Enforcing Dynamic Spectrum Access with Spectrum Permits

Lei Yang*, Zengbin Zhang, Ben Y. Zhao, Christopher Kruegel and Haitao Zheng
Department of Computer Science, University of California, Santa Barbara
*Intel Labs, Hillsboro, Oregon
lei.t.yang@intel.com, {zengbin, ravenben, chris, htzheng}@cs.ucsb.edu

ABSTRACT

Dynamic spectrum access is a maturing technology that allows next generation wireless devices to make highly efficient use of wireless spectrum. Spectrum can be allocated on an on-demand basis for a given geographic location, time duration and frequency range. However, a major obstacle to adoption remains. There are no effective solutions to protect licensed users from *spectrum misuse*, where users transmit without properly licensing spectrum, and in doing so, interfere and disrupt legitimate flows to whom the spectrum is assigned. Given the flexibility of today's cognitive radios, an application can easily transmit on frequencies outside of its allocated range, either accidentally due to misconfiguration, or intentionally to avoid spectrum licensing costs. In this paper, we propose a system to secure dynamic spectrum transmissions, where authorized users embed secure *spectrum permits* into data transmissions, thus enabling patrolling trusted devices to detect devices transmitting without authorization. We focus our attention on the development of spectrum permits, and describe *Gelato*, a spectrum misuse detection system that minimizes both hardware costs and performance overhead on legitimate data transmissions.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Network Architecture and Design

General Terms

Design, Experimentation, Security

Keywords

Dynamic spectrum access, spectrum permits, spectrum misuse detection, cognitive radios

1. INTRODUCTION

Dynamic spectrum access, where cognitive radios are used to access unused spectrum ranges on demand, is the clear and widely-accepted solution to the spectrum shortage problem. A significant volume of recent research has built the core algorithms and techniques necessary for the deployment of dynamic spectrum networks in various frequency regions. Particular emphasis has been given to algorithms that maximize spectrum utilization through highly efficient, short-term, local spectrum allocations. While the FCC uses multi-day auctions that define nation-wide usage of large spectrum bands for many years, the vision of the ideal system is the opposite [4]: "local" spectrum owners allocate spectrum segments for a geographic location for short time periods, generally through automated, short-term auctions [36]. By letting realistic short-term demands dictate the size and duration of spectrum allocations, this approach would significantly increase utilization.

A major obstacle remains on the way to adopting current proposals of dynamic spectrum networks. Thus far, policy makers and researchers have not been able to find an effective solution to the problem of *spectrum misuse*. Specifically, we must allow users who have spectrum licenses to transmit, while preventing unauthorized users from transmitting and interfering with authorized transmissions. Without effective protection, users have no assurances their transmissions would operate without interference, and would have no incentive to pay for this type of spectrum access.

There are two general approaches to address this problem. A system can either seek to completely prevent unauthorized access, or it can detect and locate the misbehaving device during the offense. Here, prevention implies building tamper-proof mechanisms into each device to prevent it from operating without a valid spectrum license [2, 9, 33]. Given the ever-increasing flexibility of software defined radios, it is difficult to envision a completely tamper-proof prevention mechanism. Such changes would likely be costly, and inflexible to varying local conditions or spectrum policies.

We can draw an effective analogy between our problem and the problem of enforcing vehicle speed limits on roads and highways. Building a speed control into each vehicle would be difficult and costly, but a selective detection and punishment scheme in the form of highway patrols can be a very effective deterrent against speeding. Another similar problem is deterring illegal car parking, where authorization to park is dependent on the specific time and geographic location. Instead of a costly and complex per-vehicle solution, parking patrols (*e.g.* meter-maids) provide a much lower-cost and more practical deterrent.

Similarly, we believe a probabilistic system that detected and punished unauthorized transmitters is the approach most likely to succeed in practice. The solution should avoid prohibitively high hardware costs, such as those from densely deployed spectrum sensors [21, 33], and per-device identifiers or signatures, which can be duplicated with sophisticated hardware [33]. Like highway patrols or meter maids, our solution involves a number of trusted mobile devices that patrol transmission areas to detect unauthorized users. Authorized users display time-varying one-time keys that

MobiHoc'12, June 11–14, 2012, Hilton Head Island, SC, USA.

are easily verified but cannot be duplicated. Once an unauthorized transmission is detected, trusted devices can use secure localization techniques [20, 29] to locate the misbehaving devices and stop the unauthorized transmissions.

In this paper, we propose a system for securing dynamic spectrum transmissions through detection of spectrum misuse. When a user purchases a license to transmit on a given spectrum frequency, at a specific time and location, it receives from the spectrum owner a *spectrum permit*, a secure sequence of keys that prove its authorization to transmit in its operating spectrum. Users transmit their spectrum permits on a low-bit rate control channel embedded inside their data transmissions, while trusted *police devices* patrol transmission areas to detect misbehaving devices whose transmissions lack the necessary spectrum permits. We believe a successful spectrum permit system will help pave the way for wide-spread adoption of dynamic spectrum networks.

Spectrum permits have three key requirements: a) they must be flexible enough to specify a license for a given location, time and spectrum range; b) they must be intrinsically linked with the data transmission; and c) they must be readable by other devices without having to decode the data.

A potential solution is to insert permit into packets either as packet header or watermark within the data. But this is insufficient. First, observers must decode the packet to extract the permit, which creates serious privacy and performance issues. But more importantly, this method does not intrinsically link the permit to spectrum usage, and is thus vulnerable to spectrum misuse. An adversary can allow the permit to transmit successfully, and then override the data segment with its own transmission payload.

To meet these requirements, we propose to build a new control channel *physically embedded* inside the data transmission to carry spectrum permits. We leverage "cyclostationary signatures," a PHY layer feature where by intentionally repeating values of a range of frequency subcarriers, we can construct artificial signal peaks in the Spectral Correlation Function (SCF), which is easily detectable by external receivers. Where prior work used this to send a single constant bit between devices [27], we develop techniques to get fine grain control over positions of these signal peaks, effectively converting any arbitrary permit bit streams into features. Thus we construct a signaling channel that is embedded in the data transmission, but can be read by external receivers without decoding the data stream. We show that authorized spectrum users can repeatedly broadcast a secure certificate on this in-band signaling channel, proving their authorized status to any nearby police devices.

Building a robust spectrum permit system out of cyclostationary features faces several challenges. *First*, different hardware transmitters use different feature encoding schemes, and a receiver must understand the transmitter's scheme to decode its signals. *Second*, since each feature is transmitted in a single packet, we must discretize the signal stream into packets identifiable by observers. *Finally*, reading spectrum permits must be robust against frequency offset artifacts at each transmitter, as well as channel impairments and external malicious attacks.

The Gelato Spectrum Permit System. We develop *Gelato*, a robust spectrum permit system that embeds spectrum permits into the data channel in a way that is universally and reliably decodable by observer devices. Specifically, we make the following key contributions:

1. *Permit transmission*. A novel method to encode spectrum permits as features in the data channel, and a bootstrapping feature preamble that allows any observer to decode the permit.
2. *Permit decoding*. Time and frequency-domain tracking mechanisms to accurately detect features, despite loose synchronization and frequency offsets.
3. *Attack detection*. Addressing and detecting potential attacks, by estimating signal strength from features.
4. *Prototyping*. USRP2 GNU radio implementation, with both narrowband and wideband experiments.

2. THE SPECTRUM MISUSE PROBLEM

Despite efforts to improve spectrum allocation techniques, a dynamic spectrum network cannot function correctly without a way to enforce spectrum allocations and detect spectrum misuse. An application can easily transmit on frequencies outside of its allocated range, either accidentally due to bugs or misconfiguration, or intentionally to avoid paying spectrum license costs. Unchecked, interference from these "misuse" events will disrupt legitimate transmissions, ultimately destabilizing the system and preventing adoption.

In this section, we examine approaches to address spectrum misuse, describe our assumptions and goals, and define the threat model it must protect against. A spectrum user is "licensed" if she is authorized to transmit on a given spectrum range, at a particular location and time. We do not specify a transmit power limit in our definition, but assume it is specified by the operating spectrum range, and thus hard-coded into radio hardware. We also assume that any spectrum allocation system maintains spectrum exclusivity, *i.e.* no two users can be authorized to transmit on the same frequency, time and location. More specifically, we focus on spectrum misuse detection at the Access Point or Base Station. For potentially misbehaving client devices inside these networks, we rely on existing client registration mechanisms to identify such behavior [16].

2.1 Candidate Solutions

One general approach is *per-device prevention* using secure hardware or firmware. By deploying a spectrum enforcement module in each radio device, this approach aims to directly prevent each radio from accessing unauthorized spectrum. The enforcement module can be built into the radio hardware [33], or placed in the kernel and user space of the radio software [9]. Given the power and flexibility of software defined radios, however, studies assert that a per-device prevention mechanism would be costly and difficult to perfect [12]. This is particularly true when the allocation of spectrum varies over time, *e.g.* when spectrum is allocated in small time segments. In addition, attackers can modify software and firmware to bypass any enforcement modules. Subsequent advances on both sides can lead to an arms race between designers and attackers.

A second approach is to *detect spectrum misuse* in real time, after which police nodes can use secure localization mechanisms to locate and terminate unauthorized transmissions. Prior work has proposed solutions that rely on dense deployments of spectrum sensors, which would record local RF signal measurements, along with a device identifier for each transmission [21, 33]. The unique per-device identifiers can be used to distinguish licensed users from unauthorized users. These approaches have two significant limitations. First, they require a dense deployment of costly spectrum sensors for any geographic area using this system. This is because radiometric signatures can change over time and space [5], and it is very difficult to maintain and distribute per-device identifiers without a dense sensor deployment. Second, per-device unique identifiers are insecure, as hardware and MAC addresses can be forged, and even intrinsic hardware signatures can be replicated given the right equipment [7].

2.2 The Need for Spectrum Permits

Our goal is to build a system that protects authorized spectrum users by detecting spectrum misuse. In this context, we introduce the concept of *spectrum permits*, secure, verifiable keys demonstrating authorization to use a spectrum. An authorized user receives a secret from the spectrum owner or authority, uses it to generate a sequence of one-time cryptographic keys, and announces them sequentially over time to any nearby observers. Spectrum permits have several advantages over prior solutions. First, permits are simple to read and verify, thereby simplifying and reducing the cost of the detection infrastructure. Second, permits are implemented as one-time, cryptographic keys. As a result, they are tamperproof, and not vulnerable to attacks leveraging sophisticated hardware.

Our work makes several assumptions:

- *Spectrum allocation granularity.* Spectrum assignments are made on three dimensions: frequency, geographic area, and time. Geographic areas are no less than the transmission range of a device operating at max power.
- *Secure communication between users and spectrum owners.* Spectrum owners can securely disseminate secrets to users via a secure communication channel without fear of compromise.
- *Loosely synchronized clocks.* All transmitters should have clocks loosely synchronized with clock servers, *e.g.* NTP.

Based on our assumptions and constraints of prior approaches, we define three key goals for spectrum permits:

- *Universally decodable.* Spectrum permits must support devices operating on a variety of spectrum ranges. To avoid costly hardware requirements for police nodes, spectrum permits should be decodable by any wide-band receiver without decoding data packets.
- *In-band permit transmission.* Spectrum permits can be transmitted on a dedicated control channel. But securely associating a permit with a data transmission is difficult, especially if police nodes cannot decode the data transmission. Instead, our goal is to send spectrum permits *embedded* inside the data channel.
- *Reliability.* Spectrum permits must be transmitted reliably even in the presence of lossy data channels.

2.3 Threat Model

Our goals specify intended properties in the absence of adversaries. We now consider the types of adversaries and attacks that a wireless spectrum permit system is designed to detect. We define "attackers" to include both users who transmit without license either by accident or misconfiguration, and users who do so intentionally to avoid the costs of spectrum licenses, possibly modifying their software defined radios in the process. In either case, we assume attackers' data traffic resemble legitimate transmissions, but can be altered to avoid detection. To detect attackers, spectrum owners (*e.g.* the FCC) deploys trusted, highly mobile devices (*police nodes*) to monitor a transmission area for spectrum misuse.

Attackers in our model have these properties. First, each attacker has full control of its software defined radio, and can use it to eavesdrop on legitimate transmissions and transmit arbitrary data. Second, they can tune parameters such as transmission power and operating frequency, but are limited by device hardware constraints, *e.g.* finite transmission power. Third, attackers have reasonable resource limitations that prevent them from computationally revealing the secret keys, *i.e.* they cannot break strong cryptography via brute force. Finally, police nodes are mobile devices, do not transmit data, and cannot be found or compromised by attackers.

3. SPECTRUM PERMITS VIA GELATO

We propose Gelato, a *spectrum permit* system for dynamic spectrum networks. The idea is that an authorized user of a spectrum range receives a secure key that allows it to generate valid permits for a fixed time period and a specific location. In Gelato, each user broadcasts its valid spectrum permit once during each time window. Mobile "spectrum police" nodes can scan different spectrum ranges, passively listen to each transmitter's permit, and verify its validity in real time with the help of an online spectrum allocation server.

The Gelato system consists of two key components, a *permit authentication* mechanism that generates and authenticates spectrum permits at the application layer, and a *permit attachment* mechanism at the physical layer that allows each user to broadcast its valid spectrum permit in its physical transmissions, and each police device to reliably detect and decode permits without decoding actual data packets. In the following, we present the permit authentication design and leave the detailed description of permit attachment to Section 4.

3.1 Spectrum Permit Authentication

The spectrum owner runs an online spectrum allocation database on a trusted server. It allocates spectrum in small time blocks of fixed-size T_{int}. Given a geographic location, time and frequency range, if the spectrum is allocated, then the spectrum database returns in real time a secret K_n that represents the tail of a secure, one-way hash chain [19].

Our license verification scheme uses a secure one-way hash chain scheme, similar to authentication mechanisms used for broadcast authentication [24]. When a user U is allocated a spectrum range for n time blocks from t_0 to t_{n-1}, it is given a secret K_0. The user then computes a chain of hash codes by applying a secure one-way hash (*e.g.* SHA-1) recursively n times, producing:

$$K_0 \xrightarrow{SHA-1} K_1 \xrightarrow{SHA-1} K_2 \xrightarrow{SHA-1} \cdots K_{n-1} \xrightarrow{SHA-1} K_n$$

Starting at time t_0, the user U transmits key K_x on the embedded control channel, where x is a counter starting from n-1 that decrements once per time block. That is, the keys are transmitted sequentially in time in reverse order of the one-way hash chain, $K_n, K_{n-1}, \cdots, K_1, K_0$. Since the one-way hash function SHA-1 cannot be reversed, a node can only generate K_i from K_{i-1}. This means that attackers cannot generate valid keys for successive time windows using past key observations.

To verify the authenticity of a transmitter, a police node uses its location, time and spectrum range of the observed transmission to obtain from the database a hash chain tail K_n and a start time t_0. It computes the number of time blocks elapsed since t_0 to get the current index x of the hash chain. Assuming the key sent on the Gelato channel is K_x, the police node applies the SHA-1 hash recursively n-x times to generate the rest of the chain. If the final result matches K_n, it proves the transmitter knows K_0, and is therefore authorized to transmit on this spectrum, location and time.

An authorized user U transmits its key K_x once per time block. Since the key can be copied and retransmitted by any nearby device, an observing police node will only consider the first transmission of K_x as valid. Even if U is not transmitting, an attacker cannot replay a previously used key K_r, because K_r does not match the correct key in the hashchain corresponding to the current time block. A police node can detect a replayed key K_r, because the number of hashes between K_r and K_n does not match the number of time blocks between the current time and t_{n-1}.

Choosing T_{int}. The choice of time block size is a tradeoff between permit efficiency and effectiveness. Because a legal user transmits one key per time block, the overhead of the permit scales inversely with T_{int}. On the other hand, since the time to transmit each permit is less than T_{int}, an attacker can transmit in between permits to evade detection. To make permit verification more reliable within a fixed time, Gelato uses small sized time blocks, *e.g.* 1 minute. We will also discuss in Section 5.2 ways to detect these attackers.

Using small time blocks can produce a longer hash chain in the verification process. For example, in the worst case, a police node verifying a one day spectrum permit has to perform 1440 hash operations. There are several ways to address this. First, the spectrum hash chain can be refreshed on a small fixed interval, a new hash chain secret sent to the user, and the hash chain tail sent to the allocation database. Alternatively, a police node can cache an already announced key K_{ann}, and verify a new key by terminating the hash once it reaches K_{ann}. We can also speed up verification by either embedding additional information with the chain tail [10], or trading space for verification speed by using hash trees [23].

4. ATTACHING SPECTRUM PERMITS

We now examine the issue of attaching permits to transmissions. One straightforward solution is to transmit permits on an out-of-band control channel. This solution, however, suffers two disadvantages. First, it usually requires an extra radio to transmit on the out-of-band control channel, leading to higher hardware cost and complexity. Second, the out-of-band transmission makes it highly difficult to associate spectrum permits with data transmissions. Upon detecting a permit is being transmitted by a legitimate user, an attacker can transmit comfortably on the radio frequency covered by the permit. In this way, it hides behind the legal transmissions and evades detection. Thus, for a permit to be effective, it must be intrinsically linked to the current data transmission.

In Gelato, our solution is to build on a technique in the wireless physical layer called *Cyclostationary Features*. At a high level, cyclostationary features are created when signals across some sequence of wireless frequency segments are repeated, thus generating an easy to detect energy peak in the signal's spectral correlation function (SCF) map. A transmitter embeds license stream into the data transmission by controlling where it inserts energy peaks into the SCF map. The result is visible to any police device that can sense signals on the transmitter's frequency, without decoding data content on the frequency. And more importantly, the spectrum permit is intrinsically linked to the data transmission, reflecting the actual spectrum usage.

Next, we present the detailed design of Gelato's permit attachment. We first briefly introduce cyclostationary features, and then describe how Gelato devices embed permits to their data transmissions, and how Gelato police decodes the permits.

4.1 Background on Cyclostationary Features

A cyclostationary signal $x(t)$ is a digital signal whose autocorrelation function is periodical in t for any time lag [27]. This property manifests into unique features in the frequency domain – a signal peak at a specific location in $x(t)$'s spectral coherence function (SCF). External devices can detect each feature by capturing the RF signal on the transmitter's frequency and applying a N-point FFT to compute a normalized, discretized version of the SCF, as $S_x(\alpha, k)$. Here α defines the *cyclic frequency* and k defines the *spectral frequency*, both in the unit of frequency subcarriers.

Our design leverages a fact in wireless communications: OFDM is the prevailing scheme for data communication. It is widely adopted in current and upcoming wideband wireless technologies, such as 802.11a/g/n, LTE, DAB and Bluetooth 4.0. Using OFDM, we can intentionally introduce a cyclostationary feature into a digital signal by organizing its symbols [27]. Each OFDM symbol consists of N frequency subcarriers. We select w contiguous subcarriers indexed from p to $p+w-1$, and repeat their signals at subcarriers indexed $p+D$ to $p+D+w-1$. This new arrangement generates a group of w contiguous peaks in the SCF map at locations (α^*, k^*):

$$\alpha^* = D, \quad k^* = p + D/2 + i, \quad i = 1, 2, ..., w \tag{1}$$

Thus using a set of subcarrier repetition parameters (w, D, p), we can produce a distinct cyclostationary feature as a vertical strip of width w, centered at position ($\alpha = D$, $k = p + W/2 + D/2$). Figure 1(a) illustrates a sample feature generated using $(12, 64, 64)$. In this paper, we assume all Gelato transmitters use the same w/N. The peak strength s of the vertical strip depends on the received signal to noise ratio (SNR) of the data packet:

$$s = \frac{SNR}{1 + SNR} \tag{2}$$

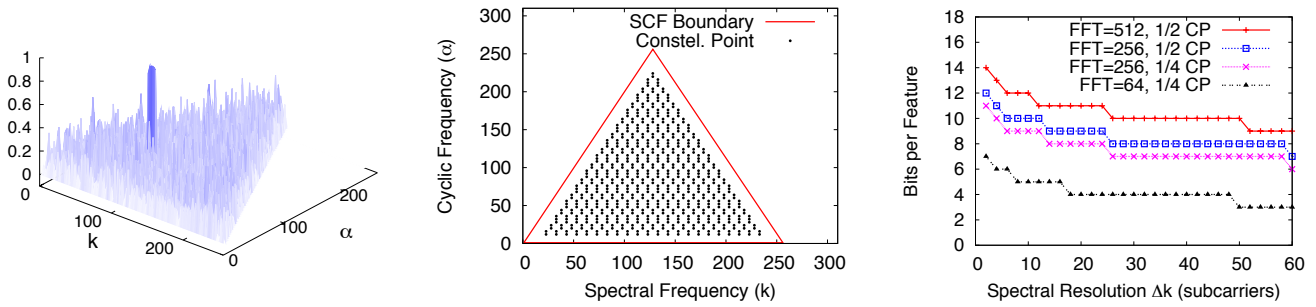
Finally, each feature needs to be transmitted continuously for a period of time (by a group of OFDM symbols). This is to ensure that the receiver can build a stable characterization of the SCF map, and suppress the impact of frequency-selective multipath fading [26]. Therefore, in Gelato, each data packet carries a single feature to maximize its robustness.

Cyclostationary features can be decoded using standard signal processing techniques without demodulating and decoding data packets. To detect cyclostationary features, each receiver computes the discrete SCF map from raw OFDM symbols and locates feature peaks. The correlation-based feature detection method has been shown to be optimal [11]. It computes the correlation between the SCF map and an ideal peak pattern (a vertical strip of width w), producing a new SCF map. This step eliminates noise in the system, as well as random occurrences of cyclostationary property in the packet data itself. Using the new SCF map, we can easily detect the feature location (α^*, k^*) by detecting peak on the projected cyclic and spectral frequency domain.

While injecting cyclostationary features requires modifying OFDM subcarriers, the decoding process can be made completely transparent to normal data transmissions. Each receiver can first detect and extract the feature, and proceed with data decoding by ignoring all subcarriers that have been identified to carry redundant data as part of the feature. Permit transmissions will not interfere with data packet delivery, because permit decoding is more robust than packet decoding (we confirm this via testbed experiments in Section 7). Of course there is a cost to transmit these control signals – a certain number (w) of subcarriers are no longer able to carry data. Our testbed experiments show that we can achieve reliable feature delivery with per-packet overhead as small as 5% for packets carrying cyclostationary features. As we discussed in the previous section, an authorized user only transmits its spectrum permit once per time block (*e.g. 1 minute*), thus the overall throughput overhead is $\ll 5\%$.

4.2 Displaying Spectrum Permits

The goal of Gelato is allowing each transmitter to display a stream of its spectrum permit bits as cyclostationary features. Thus the permit is intrinsically linked to its data transmission and readable by police devices without decoding data. To do so, Gelato faces two key challenges. First, we need an effective method to convert any arbitrary permit bit streams into features. Where prior works create signal peak to send a single constant bit between devices, we must



(a) SCF of Feature Example (b) Sample Feature Constellation Map (c) Feature Capacity

Figure 1: Building and encoding cyclostationary features. (a) A cyclostationary feature at ($\alpha = 64$, $k = 102$). (b) A sample feature constellation map for transmitters with FFT size of 256, CP length of $1/4$ and $\Delta k = 10$, mapping to 9 bits per feature. (c) Feature encoding capacity under different OFDM configurations (FFT size and CP length) and Δk.

develop systematic techniques to get fine grain control over positions of these peaks. Second, the content carried by each feature depends on the underlying OFDM configuration, which can differ across devices. Police devices must first obtain the configuration in order to decode the permit.

Gelato addresses these challenges via two novel solutions. First, it builds a *feature constellation map* to associate each bit pattern with a feature peak location. Second, it introduces a *feature preamble* to "broadcast" the OFDM configuration, therefore bootstrapping the permit decoding process.

A Feature Constellation Map. We encode bit patterns by associating a given bit pattern with a feature at a specific SCF map location. Since this approach is reminiscent of "constellation maps" used in digital signal modulation, *e.g.* QPSK or QAM, we refer to the collection of feature locations as the *feature constellation map*, and each location as the *feature constellation point*. Figure 1(b) illustrates a sample feature constellation map. To decode the feature, the receiving device first locates the feature peak from the SCF map, then computes encoded data as the bit pattern associated with the constellation point closest to the detected location.

The number of bits a feature can carry depends on the total number of distinct constellation points that can be reliably distinguished on an SCF map. This depends on the resolutions in the spectral frequency (k) and cyclic frequency (α) domains, *i.e.* the minimum spacing between adjacent constellation points to make them uniquely separable at the feature detector. We use Δk and $\Delta\alpha$ to represent the two.

Choosing Δk. For two reasons, the spectral frequency k is more sensitive to noise and channel artifacts compared to the cyclic frequency α. First, each cyclostationary feature maps to a single peak at a cyclic frequency α^*, but w consecutive peaks in the frequency domain (see Eq. 1). To decode k, we must accurately identify the center of the peak, which is sensitive to noise and channel artifacts. Second, frequency offsets between transmit and receive devices introduce more variability in the value of k. Therefore, Δk should be large enough to compensate noise and frequency offsets.

Choosing $\Delta\alpha$. The cyclic resolution $\Delta\alpha$ is determined by the transmitter's CP configuration. CPs are used to prefix each symbol with a repetition of its end, eliminating cross-symbol interference and mitigating multipath fading [13]. The use of CPs, however, changes the resolution of the SCF map, thus the detectable α positions. For transmissions with CP length of $1/M$, we can only

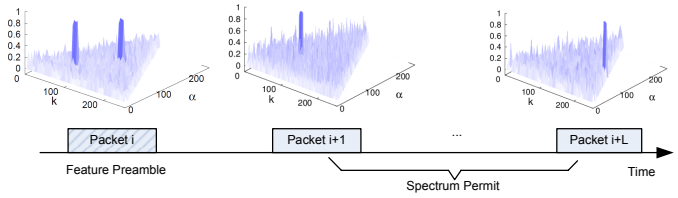


Figure 2: Each Gelato permit consists of its feature preamble and a group of features carrying the permit bit stream. The preamble carries two features, one on each half of the k-axis, carrying information on the FFT size and CP length required to decode the subsequent feature packets.

detect features at $\alpha = i \cdot M,\ i = 1, 2, 3, ...$, mapping to a cyclic resolution of $\Delta\alpha = M$. Typical values of M are 1, 2, 4 and 8 for existing OFDM systems, *e.g.* $M = 4$ in 802.11a/g [17].

For a given combination of Δk and $\Delta\alpha$, we create the optimal feature constellation map that places the maximum number of constellation points on the SCF map while satisfying the minimum spacing defined by $(\Delta k, \Delta\alpha)$. For example, for OFDM configurations with FFT size of 256, cyclic prefix (CP) length of $1/4$, and $\Delta k = 10$, we can encode 616 distinctive features on the SCF map (shown in Figure 1(b)). This means that each feature will carry $n = \lfloor \log_2(616) \rfloor = 9$ bits. Figure 1(c) shows the maximum number of bits each feature can carry under different configurations of FFT size, CP-length, and spectral resolution Δk.

A Feature-bootstrapping Preamble. Embedding bit patterns into data packets is not enough to produce a signaling channel to embed spectrum licenses. We face an additional challenge. Different transmitters can encode their data using very different values of FFT size and CP length, both of which must be known to define a feature constellation map.

Our solution is to introduce a feature preamble carrying the transmitter's FFT size and CP length to bootstrap the receiver. Figure 2 shows an example where each spectrum license message of size M is split across a group of $L + 1$ ($L = M/n$) data packets. We embed inside the first packet of the sequence a feature preamble that "broadcasts" the FFT size and CP length. Each of the next L packets carries a n-bit cyclostationary feature.

The preamble must be decodable by all devices regardless of their OFDM configuration, and easily distinguished from normal

spectrum license signal features. We encode the preamble as a set of two features, as shown in the SCF map of Figure 2. First, to make them even more easily distinguishable from normal spectrum license features, we make the width of preamble features twice the normal size. Second, we observe that common OFDM systems use a very limited number of FFT and CP length configurations, which can easily be represented by 4-5 bits. The two features in the preamble represent values for the FFT size and CP length, and the position of each feature is associated with a particular value for that parameter. For example, the feature on the left can represent one of four possible FFT sizes (64, 128, 256, 512), by dividing the left half of the SCF map into four quadrants and assigning a value to each quadrant. Knowing the values associated with each quadrant, the receiver can determine the FFT size by looking at the relative position of the left feature on the SCF map.

The FFT used in this step to build the SCF map is independent of the encoding FFT size, because the feature decoding only depends on its relative position on the SCF map. We can apply the same technique to encode one of four possible CP length values (1, 1/2, 1/4 and 1/8). Decoding the preamble allows the receiver to decode subsequent features at a higher rate with a fine-resolution constellation map.

4.3 Decoding Spectrum Permits

Intuitively, the basic feature decoding method (described in Section 4.1) should be sufficient. However, preliminary efforts to evaluate our system on a GNU radio prototype revealed several additional challenges. Next, we explain these challenges and our mechanisms to address them.

Tracking Permit in Time & Frequency. Our first challenge comes from the lack of time synchronization between the transmitter and the police. Because Gelato's feature embedding works on a per-packet basis, lack of time synchronization between devices means police receivers cannot accurately detect the beginning and end of discrete packets, leading to significant decoding errors.

To address this issue, Gelato police devices detect packet boundaries in time using an edge-detection based technique. This is done by identifying the sudden rise and drop of received signal strength that correspond to the beginning and end of each packet transmission. Specifically, a Gelato police monitors the raw received energy e on a given frequency band and computes its first-order directional derivative $\partial_t(e)$ in time. It detects a rising edge if $\partial_t(e) > \beta$, and starts to compute the SCF map. It detects a dropping edge if $\partial_t(e) < -\beta$ which marks the end of a feature transmission. If the time lag between the rise and the drop is greater than a threshold, it moves to detect and decode the feature using the captured time-averaged SCF map.

Our second challenge comes from the fact that per-device hardware artifacts produce *frequency offsets*, differences between devices' carrier frequencies that introduce large errors in the decoded spectral frequency k. Unaddressed, this would force us to use very large values for Δk, resulting in much lower bit-rates for embedding license permits. We address this challenge by applying an edge-detection technique to detect changes in the frequency domain, effectively removing the majority of frequency offsets [34]. Our testbed experiments show that the proposed tracking method effectively reduces the frequency offset to $< \pm 1$ subcarrier.

Coping with Frequency Selective Fading. Wideband transmissions often experience frequency-selective fading [15] where frequency subcarriers are attenuated differently. Thus we adjust Gelato's correlation-based feature detection to explicitly consider channel fading: we compute a new pattern for each feature location using the correlation of the channel fading pattern at the corresponding subcarriers.

Extracting Interleaved Features. Gelato is designed for dynamic spectrum networks where only one transmitter displays the spectrum license in a specific location and frequency. In rare cases, however, the police node may hear multiple permit transmissions at network boundaries. Since each spectrum license permit spreads over multiple features (thus over multiple packets), to correctly decode spectrum license permits, Gelato police nodes need to differentiate features (and packets) from different transmitters. To address this challenge, our solution separates transmitters by continuously comparing their radiometric features, including frequency offset, signal amplitude and radio transient shape developed by prior works [3, 6]. Note that we only use radiometric features as temporary radio identifiers. This differs from per-device identifier solutions that require dense sensor deployments to record radiometric features for every authenticated device [21, 33].

5. DEFENDING POTENTIAL ATTACKS

In this section, we examine in detail adversarial attacks against the Gelato spectrum permit system, and describe Gelato mechanisms to address and detect each type of attack.

Since the primary goal of our spectrum permit system is to detect spectrum misuse, we *explicitly do not seek to prevent or defend against denial of service attacks*, where an attacker sends unauthorized signals to intentionally disrupt an ongoing legitimate transmission. Such attacks are easy to detect and localize. In addition, while physically locating and punishing attackers are essential steps following attack detection, this paper focuses on spectrum permits and leave those topics as subjects for ongoing work.

5.1 The Copycat Attack

To use spectrum without a permit, attackers can eavesdrop on a legitimate transmission, extract its spectrum permit, and then attempt to use the permit for its own data transmissions. This attack is relatively easy to detect, since each legitimate user only transmits her permit once during each time block. The police node can easily detect an attacker if the same permit is transmitted twice.

Within the allocated geographic area for a given permit, there might be regions where the legitimate transmission signal is weak, and the copycat transmission will go undetected. However, since each spectrum allocation request is for a given usage area, such regions are likely small compared to areas where both transmissions overlap, and the attacker can be detected as police nodes move around the area.

5.2 The Free-rider Attack

This attacker hides behind legitimate users, *i.e.* by sending data packets in parallel without embedding spectrum permits. If the interference from the attacker is moderate, a casual observer would only observe a legitimate permit and a single transmission formed by the union of the legitimate transmission and the free-riding transmission.

Gelato police nodes can detect this attack by comparing the signal strength of the embedded control features to the raw received signal strength to detect the contribution of hidden free-riders. If the raw signal strength is significantly higher than the signal strength observed on the control features, then one or more hidden transmitters are close by. To detect this, Gelato offers a tool that estimates the received signal strength of a transmitter from the peak strength values of its features. Specifically, Gelato estimates the signal strength S^* of a transmitter from its feature strength s,

$$S^* = \left(\frac{1}{\rho/s - 1}\right) \cdot N_0 \quad (3)$$

where N_0 is the thermal noise power, and $\rho \leq 1$ is a device-dependent parameter, *e.g.* 0.9 for the USRP2 radios that we use to prototype Gelato. If S^* is less than the raw signal strength beyond a threshold, we claim a free-rider is present.

Addressing Frequency Selective Fading. Frequency-selective fading creates an additional challenge in extracting signal strengths from features. Since each feature is carried by a subset of subcarriers, the feature strength s only depends on the received signal strength on these subcarriers, rather than the overall received signal strength S^*.

Gelato addresses this by utilizing the fading profile observed by the police node. Since feature strength s is affected only by the subcarriers used to generate the feature, S^* estimated by (3) only reflects the average signal strength at the corresponding subcarriers. Hence, we can compensate the overall signal strength estimate by a factor η,

$$\eta = \frac{\sum_{i=1}^{N} \psi_i^2/N}{\sum_{i=1}^{w} \psi_{p+i} \cdot \psi_{p+D+i}/w} \quad (4)$$

where N is the total number of data subcarriers and ψ_i is the channel response at subcarrier i observed by the police node. We then use the compensated signal strength ηS^* to detect free-riders.

Some wireless technologies, such as LTE and WiMAX, apply transmit power adaptation on a per-subcarrier basis. As a result, transmit power can differ across subcarriers which leads to non-uniform receive signal strength. Similarly, Gelato police nodes compensate by measuring p_i, the receive power level at each subcarrier i, and applying a compensation factor η' according to (4), except by replacing ψ_i with p_i.

Free-riders in Transmission Gaps. If the transmission of a legitimate user U has a gap, *i.e.* it does not use the entire time block, an attacker could transmit for the remainder of the block, since U has given the current key. While this does not interfere with U's transmission, we can still detect the unauthorized transmission by observing changes in transmission properties such as signal strength and thermal noise power. For example, since frequency-selective fading is unique for each transmitter location, Gelato police can detect the attack by monitoring channel fading profiles.

5.3 The Bad-mouth Attack

Another type of intelligent attackers can seek to "bad-mouth" a legitimate user, *i.e.* frame an innocent user to look like she is transmitting illegally. The attack can be performed by "replacing" the victim's features with false ones. Specifically, the attacker occasionally transmits one or more false features at high power in parallel to the legitimate transmissions, which overpower and override the legitimate features. The police node would only observe replacement bits, thus corrupting the legitimate permit.

Gelato police can detect the presence of bad-mouth attacks by comparing the observed raw signal strength and the one estimated from the detected feature. In order to overpower the legitimate feature, the attacker receive power must be no less than that of the legitimate user. Thus if the observed permit is false, and the raw signal strength is occasionally more than twice the average feature-estimated strength, then a bad-mouth attacker is likely to be present.

We note that a legitimate permit can also be corrupted due to channel fading or unexpected interference. These impairments in general prevent a feature from being detected, rather than producing a false feature. Thus police nodes can examine the length of a received permit and separate these scenarios from the above bad-mouth attack.

6. IMPLEMENTATION

We implemented a Gelato prototype on USRP2 GNU Radios. It includes Gelato transmitter and receiver pairs for normal data communication, and police nodes for verifying spectrum permits and detecting attackers. While we chose GNU Radios for their availability, our design can be ported to other platforms [14,28,30] for improved frequency bandwidth and processing speed.

Gelato Transmitter & Receiver. Each Gelato transmitter consists of two processing paths: the normal data path and the permit displaying path. To display a permit, we modify the OFDM subcarrier mapping module in the data path to create subcarrier repetition. We implemented pilot tones following the same pilot/data ratio of WiFi. These pilot subcarriers do not follow Gelato's repetition rule, and can degrade the feature strength.

Gelato receivers are like normal data receivers, except that we add a permit detection and removal path. This is because bits from subcarriers carrying repetitive information to display spectrum permits should be removed from the data packet. Therefore, we modify each receiver to add a feature detection module. After locating the feature, the receiver's subcarrier demapping module simply removes the w duplicated subcarriers.

To determine the proper feature width w, we used different w to understand the tradeoff between feature robustness and packet overhead. Overall, we set the feature width $w = 12$ (5% overhead), and $\Delta k = 6$ such that each feature carries 9 bits of permit information. Each Gelato receiver then applies a detection threshold to distinguish real feature peaks from those caused by random factors including noise, interference, and more importantly, *inherent cyclostationary features displayed by data packets themselves*. To support a wide range of SNR values (0–20dB in our experiments), we choose a threshold of 0.4, which can reliably detect real features, but is still well above the "noise floor."

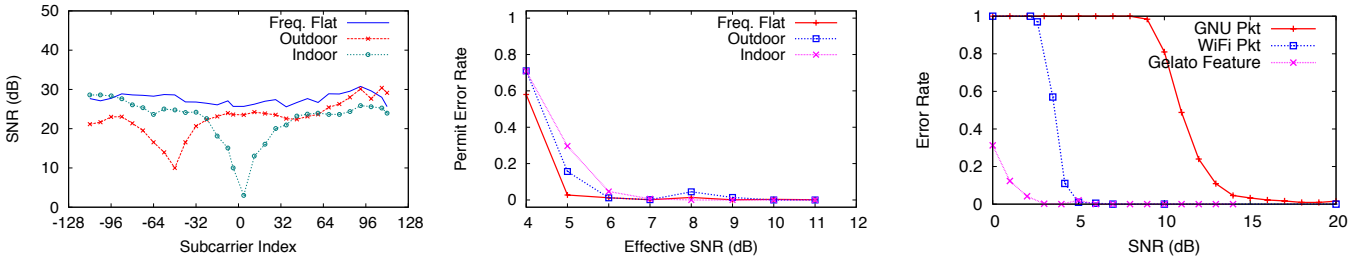
Gelato Police. We implement each Gelato police node as a standalone spectrum permit detector. The police reads OFDM signals from its USRP radio's A/D converter, and applies our proposed mechanisms to track packet boundaries and compensate for frequency offsets. We implement the proposed cyclostationary feature detection module to identify feature peaks and extract bits. The decoded permit bit stream is then validated using the proposed permit authentication process.

Gelato police nodes are much less complex compared to typical OFDM receivers. In addition to not performing packet demodulation/decoding, they require no synchronization in time and frequency. Both are among the most complex blocks in typical OFDM receivers [13]. We show in Section 7 that Gelato police nodes can decode features reliably without any FFT symbol level synchronization.

Frequency Configuration. Given the hardware limitation of USRP2 radios, our implementation currently supports a maximum FFT size of 2048 and a frequency bandwidth of 2MHz. To overcome this narrow-band limitation in our evaluation, we performed extensive wideband channel measurements in both indoor and outdoor environments using the tool provided by [15], and fed the measurement traces to our USRP2 transmitters to emulate wideband transmissions (20MHz) with frequency-selective fading.

7. EXPERIMENTAL EVALUATION

We evaluate Gelato using the aforementioned prototype implemented using USRP2 GNU radios. Since there are no existing



(a) Sample Channel Profiles (b) Gelato Permit Error Rate (c) Gelato Permit vs. Data Transmission

Figure 3: Reliability of Gelato spectrum permits. (a) Sample channel profiles for narrowband (frequency flat), outdoor wideband outdoor (frequency selective) and indoor wideband transmissions (frequency selective). (b) Gelato permits achieve a less than 5% permit error when the effective SNR is greater than 6dB for all three channel environments. (c) Gelato's feature detection is much more reliable than packet decoding.

comparable systems to Gelato, we instead focus on verifying its performance under various network configurations.

Experiment Setup. For each experiment, we build a set of 1600 permits, each 160-bit long. We embed each permit into a set of 18 randomly generated data packets. Each packet contains 32 OFDM symbols and carries a single cyclostationary feature. We also inject random gaps between packets. We focus on two representative indoor/outdoor scenarios in our experiments: complex indoor environments with furniture and walls, and outdoor environments with surrounding buildings, where both experiments are performed on our university campus. To examine the impact of channel fading, we also experiment with static/mobile scenarios: a static scenario where devices were placed statically, and a low-mobility scenario where we walked around the room with the feature receiver at a normal pedestrian speed. For both scenarios, there were random human movements throughout the experiments. Finally, while our prototype supports various transmission configurations on transmit power, FFT size and CP length, we observe in our experiments that these configurations lead to similar conclusions. Thus in the following we only show the results for 256 FFT and 1/4 CP length.

Our evaluation seeks to answer three questions:

1. *Can Gelato permits serve as a reliable method to authenticate spectrum usage in the presence of channel impairments and interference*?
2. *Will Gelato's feature transmissions be more reliable than data packet reception, so that they stay transparent to data transmissions*?
3. *Can Gelato police detect the presence of attackers, using the proposed feature-based signal strength estimation*?

7.1 Reliability of Gelato Spectrum Permits

Permit Error Rate. We first examine Gelato's permit error rate under different wireless transmission profiles. Specifically, we consider narrowband (1MHz frequency band with frequency-flat fading) and wideband channels (20MHz frequency band with frequency-selective fading). Figure 3(a) shows three illustrative examples of these channel profiles: one frequency-flat and two frequency-selective fading channels measured in indoor and outdoor. For the latter two, we observe large deviations across subcarrier SNRs. Figure 3(b) shows the error rate of Gelato permit reception. Since each permit is delivered by multiple features, it can only be successfully retrieved if *all* the features are received correctly. Overall, we see that the error rate reduces to <5% when the effective SNR (ESNR)* exceeds beyond 6dB. For outdoor WiFi access points, this requirement typically maps to 200-300 meters of detectable range from the police node to the transmitting access point [25]. This result implies that Gelato police might need to move around a legitimate user to get a "clearer" view of its permit.

Impact on Data Transmission. A key requirement (and advantage) of Gelato is to guarantee that data transmission will not be affected by the permit display except the expected throughput loss due to subcarrier repetition. To do so, the intended receiver of each data packet needs to detect the cyclostationary feature embedded in the data packet, and uses the corresponding subcarrier repetition pattern to correct the subcarrier demapping, *i.e.* removing the repeated subcarriers. This requires that the feature decoding is at least as robust as the packet decoding at each intended data receiver.

To verify this requirement, in Figure 3(c) we plot the feature decoding error compared to the packet decoding error for the packets containing no features, both implemented using USRP2 radios. For a fair comparison, we ignore feature errors caused by inaccurate packet locking, because it also prevents packet reception. Thus the corresponding feature error rate is better than that in Figure 3(b). Overall, we see that Gelato's feature detection is much more robust than packet decoding. Considering the fact that our implementation of data transmission may not be as sophisticated as that of commodity wireless transceivers, this comparison might not be representative. As a reference, we also compare our feature detection performance with the empirical result obtained from a recent WiFi study [15]. Again the feature detection outperforms the WiFi packet decoding. These confirm that Gelato permit is transparent to data transmission.

Mobile Police Nodes. To capture the impact of police mobility, we carried the police node and walked around to generate a low-mobility scenario. We used the same configuration as the above static experiments and repeated it 10 times. We found that mobility has very little impact on Gelato. For example, after sending 12 permits (216 features), only two 2 features that suffer very low SNRs were not decoded, leading to 2 corrupted permits (shown in Figure 4). We believe that this can be compensated by adding a low-level of error-correction coding [8] redundancy into each spectrum permit.

7.2 Attack Detection

*With frequency-selective fading, the average SNR does not accurately reflect channel quality. Thus we quantify channel condition using the Effective SNR metric [15], which is biased towards weaker subcarrier SNRs that contribute to most of the bit errors.

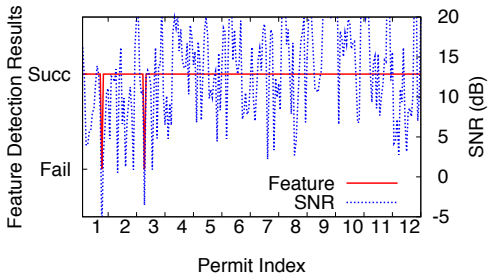


Figure 4: Impact of mobile police nodes. When walking around a large 12m×7m room with a Gelato receiver, we observe very few feature decoding errors caused by deep channel fades.

Next, we examine Gelato's ability to detect adversarial attacks. Since reliable permit transmissions and verification already enable the detection of copycat attacks, we focus on examining free-riders and badmouth attacks.

Accuracy of Feature-based Signal Strength Estimation. Since Gelato detects attacks by comparing the observed signal strength with the feature-estimated signal strength, we first verify the proposed signal strength estimation. To explore the impact of channel noise and interference (from other transmitters or attackers), we activate another transmitter to inject interference to the police node in the presence of the legal transmitter's transmission, and record the SINR observed at the police node. Figure 5 compares the estimated signal strength with the true value. We see that the estimation is quite accurate when the SINR is less than 8dB, but the accuracy drops at larger SINR values. This is due to the non-linear mapping between the SINR and peak strength. At high SINRs, a small deviation in peak strength computation manifests into larger errors in the estimated signal strength. Furthermore, we observe that frequency selective fading has negligible effect on the estimation accuracy *after* using our compensation method.

Attack Configuration. We implement both attacks and vary the attacker power to emulate different physical distance or power profile. Our experiments consists of an attacker, a legitimate transmitter (victim) and a police node. For both attacks, we use the *Relative Attacker Power* $(S_A(dB) - S_V(dB)$ to capture the difference between the received power of the attacker $S_A(dB)$ and that of the victim observed at the police node $S_V(dB)$. Because the legitimate receiver can be at any location within the legitimate transmitter's coverage area, as the police node moves around the network, the relative attacker power it observes also reflects the one observed at the legitimate receivers. The higher the relative attacker power observed at the victim receiver, the higher the performance degradation to the legitimate transmissions.

Detecting Free-riders. Figure 6(a) shows that in our indoor settings Gelato can reliably detect almost all (95+%) of free-riders whose signal strength is no more than 6dB weaker than the legitimate user. This means that the attacker needs to transmit at a very low power level to evade the detection, thus producing much less harmful interference to the legitimate users. Detecting weaker attackers is less reliable due to increased errors in feature based signal strength estimation at high SINRs (see Figure 5). The presence of a weak attacker only leads to a small drop in feature peak strength, which could also be caused by random noise and interference. This ambiguity increases false negatives (or miss detections). We also repeat the above experiments in outdoor scenarios and observe a slightly degraded accuracy (80% detection rate.) This is because outdoor transmissions suffer higher temporal variations from dynamic surroundings such as vehicles passing by. These variations introduce additional noise to feature peaks, degrading the detection accuracy.

For both scenarios the rate of false positives remains insensitive to attacker power settings. This is because false positives are mainly caused by the use of pilot tones which degrades feature peak strength and leads to false alarms. The impact depends on pilot locations rather than attacker power, and thus remains constant throughout the experiments.

Detecting Bad-Mouthers. To overwrite the victim's feature, a bad-mouther must transmit false features at a sufficiently high power. Figure 6(b) shows the performance of detecting bad-mouth attacker as a function of the attacker's relative power level for indoor scenarios. We see that Gelato's attack detection is highly effective – it forces the attacker to transmit at a significantly higher power (6+dB over the victim) in order to evade detection. These high-power attacks, however, are more visible and can be easily detected by checking signal strength and data transmission consistency over space and time, such as those proposed by [32]. Finally, we observe similar trends on false negatives and false positives like those of the free-rider attacks.

8. RELATED WORK

Spectrum Authentication and Misuse Detection. Existing work can be divided into two categories: *per-device prevention* and *external monitoring & detection*. Proposals in the first category apply on-device enforcement to prevent devices from operating without a valid spectrum license [2,9,33]. The second category includes diverse solutions designed for different network contexts. In the context of opportunistic spectrum access that contains primary and secondary users, prior works can authenticate each primary user using its unique link transmission characteristics created via a "helper" node [22], detect extra (illegal) transmitters by examining received signal strength [21], or apply extensive signal measurements to locate each transmitter and comparing their locations with those of legitimate users to identify violators [5]. These solutions require dense and costly deployments of monitoring sensors and helpers, and often assume ideal propagation models. More importantly, they place the burden of misuse detection completely on the detection infrastructure, making it costly and highly complex to perfect. Gelato takes a different direction - by forcing legitimate users to display their spectrum permits, Gelato shifts the responsibility to the users, significantly reducing the complexity and cost of the detection infrastructure.

Gelato also targets a different context, where wireless devices receive spectrum allocations on a short-term basis. In this context, the most relevant work is [1], where authenticated spectrum users are assigned unique slotted transmission patterns that serve as their authentication identities. This approach however, requires devices to share spectrum in the time domain, and also requires precise time synchronization.

Signal Embedding. Research efforts in this area have developed strategies to embed "side" information either directly into raw data bits (*i.e.* digital watermarking), or into physical-layer signals [18,31,35]. These solutions, however, all require demodulation/decoding of the original data transmission, which is infeasible in our scenario.

Gelato is motivated by prior work on cyclostationary features [27], but applies the concept in the context of displaying spectrum permits within transmissions. Unlike prior work, Gelato proposes a

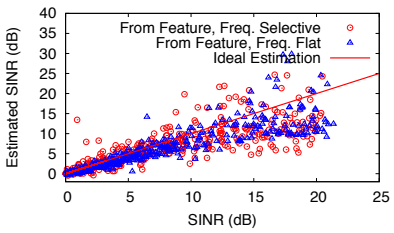
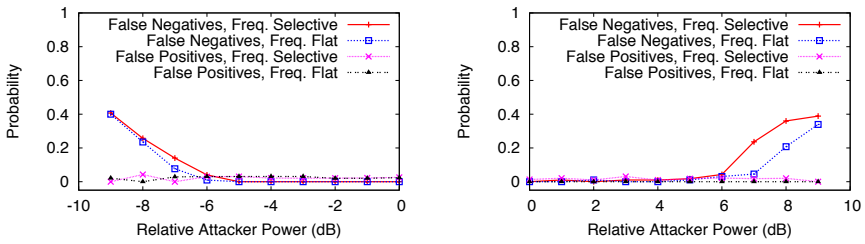


Figure 5: Estimating the signal strength of a legitimate user from its feature strength, for both indoor and outdoor environments.



(a) Detecting Free-riders (b) Detecting Badmouth Attackers

Figure 6: Performance of Gelato's attacker detection in indoor environments.

novel feature constellation map that allows features to carry arbitrary control information, and a robust detection framework to decode features in the presence of transmission artifacts and attacks.

9. CONCLUSION

We present Gelato, an initial step towards a robust spectrum permit system for authenticating spectrum usage and detecting misuse. Gelato devices transmit spectrum permits as cyclostationary features embedded inside their data transmissions, while trusted police devices patrol transmission areas to detect misbehaving devices. Gelato permits are reliable and "universally" decodable without requiring packet decoding. Detailed testbed experiments show that Gelato is a feasible, practical and cost-effective method for enforcing spectrum allocation.

10. REFERENCES

[1] ATIA, G., SAHAI, A., AND SALIGRAMA, V. Spectrum enforcement and liability assignment in cognitive radio systems. In *Proc. of IEEE DySPAN* (2008).

[2] BRIK, V., ET AL. Towards an architecture for efficient spectrum slicing. In *Proc. of HotMobile* (2007).

[3] BRIK, V., ET AL. Wireless device identification with radiometric signatures. In *Proc. of MobiCom* (2008).

[4] FCC. Connecting America: The National Broadband Plan, March 16, 2010. http://www.broadband.gov/plan/.

[5] CHEN, R., PARK, J.-M., AND REED, J. Defense against primary user emulation attacks in cognitive radio networks. *IEEE JSAC* (Jan. 2008), 25 –37.

[6] DANEV, B., AND CAPKUN, S. Transient-based identification of wireless sensor nodes. In *Proc. of IPSN* (2009).

[7] DANEV, B., ET AL. Attacks on physical-layer identification. In *Proc. of WiSec* (2010).

[8] DAVEY, M., AND MACKAY, D. Reliable communication over channels with insertions, deletions, and substitutions. *IEEE Trans. on Information Theory 47*, 2 (Feb. 2001), 687 –698.

[9] DENKER, G., ET AL. A policy engine for spectrum sharing. In *Proc. of IEEE DySPAN* (2007).

[10] FISCHLIN, M. Fast verification of hash chains. In *Proc. of CT-RSA* (2004).

[11] GARDNER, W. A. Signal interception: a unifying theoretical framework for feature detection. *IEEE Trans. on Commun. 36*, 8 (1988), 897Ű–906.

[12] GIACOMONI, J., AND SICKER, D. Difficulties in providing certification and assurance for software defined radios. In *Proc. of IEEE DySPAN* (2005).

[13] GOLDSMITH, A. *Wireless Communications*. Cambridge University Press, New York, NY, USA, 2005.

[14] GUMMADI, R., NG, M. C., FLEMING, K., AND BALAKRISHNAN, H. Airblue: A system for cross-layer wireless protocol development and experimentation. In *MIT Tech. Report* (2008).

[15] HALPERIN, D., ET AL. Predictable 802.11 packet delivery from wireless channel measurements. In *SIGCOMM* (2010).

[16] HAMID, M., ISLAM, M., AND HONG, C. S. Misbehavior detection in wireless mesh networks. In *Proc. of ICACT* (2008).

[17] IEEE 802.11-2007. `http://standards.ieee.org/getieee802/802.11.html`.

[18] KLEIDER, J., ET AL. Radio frequency watermarking for OFDM wireless networks. In *Proc. of IEEE ICASSP* (2004).

[19] LAMPORT, L. Password authentication with insecure communication. *Comm. of the ACM 24*, 11 (Nov 1981).

[20] LAZOS, L., AND POOVENDRAN, R. Serloc: secure range-independent localization for wireless sensor networks. In *Proc. of WiSe* (2004).

[21] LIU, S., CHEN, Y., TRAPPE, W., AND GREENSTEIN, L. ALDO: An anomaly detection framework for dynamic spectrum access networks. In *Proc. of INFOCOM* (2009).

[22] LIU, Y., NING, P., AND DAI, H. Authenticating primary users' signals in cognitive radio networks via integrated cryptographic and wireless link signatures. In *Proc. of IEEE S&P* (2010).

[23] MERKLE, R. A digital signature based on a conventional encryption function. In *Proc. of Crypto* (1987).

[24] PERRIG, A., ET AL. Efficient and secure source authentication for multicast. In *Proc. of NDSS* (2001).

[25] ROBINSON, J., ET AL. Assessment of urban-scale wireless networks with a small number of measurements. In *Proc. of MobiCom* (2008).

[26] SUTTON, P., LOTZE, J., NOLAN, K., AND DOYLE, L. Cyclostationary signature detection in multipath rayleigh fading environments. In *Proc. of Crowncom* (2007).

[27] SUTTON, P., NOLAN, K., AND DOYLE, L. Cyclostationary signatures in practical cognitive radio applications. *IEEE JSAC* (Jan. 2008), 13 –24.

[28] TAN, K., ET AL. SORA: High performance software radio using general purpose multicore processors. In *NSDI* (2009).

[29] ČAPKUN, S., ET AL. Secure location verification with hidden and mobile base stations. *IEEE Trans. on Mobile Computing* (April 2008), 470–483.

[30] Wireless open-access research platform, http://warp.rice.edu.

[31] WU, K., ET AL. Free Side Channel: Bits over Interference. In *Proc. of MobiCom* (2010).

[32] XU, W., ET AL. The feasibility of launching and detecting jamming attacks in wireless networks. In *MobiHoc* (2005).

[33] XU, W., KAMAT, P., AND TRAPPE, W. TRIESTE: A trusted radio infrastructure for enforcing spectrum etiquettes. In *Proc. of SDR workshop* (2006).

[34] YANG, L., HOU, W., CAO, L., ZHAO, B. Y., AND ZHENG, H. Supporting demanding wireless applications with frequency-agile radios. In *NSDI* (2010).

[35] YU, P., BARAS, J., AND SADLER, B. Physical-layer authentication. *IEEE Trans. on Information Forensics and Security 3*, 1 (2008), 38–51.

[36] ZHOU, X., GANDHI, S., SURI, S., AND ZHENG, H. eBay in the sky: strategy-proof wireless spectrum auctions. In *Proc. of MobiCom* (2008).