process yields a fresh, unlinkable pseudonym and selectively reveals an attribute subset D via a set commitment construction. In practice, the dominant runtime overhead arises from signature conversion, representation normalization, and rebinding to a new set commitment, which require approximately 2 ms, 5 ms, and 13 ms, respectively. In the adopted PK-DBP, the interactive AKA phase incurs one ECC multiplication (0.61 *ms*), a single hash evaluation (0.35 *ms*), and random value generation (0.05 *ms*). The subsequent rapid bit-exchange phase operates at nanosecond granularity; the resulting distance-fraud error margin (on the order of $1 - 10$ *m*) is negligible relative to the remaining protocol costs. Proximity verification at the AP, implemented via RSS and RTT measurements in `ProxVerify`$(\cdot)$, typically completes within 1λ10 *ms* depending on client distance and environmental conditions. When instantiated using ECDSA, the `SGN` operation requires about 0.3 *ms* for signature generation and 3.8 *ms* for verification. For the RLRS component, signing, verification, linking, and revocation require approximately 15.8 *ms*l, 14.98 *ms*, 0.12 *ms*, and 0.6 *ms*, respectively. These costs can be further reduced at the PSD through batch verification and parallelized linking. Finally, solving a VDF on commodity hardware incurs increasing latency with higher difficulty parameters κ: about 17 *ms* for 10^3 squarings, 61 *ms* for 5×10^3, 121 *ms* for 10^4, 874 *ms* for 8×10^4, and 3.17 *s* for 3×10^5. In contrast, VDF setup and verification remain constant-time, averaging roughly 89 *ms* and 4 *ms*, respectively, across all difficulty levels.

Computational Costs: *(i)* `PoL.AP` *Phase:* The client proves its credential $\texttt{cred}_c$ and verifies the RLRS signature, while the AP validates the credential, executes `ProxVerify`, and issues an RLRS signature as the location proof. *(ii)* `PoL.ND` *Phase:* Two users engage in interactive protocols that include credential proving and verification, key agreement AKA, symmetric-DBP, and credential delegation with the location proof embedded as a certified attribute. *(iii)* `Query` *Phase*: The client submits a valid `PoL` (from either an AP or an ND) to query the PSD for spectrum availability at a given location and time. The PSD verifies the location proof and linkability (via RLRS or $\texttt{cred}_c$), then returns spectrum data together with a user-specific VDF. *(iv)* `Service.Request` *Phase:* The client evaluates the VDF, proves possession of a valid credential, and submits the result. The server (or PSD) verifies the solution before authorizing service access or updating the `DB`.

Communication Overhead: We consider groups of size $|\mathbb{G}_1| = |\mathbb{Z}_p| = 256$ bits, $|\mathbb{G}_2| = 512$ bits, and $|\mathbb{G}_T| = 3072$ bits, with arithmetic performed modulo a 2048-bit integer. Message payloads satisfy $|m| < 256$ B, timestamps occupy 8 B on a 64-bit Unix system, precise location coordinates require 16 B, and spectrum availability data β, derived from FCC records, is approximately 560 B. All attributes in the system are assumed to have uniform size. Credentials encapsulate the values $|\texttt{cred}|$, $|\texttt{sk}|$, and $|\texttt{nym}|$ within set commitments and SPSEQ-UC signatures, yielding a constant-size representation that does not depend on the number of attributes. Specifically, the credential size is $4|\mathbb{G}_1| + |\mathbb{G}_2| + |\mathbb{Z}_p|$, totaling 1792 bits. The size of the commitment vector $\overrightarrow{C}$ scales with the delegation depth (here $L = 2$), resulting in communication overhead that grows linearly with both the number of attributes and delegation levels. The RLRS signature is independent of the number of APs and contains $\sigma = 2|G_1| + 3G_1 + 15Z_p = 640$ bytes. Using publicly available FCC database records[6], we estimate each spectrum database entry to be roughly 560 bytes, augmented with synthetic data for evaluation. The aggregate communication costs across all protocol phases are summarized in Table 1.

Comparison with SOTA: We present a qualitative and analytical comparison of the functionality provided by `SLAPX` against representative location-privacy schemes, summarized in Table 2. To ensure a balanced assessment, we evaluate spectrum query overhead from both the client and PSD perspectives, overall communication costs, and the end-to-end latency required to retrieve a single geolocation database record as an indicator of system scalability. As reflected in Table 2, `SLAPX` uniquely supports the full set of security and privacy requirements for anonymous spectrum

[6] `https://enterprisefiling.fcc.gov/dataentry/public/tv/lmsDatabase.html`

Table 2
Qualitative and analytical comparison with existing location privacy schemes

Scheme	Features					Delay				Total
	Setting	Loc.Privacy	Anonym	Loc.Verification	Counter-DoS	SU	PSD	E2E	PoL	Communication
Troja et al [41]	*1-DB*	*Peer-to-Peer*	✓	✗	✗	1650 *ms*	11760 *ms*	13410 *ms*	✗	12 *MB*
Li et al [29]	*1-DB*	*Pseudo-ID*	✗	*WiFi AP+Loc.Server*	✗	✗	✗	✗	210 *ms*	✗
Xin et al [47]	*1-DB*	*PIR*	✗	*WiFi AP+QRA*	✗	292.8 *ms*	142.7 *ms*	407.4 *ms*	430.1 *ms*	325*KB*
LP-Chor [23]	ℓ*-DB*	*PIR*	✗	✗	✗	7.7 *ms*	480 *ms*	620 *ms*	✗	753 *KB*
LP-Goldberg [23]	ℓ*-DB*	*PIR*	✗	✗	✗	320 *ms*	1210 *ms*	1780 *ms*	✗	6 *MB*
RAID-LP-Chor [23]	ℓ*-DB*	*PIR*	✗	✗	✗	0.4 *ms*	22 *ms*	210 *ms*	✗	125 *KB*
Zeng et al [48]	*1-DB*	*BS+ECC*	*PseudoID*	✗	✗	87 *ms*	27 *ms*	135 *ms*	✗	1.24 *KB*
TrustSAS [24]	ℓ*-DB*	*PIR*	*EPID*	✗	✗	329.4 *ms*	324.6 *ms*	4954 *ms*	✗	1.25 *MB*
PACDoSQ [14]	ℓ*-DB*	*PIR*	*Tor*	✗	*HBP*	28.1 *ms*	199 *ms*	1373.6 *ms*	✗	605.92 *KB*
SLAPX	*1-DB*	*DAC*	*DAC*	*WiFi-AP+RLRS* *DBP+DAC*	*VDF* *Rate-Limiting*	17.23 *ms*	74.34 *ms*	78.61 *ms*	107.99 *ms* 108.85 *ms*	3.72*KB*

Libraries: Virtual Machines running Ubuntu simulated PIR costs, using the *percy++* library[7] for multi-server PIR, the *Open Quantum-Safe* library[8] for PQC primitives, and *OpenSSL* for cryptographic operations and arithmetic. **Variables:** We consider six databases for multi-DB schemes with $|DB| = 560$ *MB* and 400 rows/columns as described in [47]. Key terms include *BS* (base station), *HBP* (hash-based puzzles), *QRA* (quadratic residue assumption), and *EPID* (enhanced privacy ID based on direct anonymous attestation).

access, while enabling flexible deployment options for location verification and achieving low latency with minimal communication overhead.

5.2. Network-Level Evaluation

This section presents an instantiation of the SLAPX protocol in a realistic wireless network setting and evaluates its performance through simulations. We use ns3 [1] to model the deployment of SLAPX and analyze its communication overhead, while the 5G NR (New Radio) LENA module [37] is employed to simulate the 5G infrastructure.

5.2.1. Simulation Setup and Configuration

Overview of the ns3 Setup: We assume that the regulatory authority (e.g., the FCC) has already provisioned all required certificates, keys, and credentials. As a result, the FCC is not explicitly modeled in the ns3 environment. The remaining entities in the SLAPX framework are mapped as follows: the client is modeled as a 5G NR User Equipment (UE), the AP as a 5G gNodeB, and the PSD as a server node. Furthermore, servers that provide cloud services are modeled as one single server node. In this setup, each UE operates within the radio access network (RAN) and connects to the nearest gNodeB over a wireless channel using 5G NR links. Server nodes are deployed in the core network and are connected to the gNodeBs via wired backhaul links.

5G Parameter Configuration: The 5G NR LENA module offers a wide range of configuration options for realistic 5G simulations. For simplicity, we retain default settings for most parameters and manually configure only a selected subset. The configured parameters and their values are summarized in Table 3. Specifically, the channel model follows the standard NR specification [2] used by default in the LENA module. The transmission power of both UE and gNodeB nodes is set to 30 dBm, reflecting typical deployment conditions. The Maximum Transmission Unit (MTU) is set to the standard Ethernet frame size of 1500 bytes, and the backhaul network is modeled with a data rate of 1 Gbps and an end-to-end latency of 2 ms.

Table 3
Simulation parameters in the ns3 experiments

Parameter	Value
Channel Model	3GPP TR 38.901 (default)
UE Tx Power	30 dBm
gNodeB Tx Power	30 dBm
Number of cells per gNodeB	3
Beam width	50 degrees
Backhaul data rate	1 Gbps
Backhaul delay	2 ms
MTU (DoS)	1500 bytes
MTU (Fragmentation)	576-9000 bytes

Server Modeling: Because ns3 is a discrete-event simulator, directly modeling real-time server overload caused by DoS attacks is nontrivial. While network-level congestion can be simulated, capturing the impact of DoS attacks on server-side processing requires an abstracted model. Therefore, each server is represented using a queue-and-worker abstraction, a common approach in analytical DoS modeling [44]. Specifically, the server consists of two components: *(i) Workers*, which model independent computational units (e.g., CPU cores) capable of processing requests in parallel; and *(ii) Queue*, which buffers incoming requests when all workers are busy. However, when the queue is full, incoming requests are dropped. This model enables us to capture server saturation and request loss under DoS conditions, serving as the basis for evaluating DoS scenarios in our simulations.

Terminology: To present the evaluation results clearly and concisely, we use a set of symbols to represent the performance metrics and experiment parameters used in our DoS scenarios, as shown in Table 4.

5.2.2. Simulation Results

Fragmentation Analysis of SLAPX: Based on a realistic 5G deployment, we evaluate SLAPX through numerical analysis focusing on fragmentation behavior and communication overhead. Due to the relatively large size of certain protocol messages, packet fragmentation becomes

Table 4
Symbols used in the scope of DoS experiments

Symbol	Description
N_U	Total number of UEs spawned
N_Q	Total number of messages that are queued
N_D	Total number of dropped benign messages due to busy workers and full queue capacity
t_Q	The average waiting time of a message in the queue (milliseconds)
r_{mal}	The ratio of malicious UEs to the total number of UEs

observable as illustrated in Figure 5, where the upper illustration shows the relationship between MTU values and fragmentation across different SLAPX message types. Although the 5G standard does not mandate a fixed MTU, many deployments adopt the standard Ethernet MTU of 1500 bytes [18]. Under this setting, only the *Service Request* and *Spectrum Request* messages experience fragmentation, increasing from one to two packets per message. The lower plot in Figure 5 further analyzes communication overhead under varying MTU sizes, illustrating the proportion of protocol and network headers relative to payload size. As expected, messages with smaller payloads incur higher relative overhead due to header dominance.

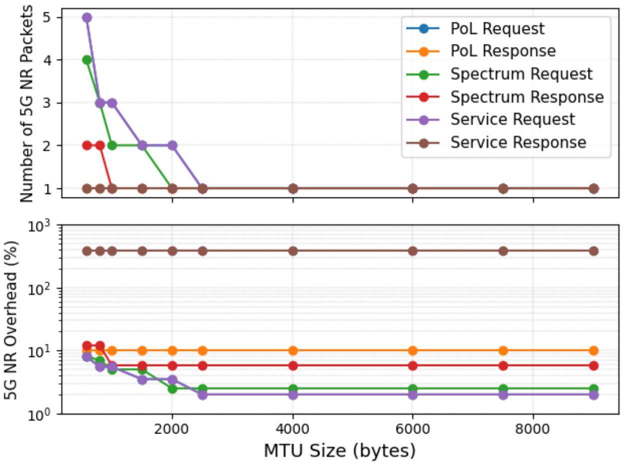


Figure 5: Fragmentation analysis for SLAPX protocol messages with varying MTU sizes

DoS Experiments: Each of the three attack cases, along with a baseline scenario without SLAPX, is evaluated independently. For each scenario, we simulate 10 seconds of network time, with the attack initiated at $t = 2$ and terminated at $t = 8$.

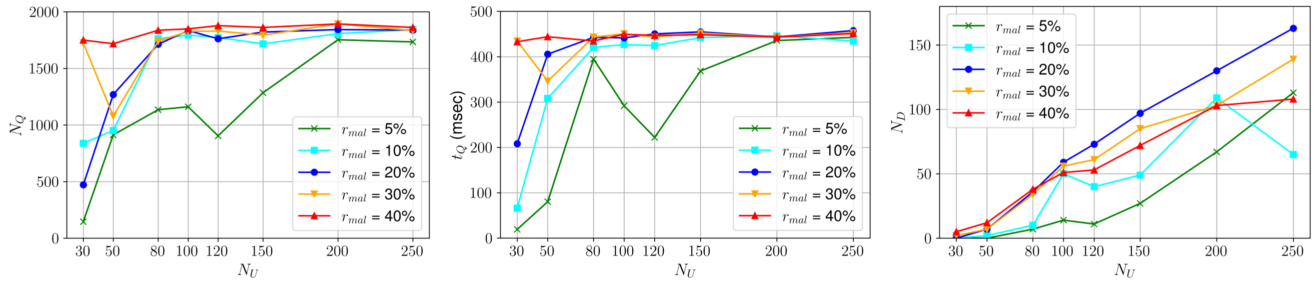
Baseline DoS: To establish a baseline and expose system vulnerabilities under adversarial load, we first evaluate the network in the absence of any protection mechanism. In this baseline scenario, SLAPX is disabled and malicious UEs directly flood the server with requests to induce service disruption. We conduct a large set of experiments with varying N_U and r_{mal}. System performance is evaluated using three key metrics: (i) the number of queued messages (N_Q), (ii) the average delay experienced by messages in the queue (t_Q), and (iii) the number of requests from benign UEs that are dropped due to full queue capacity and busy workers (N_D).

Among the evaluated performance metrics, Figure 6a illustrates the number of requests that cannot be processed immediately and are therefore placed into the server queue, denoted by N_Q. Apparently, N_Q increases sharply with both N_U and r_{mal}, reaching nearly 2000 messages in environments with high ratio of malicious UEs. Note that this value reflects the cumulative number of messages enqueued over time, rather than the maximum queue length at any instant. This distinction arises because queued messages are continuously dequeued for processing as workers become available while new requests arrive concurrently. Similar trends are observed in Figure 6b, which reports the average waiting time of messages in the server queue, representing the time a request spends queued before being processed by an available worker and detoted by t_Q. Without anti-DoS countermeasures provided by SLAPX, t_Q exceeds 100 ms in most scenarios and can rise to nearly 500 ms as N_U and r_{mal} increase. Such delays can severely impact time-sensitive applications and degrade overall service quality.

The most severe performance degradation is observed in Figure 6c, which shows the number of benign requests dropped by the server due to the DoS attack, denoted by N_D. As expected, N_D increases with N_U, since a larger population generates more legitimate traffic. Notably, the highest drop rate occurs when r_{mal} is 20%, exceeding even the $r_{mal} = 40\%$ case. This behavior can be explained by the reduction in the number of benign UEs as the malicious fraction grows, which ultimately limits the volume of legitimate traffic subject to dropping. This finding displays the vulnerability of a regular SAS without any counter-DoS mechanism.

Full Protocol DoS: We conduct experiments with varying N_D and r_{mal}, and summarize the results in Figure 7. System performance is evaluated using three metrics: N_Q, t_Q, and N_D. As shown in Figure 7a, SLAPX decreases N_Q to fewer than 50 messages during the simulations, a significant reduction compared to the baseline scenario, which yielded 1000 messages. Consistently, t_Q remains below 65 ms, as illustrated in Figure 7b. Moreover, no benign message drops are observed ($N_D = 0$), demonstrating that the DoS attack is effectively mitigated by SLAPX under this scenario.

Computation-Bypassing DoS: Again, we conduct experiments with varying N_U and r_{mal}, with results shown in Figure 8. The system is evaluated using the same three performance metrics as in the previous scenario. As illustrated in Figure 8b, t_Q remains almost negligible and is even lower than in the *Full-Protocol DoS* scenario, showcasing the effectiveness of SLAPX. While Figure 8a shows a relatively larger N_Q, this increase is expected and inevitable because adversaries can send many more requests without having to perform the computational challenge. Nevertheless, in this scenario, the server promptly identifies and discards malicious requests that lack valid puzzle solutions, preventing them from consuming processing resources. The effectiveness of this early filtering is further confirmed by the consistently low t_Q observed across all configurations,

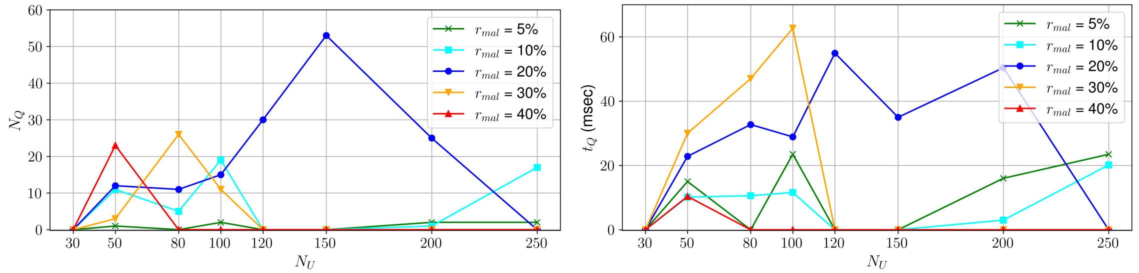


(a) Total number of queued messages (N_Q) against total number of UEs (N_U)

(b) Average waiting time of messages in the queue (t_Q) against total number of UEs (N_U)

(c) Total number of dropped messages (N_D) against total number of UEs (N_U)

Figure 6: Performance metrics for the Baseline DoS scenario across different numbers of UEs and malicious UE ratios



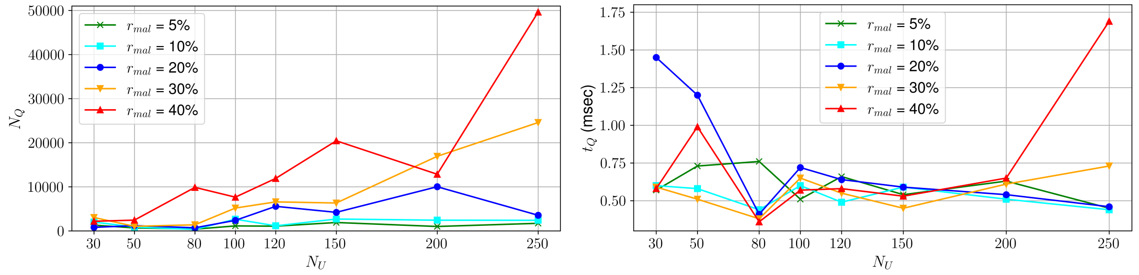
(a) Total number of queued messages (N_Q) against total number of UEs (N_U)

(b) Average waiting time of messages in the queue (t_Q) against total number of UEs (N_U)

Figure 7: Performance metrics across different numbers of UEs and malicious UE ratios, in the *Full Protocol DoS* scenario

indicating that SLAPX does not incur significant performance degradation in this scenario either. Still, one can place a network-level firewall in front of the application and add customized rules to drop high-frequency messages, from the same source, on a network level. This additional mechanism would prevent the server queue from filling quickly and thus, helps to obtain a much lower N_q.

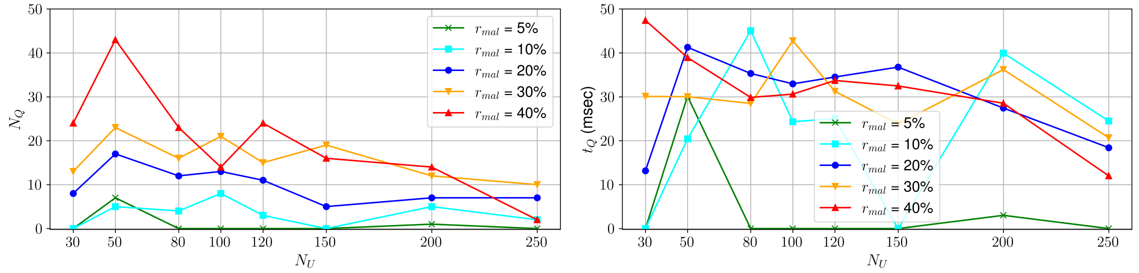
To further explain the large number of queued messages observed in this scenario, we examine the maximum queue length reached during the simulations, as shown in Figure 10. This analysis was omitted for the *Full-Protocol*



(a) Total number of queued messages (N_Q) against total number of UEs (N_U)

(b) Average waiting time of messages in the queue (t_Q) against total number of UEs (N_U)

Figure 8: Performance metrics across different numbers of UEs, in the *Computation-Bypassing DoS* scenario



(a) Total number of queued messages (N_Q) against total number of UEs (N_U)

(b) Average waiting time of messages in the queue (t_Q) against total number of UEs (N_U)

Figure 9: Performance metrics across different numbers of UEs, in the *Proof Precomputation DoS* scenario

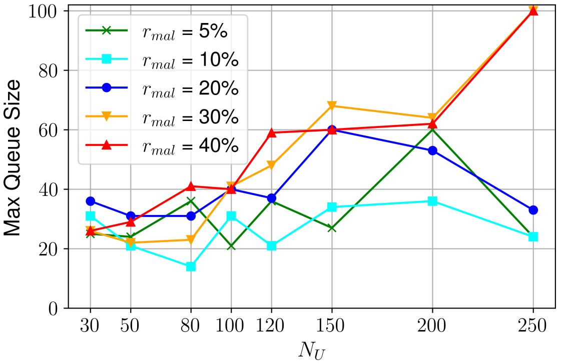


Figure 10: Maximum queue size against total number of UEs (N_U), in the Computation-Bypassing DoS scenario

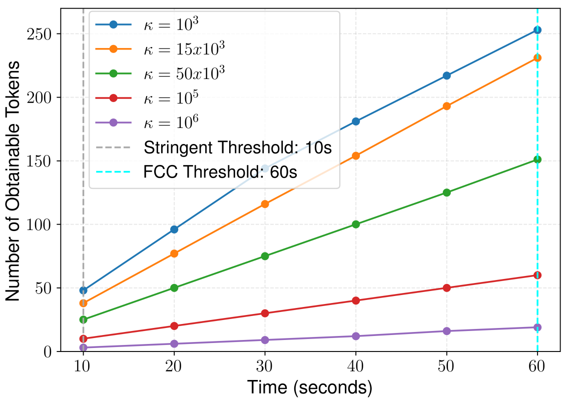


Figure 11: Limits of the proof pre-computation process

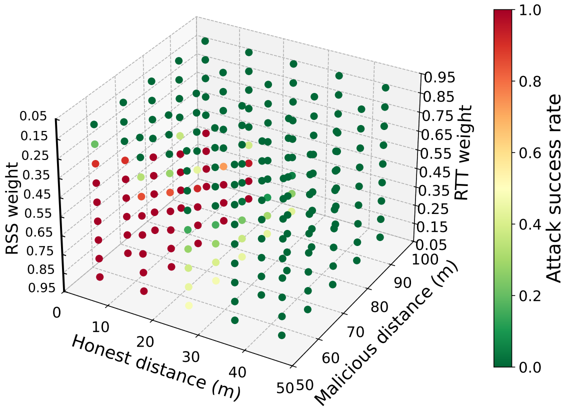


Figure 12: 3D Scatter chart for attack success rate in the distance hijacking attack

Table 5
Parameters of distance hijacking simulations

Parameter	Value Range	Value Step
Honest UE distance	0 - 50 meters	10 meters
Malicious UE distance	50 - 100 meters	10 meters
RTT weight in ProxVerify	0.1 - 0.9	0.1

DoS scenario due to the consistently low N_Q. In contrast, the present scenario exhibits a higher volume of queued requests, warranting closer inspection of queue saturation. As illustrated, the queue capacity of 100 is reached only in two configurations—both with 250 UEs and r_{mal} of 30% and 40%. Although packet drops occur in these cases, all dropped packets originate from malicious UEs, and no benign requests are denied service. This result confirms that SLAPX effectively contains the attack and preserves service availability for legitimate users.

Proof Precomputation DoS: Again, we conduct experiments with varying N_U and r_{mal}, with results displayed in Figure 9. The system is evaluated using the same three performance metrics, with results shown in Figures 9a and 9b. As observed, system performance closely resembles that of the previous protected scenarios and remains significantly better than the baseline configuration without SLAPX. Notably, zero benign messages are dropped (N_D = 0), demonstrating that SLAPX successfully prevents service disruption across all three DoS attack scenarios.

In addition to the standard metrics evaluated across all DoS scenarios, the puzzle pre-computation attack introduces an additional dimension of analysis. Although malicious UEs may attempt to pre-compute multiple puzzles in advance, the effectiveness of such an attack is inherently bounded. Each proof of location (`PoL`) includes a timestamp and is valid only within a limited time window. As a result, puzzle solutions expire after a fixed duration, and excessive pre-computation causes earlier solutions to become invalid before they can be used. Consequently, the maximum number of usable tokens an adversary can accumulate is constrained by the PoL expiration threshold and the puzzle difficulty parameters. This trade-off is illustrated in Figure 11. As illustrated, no UE can pre-compute more than 250 puzzles, even under the easiest configuration and with the FCC-defined validity threshold of 60 seconds. If the system operator adopts a stricter expiration window or increases puzzle difficulty, the number of obtainable tokens drops sharply. This analysis confirms that the defense provided by SLAPX can withstand adversaries with strong computational capabilities.

Location Spoofing Experiments: Here, we present the results of our experiments on the two location spoofing scenarios, each of which is evaluated separately.

Distance Hijacking: For one of the location spoofing scenarios, we evaluate the effectiveness of the Proximity Verification function (*ProxVerify*), which enables a gNodeB to estimate the distance to UEs using a combination of received signal strength (RSS) and round-trip time (RTT) measurements. In the considered distance hijacking attack, a malicious UE communicates with a compromised benign UE over an ad-hoc WiFi network using CSMA, while the benign UE maintains a legitimate 5G NR connection with the gNodeB. The benign UE is configured to relay all messages between the malicious UE and the gNodeB without delay. The proximity threshold is set to 50 meters, in accordance with FCC guidelines [19], such that devices within this range are classified as proximate, while those beyond it are considered distant.

To investigate the success of relay attacks under multiple factors, we vary a set of configuration parameters listed in Table 5. For each configuration, the experiment is repeated

100 times and the corresponding attack success rate is recorded. Figure 12 presents a 3D scatter plot in which color encodes the attack success rate as a function of the honest UE distance to the gNodeB, the malicious UE distance, and the relative weighting of RTT versus RSS in the *ProxVerify* function. The results show that attack success increases as the honest UE moves closer to the gNodeB and as the influence of RSS outweighs that of RTT. This observation aligns with earlier results, indicating that stronger emphasis on RTT improves resistance against distance hijacking. Moreover, as the honest UE approaches the gNodeB, the resulting higher RSS can be exploited by the malicious UE, further increasing the likelihood of a successful relay attack.

Distance Fraud: In this scenario, we examine the DBP, which enables a client to authenticate using a nearby, already authenticated device when no access point is available. We consider one honest UE that is legitimately connected to the network and one malicious UE attempting to mislead the nearby device into accepting it as being closer than its actual distance. The DBP enforces proximity by executing multiple rounds of rapid challenge–response exchanges. In each round, the device must return a sequence of bits in the correct order and within a strict time bound determined by the measured RTT. In this setup, we initiate the distance fraud scenario by first placing an authenticated, honest UE in the network and then introducing a malicious UE at a random distance between 50 and 100 meters from it. Since it is not possible to exceed the speed of light, the malicious device cannot reduce RTT to appear closer. However, the adversary can guess the challenge bits in advance and respond before receiving them. If the guesses are correct, the malicious UE may succeed in authenticating despite being outside the allowed proximity range.

To model the DBP and characterize the attacker's capabilities, we consider the following parameters:

- **Rounds**: The total number of challenge bits transmitted by the verifier (i.e., the honest UE) to the malicious UE during the DBP execution.
- **Tolerance**: The fraction of rounds in which a device is permitted to respond incorrectly while still being authenticated. For example, with a tolerance of 0.2 and 100 rounds, authentication succeeds if at least 80 rounds are answered correctly.
- **Guess Probability**: The probability that a malicious UE correctly predicts a challenge bit before receiving it. While this probability is 50% under random guessing, we conservatively allow higher values to model an attacker with partial predictive advantage.

To evaluate the success probability of distance fraud attacks, Figure 13 presents a 3D scatter plot in which color denotes the attack success rate as a function of the number of rounds, the attacker's guess probability, and the DBP tolerance threshold. As expected, attack success increases with higher tolerance levels and greater guess probabilities, since both parameters favor the adversary. Nevertheless, when the tolerance is set to a reasonable value—such as 10% or lower—the probability of a successful distance fraud attack becomes negligible.

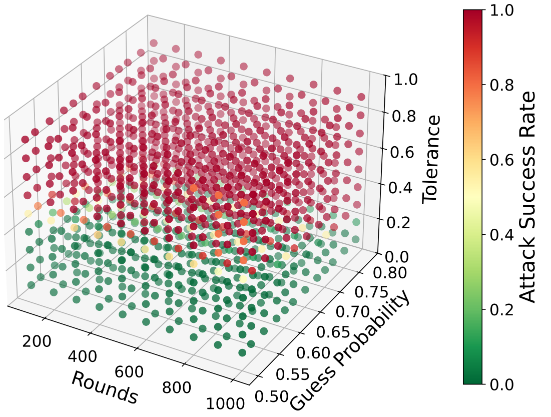


Figure 13: 3D Scatter chart for attack success rate in the distance fraud attack

6. Related Work

This section summarizes prior efforts and highlights key open problems along three major dimensions while describing our contributions beyond these efforts.

(i) Location Privacy and Anonymity in DB-CRN: In DB-CRNs, regulatory requirements mandate that unlicensed users (i.e., SUs) disclose precise location information, device characteristics, and spectrum requests to geolocation databases in order to obtain authorized channels and transmission parameters [4]. This regulatory enforcement fundamentally conflicts with the principle of privacy, as users must sacrifice their privacy for spectrum access, thereby exposing mobility patterns and long-term behavioral profiles, which then enables tracking and profiling attacks [24]. Moreover, location privacy without anonymity or unlinkability remains insufficient, as stable identifiers combined with coarse location allow tracking and linkage across queries [33]. Standard protections (e.g., TLS) do not prevent legitimate entities from observing, storing, and correlating sensitive query metadata [39], either. Furthermore, the loss of anonymity expands the attack surface and can facilitate active threats such as selective DoS, targeted blocking, and user discrimination [45, 14].

Despite extensive research efforts, existing privacy-enhancing mechanisms remain inadequate for real-world DB-CRN deployments. Schemes relying on k-anonymity or pseudonymous identifiers [49] provide only weak protection and lack rigorous security guarantees unless very large anonymity sets are maintained—an assumption that rarely holds in dense, mobile, or highly dynamic environments. Techniques based on location obfuscation or differential privacy [42], while theoretically grounded, intentionally perturb location data and consequently impair the precision and dependability of spectrum availability decisions. Cryptographic approaches leveraging Private Information Retrieval (PIR) [24, 47] offer stronger privacy guarantees but incur prohibitive computational and communication overhead, hindering scalability and practical deployment, while typically assuming benign user behavior and neglecting adversarial users. Consequently, our proposed solution utilizes primitives such as DAC and RLRS to provide effective protection in DB-CRNs through lightweight and scalable mechanisms that jointly ensure location privacy, anonymity, and unlinkability, while remaining compatible with regulatory mandates and operating in resource-constrained environments.

(ii) Location Proof and Spoofing Resistance in DB-CRN: DB-CRNs fundamentally operate as location-based systems in which spectrum availability decisions depend on real-time, user-provided location information stored in geolocation DBs. Consequently, spoofed or falsified locations undermine the basis of regulatory authorization and can cause disproportionate harm, as attackers may impersonate legitimate entities or manipulate location and usage data to obtain unauthorized spectrum access, resulting in spectrum interference, service disruption, and economic loss [34].

While some approaches introduce location-proof mechanisms, they often rely on strong trust assumptions such as trusted infrastructure, specialized hardware, or dedicated location-verification servers, which are impractical in rural, ad hoc, or infrastructure-limited deployments [47]. The need for location verification becomes even more critical when combined with location privacy and anonymity, creating a fundamental paradox in DB-CRNs: revealing precise location information to prove correctness undermines privacy, while hiding location without verification enables spoofing. Thus, many existing solutions compromise user privacy by exposing fine-grained location data to verifiers, enabling tracking and linkage. These shortcomings highlight the necessity for a practical, privacy-preserving location verification framework that can authenticate user proximity without disclosing sensitive location data, while remaining robust against diverse spoofing and distance-fraud attacks under realistic DB-CRN deployment conditions [15].

By utilizing primitives such as DBP, our proposed framework can thwart potential location spoofing attacks while preserving anonymity and unlinkability properties, which are essential for ensuring the privacy of users.

(iii) DoS Countermeasures and Spectrum Management for Next-Gen Network Systems: The centralized geolocation databases in DB-CRNs, due to their frequent interaction with massive IoT deployments, result in high-value choke points whose unavailability can disrupt spectrum access at scale [26]. Due to the cost asymmetry, even weak attackers can overwhelm spectrum DBs with excessive or illegitimate queries, causing high verification and processing overhead for servers, which in turn degrades spectrum coordination and reduces service availability.

Although various countermeasures have been proposed, including intrusion detection systems (IDSs), blockchain-based access control, and cryptographic Client Puzzle Protocols (CPPs), most existing defenses remain reactive and focus primarily on attack detection rather than prevention [13]. Such approaches often require continuous monitoring of traffic patterns, which incurs substantial operational overhead, and remain vulnerable to adversarial evasion [17]. CPP-based defenses impose computational costs on clients prior to service access [9], thereby limiting the rate of malicious requests. However, at large scale, they introduce new bottlenecks related to puzzle generation, distribution, and parallelization, which can burden both servers and legitimate users [5]. While outsourcing puzzle generation can alleviate server-side overhead [14], it merely shifts the attack surface to spectrum databases, which remain vulnerable to query flooding.

To mitigate the aforementioned limitations, our proposed framework leverages the concept of VDF and offloads the cost of puzzle computation to attackers instead. With this approach, a practical DB-CRN deployment can achieve the required lightweight and proactive rate-limiting mechanisms that are tightly integrated into the spectrum query workflow, enabling scalable DoS resilience while preserving availability and minimizing overhead.

7. Conclusion and Future Work

DB-CRNs improve spectrum utilization but introduce fundamental security and privacy challenges stemming from mandatory location disclosure, vulnerability to location spoofing, and exposure to DoS attacks on centralized infrastructure. In this work, we presented `SLAPX`, a unified and regulation-compliant security framework that simultaneously addresses these challenges under realistic adversarial assumptions. `SLAPX` enables privacy-preserving and anonymous spectrum access, provides adaptive and verifiable location proofs without revealing precise user location, and ensures robust DoS resilience through verifiable delay functions combined with RLRS-based rate limiting. Comprehensive cryptographic evaluation and network simulations demonstrate that `SLAPX` achieves significantly lower latency and communication overhead than existing approaches while effectively resisting spoofing and DoS attacks. As `SLAPX` currently targets classical security, future work will focus on extending its resilience to quantum-capable adversaries and providing long-term post-quantum security guarantees for DB-CRNs.

A. APPENDIX

A.1. Security Proofs

Here, we present the formal security proof of `SLAPX`.

Theorem 1. *The `SLAPX` framework achieves the following security guarantees: (i) anonymous user authentication, ensured by the anonymity, soundness, and unforgeability of the underlying ZKPoK and SPSEQ-UC signature schemes; (ii) location privacy, provided by the unlinkability of credential-based signatures; (iii) verifiable user location, enforced through either the unforgeability of `RLRS` combined with signal-based proximity measurements or the security of public-key distance-bounding protocols and anonymous credential delegation; and (iv) resilience to DoS attacks, enabled by `VDF` and rate-limiting mechanisms.*

Proof. *(i) Client Anonymity and Untraceability:* `SLAPX` ensures client anonymity and untraceability via *credential unforgeability* and *credential anonymity* (Definitions 5 and 6). For any PPT adversary $\mathcal{A}$ interacting with issuance, delegation, and presentation oracles, the advantage in distinguishing which of two honest users generated a valid presentation satisfies $Adv_{\mathcal{A}}^{\text{anon}}(\lambda) := \left|\Pr[b' = b] - \frac{1}{2}\right| \leq negl(\lambda)$, where b is the challenger's hidden bit selecting the presenting user. Otherwise, $\mathcal{A}$ can be reduced to either (i) breaking the knowledge soundness of the underlying ZKPoK, (ii) distinguishing re-randomized structure-preserving signatures, or (iii) distinguishing public-key–switched presentations, yielding a distinguisher for the Decisional Diffie–Hellman (DDH) problem in the underlying group $\mathbb{G}$. In particular, the SPSEQ-UC scheme guarantees *origin-hiding*, *derivation-privacy*, and *conversion-privacy* [20, 32], implying that randomized signatures, delegated commitment vectors, and key-switched credentials are computationally indistinguishable from fresh ones. These properties compose under repeated application, ensuring that credential presentations remain anonymous and unlinkable with all but negligible probability.

(ii) Client Location Privacy: `SLAPX` preserves user location privacy through *credential unlinkability* (Definition 7). In particular, location information is protected by the unlinkability of signature–commitment pairs generated via the SPSEQ-UC scheme [32]. The scheme supports signature re-randomization and public-key switching, ensuring that multiple presentations of the same underlying commitment yield computationally independent transcripts. Formally, for any PPT adversary $\mathcal{A}$ interacting with the proving oracle and observing a sequence of credential showings associated with the same location attribute ℓ, the advantage of linking two presentations satisfies $Adv_{\mathcal{A}}^{\text{link}}(\lambda) := \left|\Pr[b' = b] - \frac{1}{2}\right| \leq negl(\lambda)$, where b denotes the challenge bit selecting which of two unlinkable presentations is shown. Otherwise, $\mathcal{A}$ can be used to distinguish re-randomized SPSEQ-UC signatures or correlate switched public keys, contradicting the unlinkability of SPSEQ-UC under the generic group model. Consequently, repeated spectrum queries and location proofs in `SLAPX` remain computationally indistinguishable and unlinkable to verifiers, ensuring location privacy with all but negligible probability.

(iii) Location verification and Spoofing Resistance: In the infrastructure-assisted scenario, security follows from the core guarantees of the `RLRS`, namely unforgeability, signer anonymity (even against the key generator), linkability, non-slanderability, and revocable-iff-linked accountability, all proven in the random oracle model. Correctness and soundness follow directly from Definition 8. Each valid location proof embeds a unique event identifier and a hidden signer exponent s, whose deterministic projection $S = u_0^s$ (with $u_0 = G_0(\mathsf{event})$) serves as a cryptographically binding link tag. This tag cannot be forged, altered, or duplicated without solving the Discrete Logarithm (DL) or Decisional Diffie–Hellman (DDH) problems in group $\mathbf{G}$, or the q-Strong Diffie–Hellman (q-SDH) problem in the bilinear groups $(\mathbf{G}_1, \mathbf{G}_2)$, yielding adversarial success probability at most $negl(\lambda)$. Linkability ensures that any two signatures issued by the same signer under the same event yield identical tags, making replay or proof reuse publicly detectable; conversely, producing unlinkable signatures from the same key induces a reduction to DL with non-negligible probability. Revocation is enabled iff linkability holds: two linked signatures allow algebraic extraction of $U = u_0^{H(\mathtt{ID})}$, which is mapped to the signer identity via accumulator soundness. Any attempt to revoke anonymity without a valid linkage reduces to breaking the q-SDH assumption.

In the ND scenario, location verification security is ensured by three orthogonal mechanisms. First, the `AKA` protocol provides session key indistinguishability under the hardness of the Diffie–Hellman and discrete logarithm problems in the random oracle model, implying $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{AKA}}(\lambda) \leq negl(\lambda)$ for any PPT adversary $\mathcal{A}$. Second, proximity is enforced through a symmetric distance-bounding protocol `DBP`, where the verifier samples $m \in \{0,1\}^{2n}$ and executes n rapid challenge–response rounds; any adversary outside the allowed distance succeeds with probability at most $\Pr[\mathsf{Accept}] \leq (3/4)^n$, which is negligible in n, providing information-theoretic resistance against distance fraud, mafia fraud, and distance hijacking. Finally, delegated location proofs are unforgeable: any adversary producing a credential embedding a forged proof $\mathtt{PoL}^\star$ can be reduced to either forging the SPSEQ-UC structure-preserving signature or violating the soundness of the underlying NIZK proof system, yielding $\Pr[\mathsf{Forge}(\mathtt{PoL}^\star)] \leq \mathsf{Adv}^{\mathsf{EUF}} * \mathsf{SPSEQ\text{-}UC}(\lambda) + \mathsf{Adv}_{\mathsf{NIZK}}^{\mathsf{ZK}}(\lambda) \leq negl(\lambda)$ under the DDH assumption. Together, these guarantees ensure correct, privacy-preserving, and accountable location verification while providing strong resistance against spoofing attacks and collusion under realistic adversarial conditions.

(iv) Authenticated and Comprehensive Counter-DoS: In `SLAPX`, the authenticity of `VDF` tokens is enforced through the existential unforgeability of the underlying digital signature scheme `SGN` under chosen-message attacks (EUF-CMA). When instantiated with `ECDSA`, security reduces to the hardness of the Elliptic Curve Discrete Logarithm Problem (EC-DLP). Formally, for any PPT adversary $\mathcal{A}$ issuing at most $q(\lambda)$ signing queries, $\Pr\left[\mathtt{SGN.Verify}(\mathtt{PK}, m^\star, \sigma^\star) = 1 \wedge m^\star \notin \mathcal{Q}\right] \leq Adv_{\text{ECDSA}}^{\text{EUF-CMA}}(\lambda) \leq negl(\lambda)$, where $\mathcal{Q}$ denotes the set of messages signed by the legitimate issuer. Any successful forgery yields an `ECDSA` forger and thus an EC-DLP solver with non-negligible probability, contradicting the assumption.

Server-side DoS resilience is achieved using a `VDF` instantiated in an RSA group $\mathbb{G}$ of unknown order (e.g., modulo $N = pq$). For input x and delay parameter T, the prover computes $y = g^{2^T}$ together with a succinct proof π, while verification requires $O(\log T)$ time. The security of the `VDF` relies on: (i) *inherent sequentiality*, as computing g^{2^T} requires T dependent squarings; (ii) *non-parallelizability*, since no asymptotic speedup is possible in groups of unknown order; (iii) *soundness*, where producing a valid proof (y^*, π^*) with $y^* \neq g^{2^T}$ implies breaking the Strong RSA assumption; and (iv) *uniqueness*, ensuring a single valid output for fixed (x, T). Let τ denote the per-request `VDF` evaluation cost. For any PPT adversary $\mathcal{A}$ issuing at most $q(\lambda)$ requests, $\Pr[\exists\,(y^*, \pi^*) : \mathtt{Verify}(x, T, y^*, \pi^*) = 1 \wedge \mathsf{Cost}(\mathcal{A}) < q(\ \lambda)\cdot\tau] \leq negl(\lambda)$, since otherwise $\mathcal{A}$ computes valid `VDF` outputs without incurring $\Omega(\tau)$ sequential work per accepted request, yielding a Strong RSA solver. Thus, accepted requests impose a provable per-request computational cost on clients while keeping verification efficient, establishing strong asymmetric DoS resistance.

To protect PSDs against query flooding, SLAPX enforces rate limiting via event-scoped linkability provided by RLRS signatures embedded in location proofs. Each valid signature implicitly contains a secret exponent s (hidden via zero-knowledge proofs) and a deterministic link tag $S = u_0^s$, where $u_0 = G_0(\text{event})$. The tag S is deterministic, event-specific, and signer-unique. For any PPT adversary $\mathcal{A}$, $\Pr[\exists\, \sigma_1, \sigma_2 : \mathtt{Verify}(\sigma_1) = \mathtt{Verify}(\sigma_2) = 1 \wedge S(\sigma_1) = S(\sigma_2) \wedge s_1 \neq s_2] \leq negl(\lambda)$, since $u_0^{s_1} = u_0^{s_2}$ with $s_1 \neq s_2$ yields a solution to the Discrete Logarithm problem. Hence, multiple signatures issued by the same signer under the same event are publicly linkable, bounding the number of valid queries by the number of distinct ring members (e.g., APs) in the region. Moreover, *revocable-iff-linked* accountability is guaranteed: only when two signatures are linked can the signer's identity be extracted via $U = u_0^{H(\mathtt{ID})}$, with correctness ensured by accumulator soundness. Any attempt to revoke anonymity without linkability reduces to breaking the q-Strong Diffie–Hellman (q-SDH) assumption.

By combining EUF-CMA authenticated VDF tokens, VDF-enforced sequential computational cost, and RLRS-based event-scoped rate limiting, SLAPX ensures authenticated request handling, provable per-request hardness, enforced rate limits, and cryptographic accountability. Any adversary that violates availability without incurring proportional cost would yield a solver for EC-DLP, Strong RSA, or DL/DDH/q-SDH with non-negligible probability, contradicting standard hardness assumptions. □

Acknowledgment

This work has received co-funding from the NSF and SNSF, in the framework of the SATUQ project under SNSF (Grant No 10000409) and NSF (ECCS 2444615).

References

[1] , 2025. NS3 Network Simulator. https://www.nsnam.org/. Accessed: 2026, January.

[2] , 2025. Study on channel model for frequencies from 0.5 to 100 GHz. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3173. Accessed: 2025, October.

[3] Abuyaghi, M., Si-Mohammed, S., Shaker, G., Rosenberg, C., 2024. Positioning in 5g networks: Emerging techniques, use cases, and challenges. IEEE Internet of Things Journal .

[4] Agarwal, P., Manekiya, M., Ahmad, T., 2022. A survey on citizens broadband radio service (cbrs). Electronics 11.

[5] Ali, I.M., Caprolu, M., Pietro, R.D., 2020. Foundations, properties, and security applications of puzzles: A survey. ACM Computing Surveys (CSUR) 53, 1–38.

[6] Au, M.H., Liu, J.K., Susilo, W., Yuen, T.H., 2013. Secure id-based linkable and revocable-iff-linked ring signature with constant-size construction. Theoretical Computer Science 469, 1–14.

[7] Bahrak, B., Bhattarai, S., Ullah, A., Park, J.M.J., Reed, J., Gurney, D., 2014. Protecting the primary users' operational privacy in spectrum sharing, in: 2014 IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN), IEEE. pp. 236–247.

[8] Boneh, D., Bonneau, J., Bünz, B., Fisch, B., 2018. Verifiable delay functions, in: Annual international cryptology conference, Springer. pp. 757–788.

[9] Bostanov, V., 2021. Client puzzle protocols as countermeasure against automated threats to web applications. IEEE Access 9.

[10] Caleffi, M., Cacciapuoti, A.S., 2014. Database access strategy for tv white space cognitive radio networks, in: 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking Workshops (SECON Workshops), IEEE. pp. 34–38.

[11] Chakraborty, T., Mitra, S., Mittal, S., 2023. Capow: Context-aware ai-assisted proof of work based ddos defense. arXiv preprint .

[12] Chen, V., Das, S., Zhu, L., Malyar, J., McCann, P., 2015. Protocol to access white-space (paws) databases. Technical Report.

[13] Darzi, S., Yavuz, A.A., 2024a. Counter denial of service for next-generation networks within the artificial intelligence and post-quantum era, in: IEEE 6th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications, IEEE. pp. 138–147.

[14] Darzi, S., Yavuz, A.A., 2024b. Privacy-preserving and post-quantum counter denial of service framework for wireless networks, in: IEEE Military Communications Conference (MILCOM), IEEE. pp. 1076–1081.

[15] Darzi, S., Yavuz, A.A., 2025. Slap: Secure location-proof and anonymous privacy-preserving spectrum access. 2025 Silicon Valley Cybersecurity Conference (SVCC) , 1–8.

[16] Das, S., Zhu, L., Malyar, J., McCann, P., 2015. Rfc 7545: Protocol to access white-space (paws) databases.

[17] Doriguzzi-Corin, R., Siracusa, D., 2024. Flad: adaptive federated learning for ddos attack detection. Computers & Security 137.

[18] Duckett, M., Moisand, J., Anschutz, T., Kourkouzelis, D., Arberg, P., 2006. Accommodating a Maximum Transit Unit/Maximum Receive Unit (MTU/MRU) Greater Than 1492 in the Point-to-Point Protocol over Ethernet (PPPoE). RFC 4638.

[19] FCC, 2025. FCC Regulations: 47 CFR 96.39. https://www.ecfr.gov/current/title-47/chapter-I/subchapter-D/part-96/subpart-E/section-96.39. Accessed: 2025, December.

[20] Fuchsbauer, G., Hanser, C., Slamanig, D., 2019. Structure-preserving signatures on equivalence classes and constant-size anonymous credentials. Journal of Cryptology 32, 498–546.

[21] Gao, Z., Zhu, H., Liu, Y., Li, M., Cao, Z., 2012. Location privacy leaking from spectrum utilization information in database-driven cognitive radio network, in: Proceedings of the 2012 ACM conference on Computer and communications security, pp. 1025–1027.

[22] Grissa, M., Yavuz, A., Hamdaoui, B., 2016. An efficient technique for protecting location privacy of cooperative spectrum sensing users, in: 2016 IEEE conference on computer communications workshops (INFOCOM WKSHPS), IEEE. pp. 915–920.

[23] Grissa, M., Yavuz, A.A., 2019. Location privacy in cognitive radios with multi-server private information retrieval. IEEE Transactions on Cognitive Communications and Networking .

[24] Grissa, M., Yavuz, A.A., Hamdaoui, B., Tirupathi, C., 2021. Anonymous dynamic spectrum access and sharing mechanisms for the cbrs band. IEEE Access 9, 33860–33879.

[25] Gür, G., 2020. Expansive networks: Exploiting spectrum sharing for capacity boost and 6g vision. Journal of Communications and Networks 22, 444–454. doi:10.23919/JCN.2020.000037.

[26] Jakimoski, G., Subbalakshmi, K., 2008. Denial-of-service attacks on dynamic spectrum access networks, in: IEEE International Conference on Communications Workshops, IEEE. pp. 524–528.

[27] Jasim, D.K., Sadkhan, S.B., 2021. Cognitive radio network: Security and reliability trade-off-status, challenges, and future trend, in: 2021 1st Babylon International Conference on Information Technology and Science (BICITS), IEEE. pp. 149–153.

[28] Kılınç, H., Vaudenay, S., . Efficient public-key distance bounding protocol, in: 22nd International Conference on the Theory and Application of Cryptology and Information Security, Springer. pp. 873–901.

[29] Li, Y., Zhou, L., Zhu, H., Sun, L., 2015. Privacy-preserving location proof for securing large-scale database-driven cognitive radio

networks. IEEE Internet of Things Journal 3, 563–571.

[30] Liang, Y.C., Chen, K.C., Li, G.Y., Mahonen, P., 2011. Cognitive radio networking and communications: An overview. IEEE transactions on vehicular technology 60, 3386–3407.

[31] Lopez-Perez, D., Garcia-Rodriguez, A., Galati-Giordano, L., Kasslin, M., Doppler, K., 2019. Ieee 802.11 be extremely high throughput: The next generation of wi-fi technology beyond 802.11 ax. IEEE Communications Magazine 57, 113–119.

[32] Mir, O., Slamanig, D., Bauer, B., Mayrhofer, R., 2023. Practical delegatable anonymous credentials from equivalence class signatures. Proceedings on Privacy Enhancing Technologies .

[33] Newman, Z., Servan-Schreiber, S., Devadas, S., 2022. Spectrum: High-bandwidth anonymous broadcast with malicious security .

[34] Nguyen-Thanh, N., Ta, D.T., Nguyen, V.T., 2019. Spoofing attack and surveillance game in geo-location database driven spectrum sharing. IET Communications 13, 74–84.

[35] Order, D., 2023. Federal communications commission .

[36] Patil, A., Iyer, S., López, O.L., Pandya, R.J., Pai, K., Kalla, A., Kallimani, R., 2024. A comprehensive survey on spectrum sharing techniques for 5G/B5G intelligent wireless networks: Opportunities, challenges and future research directions. Computer Networks 253.

[37] Patriciello, N., Lagen, S., Bojovic, B., Giupponi, L., 2019. An E2E simulator for 5G NR networks. Simulation Modelling Practice and Theory 96, 101933.

[38] Rösler, S., Eising, N., Zubow, A., Dressler, F., 2025. Improving wifi ranging through frequency diversity and mobility, in: 2025 IEEE 50th Conference on Local Computer Networks (LCN), IEEE. pp. 1–7.

[39] Shi, S., Xiao, Y., Lou, W., Wang, C., Li, X., Hou, Y.T., Reed, J.H., 2021. Challenges and new directions in securing spectrum access systems. IEEE Internet of Things Journal 8, 6498–6518.

[40] Thippeswamy, M., Prasanna, A.D., Takawira, F., 2016. Physical layer, data link layer, network layer, transport layer, and application layer in cognitive radio networks, in: Introduction to Cognitive Radio Networks and Applications. Chapman and Hall/CRC, pp. 187–200.

[41] Troja, E., Bakiras, S., 2014. Leveraging p2p interactions for efficient location privacy in database-driven dynamic spectrum access, in: 22nd ACM SIGSPATIA Conference, pp. 453–456.

[42] Ul Hassan, M., Rehmani, M.H., Rehan, M., Chen, J., 2022. Differential privacy in cognitive radio networks: a comprehensive survey. Cognitive Computation 14, 475–510.

[43] Vaudenay, S., 2015. Private and secure public-key distance bounding: application to nfc payment, in: International Conference on Financial Cryptography and Data Security, Springer. pp. 207–216.

[44] Wang, Y., Lin, C., Li, Q.L., Fang, Y., 2007. A queueing analysis for the denial of service (dos) attacks in computer networks. Comput. Netw. 51, 3564–3573.

[45] Weifang, W., 2010. Denial of service attacks in cognitive radio networks, in: 2010 The 2nd Conference on Environmental Science and Information Application Technology, IEEE. pp. 530–533.

[46] Wesolowski, B., 2020. Efficient verifiable delay functions. Journal of Cryptology 33, 2113–2147.

[47] Xin, J., Li, M., Luo, C., Li, P., 2016. Privacy-preserving spectrum query with location proofs in database-driven crns, in: 2016 IEEE Global Communications Conference (GLOBECOM), IEEE. pp. 1–6.

[48] Zeng, Y., Xu, L., Yang, X., Yi, X., 2019. An efficient privacy-preserving protocol for database-driven cognitive radio networks. Ad hoc networks 90, 101739.

[49] Zhu, R., Xu, L., Zeng, Y., Yi, X., 2019. Lightweight privacy preservation for securing large-scale database-driven cognitive radio networks with location verification. Security and Communication Networks .