Robust Location Tag Generation from Noisy Location Data for Security Applications

Di Qiu, Dan Boneh, Sherman Lo, Per Enge, *Stanford University*

BIOGRAPHY

Di Qiu is a Ph.D. candidate in Aeronautics and Astronautics working in the Global Positioning System (GPS) Laboratory at Stanford University. Her current research interests are location-based security, signal authentication and information theory.

Dan Boneh is an associate professor of Computer Science and Electrical Engineering at Stanford University. He is a well-known researcher in the areas of applied cryptography and computer security.

Dr. Sherman Lo is currently a research associate at the Stanford University Global Positioning System (GPS) Laboratory. He is the Associate Investigator for the Stanford University efforts on the Department of Transportation's technical evaluation of Loran.

Per Enge is a Professor of Aeronautics and Astronautics at Stanford University, where he is the Kleiner-Perkins, Mayfield, Sequoia Capital Professor in the School of Engineering. He directs the Stanford GPS Research Laboratory.

ABSTRACT

Location-based security service provides authorization of persons or facilities based on their distinctive location information. It applies the field of position navigation and time (PNT) to the provision of security. Location parameters from radio navigation signals are mapped into a location verification tag or geotag to block or allow certain actions or access.

Loran and Wi-Fi are chosen as case studies because of their good properties that are beneficial to location-based security services. The achievable performance and the security of the system are determined by the quantity and quality of location features. By quantity, we mean the total number of different (independent) location dependent measurements available. By quality, we mean that amount of unique location dependent information and its consistency provided by each parameter that can be used to generate a robust location tag. It is desirable to have the parameters be relatively insensitivity to temporal changes which weaken the uniqueness of the information. As a result, repeatability and repeatable accuracy are the fundamental requirement for any location-based security service.

In this paper we propose fuzzy extractor scheme to generate strong location tag from noisy location data. A fuzzy extractor is an algorithm that can reliably extract desired information from input, and is error tolerant in the sense that this information will be the same even if the input changes. We develop different constructions of fuzzy extractor for various parameter distance measures: Euclidean and Hamming metrics. A combination use of the constructions will produce a more robust location tag. In addition, we mathematically analyze the entropy loss of the fuzzy extractor and study how to choose optimal design parameters under the tradeoff.

INTRODUCTION

In this paper we introduce a security-oriented location-based service and use Loran and Wi-Fi as case studies. In general, location-based services require accurate estimation of position, such as latitude, longitude and altitude, from location measurements. We show that there is no need to map location measurements into an accurate global position for a number of security applications. Loran, which operates at most of the northern hemisphere, has many advantages over satellite-based navigation systems for secure location-based service. It is a high power terrestrial signal and easily penetrates buildings and cities, where line-of-sight signals are not available. The Loran location-based parameters are used to derive a location tag, which is a piece of information that allows or restricts one's access for security applications. Although Wi-Fi was initially designed for communications between electronic devices, the proliferation of Wi-Fi has a growing interest in indoor location-based applications. Wi-Fi is chosen to complement Loran in the indoor environments. Location tag is computed by quantizing these parameters into grid spaces and mapping into a binary string. We provide examples of the location-based

security applications in two categories: block-listing and white-listing.

- *Block-listing*: An example of block-listing application is digital manners policy (DMP). Technologies for DMP [1] attempt to enforce manners at public locations. A DMP-enabled cell phone can be programmed by the phone provider to turn off the camera while inside a hospital, a locker room, or a classified installation. Or the phone can be programmed to switch to vibrate mode while inside a movie theater. Even though these ideas are highly controversial [2], we only focus on the technical aspect of the application in this paper. Using our location tag, one can build a list of location tags where the camera is to be turned off. The device downloads an updated list periodically. When the device encounters a location tag on this blocklist, it turns the camera off. When the device leaves the blocked location the camera is turned back on. Hence, digital manners are enforced without ever telling the device its precise location.

- *White-listing*: location-based access control is a white-listing example. Consider a location-aware disk drive. The drive can be programmed to work only while safely in the data center. An attacker who steals the device will not be able to interact with it. Location-based access control using encryption was studied by Scott and Denning [3] under the name Geoencryption.

A location-based security system must survive the following attack: the attacker owns the device and tries to make the device think it is somewhere else. We make two assumptions to survive this threat. First, a device that integrates a location sensor and geoag generation algorithm should be tamper-resistant. If a device is not tamper-resistant, one can perform attacks such as replacing received location parameters with fake ones, brute force attack or tampering with tag database. Second, radio signal is self-authenticated to allow users to verify the source of incoming signals. A signal authentication protocol, Timed Efficient Stream Loss-tolerant Authentication (TESLA) is proposed on Loran. We proposed a mean of implementing TESLA for authentication. The analysis and experimental results of authentication performance were discussed in our previous papers [4,5].

Additionally, it is desirable to have tags reproducible thus location parameters should be relatively insensitivity to temporal changes. Reproducibility means that measurements at the same location at different times always produce the same tag, and is a fundamental requirement to derive a robust geotag. However, several types of errors presented in the radio frequency (RF) signals can degrade the performance of location-based security service. This paper provides efficient error-tolerant techniques, called fuzzy extractor, to generate strong geotags from noisy location data. We propose the constructions of fuzzy extractor for various distance measures based on location error patterns, such as Euclidean distance and Hamming distance. Our constructions of fuzzy extractor can also be applied to other RF signals, such as satellite-based, Wi-Fi, TV, and cellular signals, and non-RF signals like infrared and ultra sound.

The structure of the paper is organized as follows. We first describe system models of a location-based security system and the error patterns of location dependent parameters in the next section. This paper then defines fuzzy extractor, and shows four different constructions of fuzzy extractor for various distance measures. We evaluate the reproducibility and security of location tags based on the constructed fuzzy extractors in the following sections. This paper then provides a performance comparison of the fuzzy extractors and concludes with future directions of the research.

SYSTEM MODELS

Reproducibility and repeatable accuracy are desirable qualities in location-based security systems. It allows one to provide his location-dependent parameters, or the derived tag at calibration—and still have those parameters valid at a latter time for verification. Figure 1 illustrates how the system works. Location parameters from the surveyed locations are mapped into tags and stored in a central database at calibration step. At verification step, one matches his computed tag with the stored one to validate the correctness of his location.

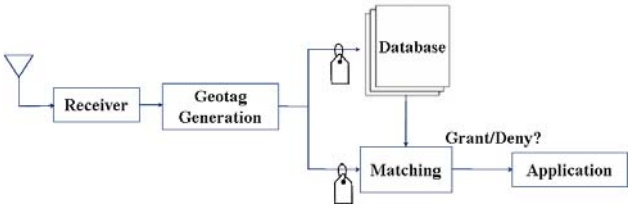

Figure 1. Location-based security system

The signal characteristics should be consistent enough so that when the user is ready to verify, measurements at the same location will yield the same previously generated tag. Temporal variation reflects instability or degree of scatter within a particular parameter at a given location and increases the likelihood to mismatch tags. Thus, we use fuzzy extractor to reliably extract location-based information from noisy Loran signal inputs. A fuzzy extractor is a form of error tolerant algorithm to reproduce desired secret information. The extraction is error-

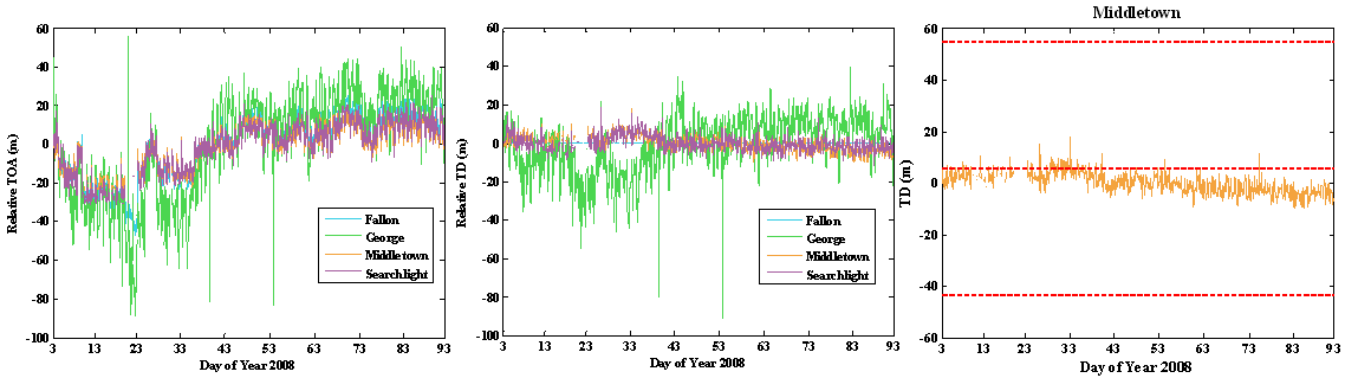

Figure 2. TOA with zero means (left); TD with zero means (middle); TD quantization (right)

tolerant in the sense that the derived tag is the same even if the input changes. The tag generation consists of three steps: extracting features or location-based parameters from the received location signals, quantizing the parameters with chosen step sizes, and mapping the quantized parameters into a binary string. The binary mapping process can be done using a hash function, which is easy to compute but hard to invert.

Error Models

To achieve optimal construction of fuzzy extractors, we study the various types of errors presented in location data. The most common error source is the thermal and atmospheric noise. The thermal noise, considered as white Gaussian, cannot be eliminated and always presents in all electronic devices and transmission media. Loran atmospheric noise, caused by lightning, is usually Gaussian but not always and can be impulsive if the lightning is local. Both thermal and atmospheric noises depend on the propagation path, distance between transmitter and receiver, quality of the receiver and the local noise floor, etc.

Another error source is bias. An example of seasonal bias in Loran signals is Additional Secondary Factor (ASF) that is the extra delay in propagation time due to the signals traveling over a mixed path: seawater path and land with various conductivities. This error introduces large seasonal variations in time-of-arrival (TOA), shown on the left of Figure 2. The four stations, Fallon, George, Middletown and Searchlight, are from Loran west coast chain, Group Repetition Interval (GRI) 9940. Fallon is the master station of GRI 9940 while the other three are the secondary stations. The monitor data was collected at Stanford University for 90 day period to observe seasonal variations on Loran signals. The delay can be significant and introduce a position error of hundreds of meters [6]. Thus ASF represents one of the largest error sources in Loran. Many factors affect ASF, including conductivity of soil, temperature, humidity, local weather, etc. Therefore, ASF varies both temporally and spatially, and this raises the difficulty modeling ASF over CONUS. The temporal component comes from all time varying aspects; while the spatial component takes into account the non-uniform ground conductivity and topography [7]. Many methodologies have been developed to mitigate ASF. In the previous study [8] we demonstrated two simple ideas: time difference and "previous day is today's correction". Time difference (TD) is the difference in TOAs between secondary stations and the master station; thus master station is used as a reference to remove the ASF bias. The second method is to use the previous day's ASF measurements as today's correction. This requires either the user receiver constantly monitors Loran data or a reference station that is nearby the users broadcasts previous day's ASF as a correction via a data channel. Both methods do not remove ASF completely. TD method has spatial decorrelation due to the different propagation paths of master and secondary stations. Previous day's correction suffers from the temporal decorrelation of ASF because previous day's ASF is different from today's ASF. In this paper we use the TD method to mitigate partial ASF temporal variations because it mitigates more ASF biases according to previous study [8]. The TD measurements from four stations are plotted in the middle of Figure 2.

In addition, quantization error, which is the difference between value of continuous parameter and the quantized value, can cause the system fail to reproduce correct location tag. The quantization error is usually correlated with thermal noise, atmospheric noise and seasonal biases discussed above. We cannot guarantee that the measurements are always in the middle of the quantization grid. The worst case is that the measurements lie on the boundary of the grid, shown in the right plot of Figure 2. The graph plots the TD measurements from Middletown with zero mean. The red dash lines represent the quantization grid boundaries. Even though the quantization step is chosen to overbound signal variations due to random noise and seasonal biases, the quantization error increases the likelihood failing to reproduce a location tag.

The last type error comes from the operations of RF system. Loran stations might be offline due to maintenance or other implementation issues. Wi-Fi access points (APs) are moving around by the users. Figure 3 illustrates the Wi-Fi AP availability or response rate, which is the percentage of time for a receiver is able to track an AP. The data was collected in an office building for four hours. Only the first AP can achieve 100% response rate thus a geotag will not be reproducible if more than 2 APs are used to derive the tag.

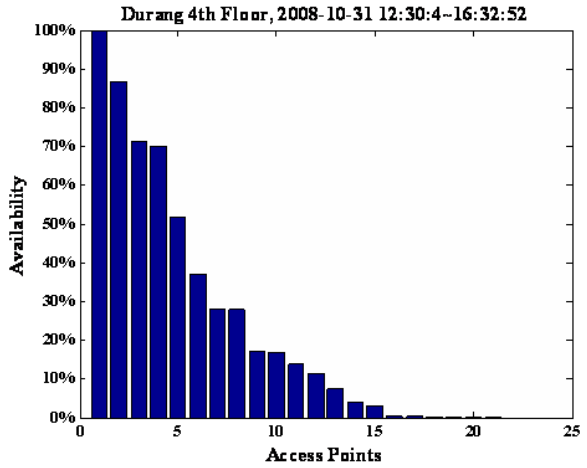

Figure 3. Wi-Fi access points response rate

Distance Metrics

The construction of fuzzy extractors depends on distance metrics of inputs. Thus it is important to analyze the error patterns and determine the proper distance metrics accordingly. In addition, the tag reproducibility under natural variations of RF signal is relative to the underlying metric in the space of location data.

Let x be the location parameter vector at calibration step, and q_x defined in Equation (1) represent the discrete parameter vector after quantization. The pair (x', q'_x) represents the parameter vectors at verification step. Δ is the quantization step vector. All the vectors are n-dimensional, where n is the number of parameters used to compute a tag. Quantized parameters q_x and q_x' are integers over Z but they are not necessarily positive. For instance, it is possible to result in a negative TD if the distance between the secondary station and a user is shorter than the distance between master station and the user.

$$q_x = \left\lfloor \frac{x_i}{\Delta_i} \right\rfloor_{i=1}^{n} \qquad (1)$$

The most common metric for the location parameter vector x is Euclidean distance. Euclidean metric fuzzy extractor is designed to tolerate the random noises, seasonal biases and quantization errors.

We consider the distance measure for last error type (missing parameter or offline transmitter) as Hamming metric. Hamming distance measures the number of different elements in quantized location parameter vectors at calibration and verification, q_x and q_x'.

Performance Metrics

The problem of deciding whether the received parameter is authentic or not, can be seen as a hypothesis testing problem. The task is to decide which of the two hypotheses H_0 (accepting as an authorized user) or H_1 (rejecting as an attacker) is true for an observed location measurement. Location-based security system can make two types of errors: 1) mistaking the measurements from the same location to be from two different locations and accepting hypothesis H_1 when H_0 is true, called false reject; and 2) mistaking the measurements from two different locations to be from the same location and accepting H_0 when H_1 is true, called false accept. Both false reject rate (FRR) and false accept rate (FAR) depend on the accuracy of the Loran receiver and the step size chosen to quantize location parameters. FAR only applies to white-listing applications while FRR can be a performance metric for both block-listing and white-listing applications.

FRR and FAR can be traded off against each other by varying the grid size. A more secure system aims for low FARs at the expense of high FRRs, while a more convenient system aims for low FRRs at the expense of high FARs.

FUZZY EXTRACTORS

Background and Definitions

The first approach of fuzzy extractor or error-tolerant cryptographic algorithm, called fuzzy commitment scheme, is proposed for biometrics by Juels and Wattenberg [9]. The scheme uses an error correcting code to handle Hamming distance. More approaches for Hamming distance, set difference, and edit distance are introduced in [10]. It also introduces a different error tolerant algorithm, called secure sketch.

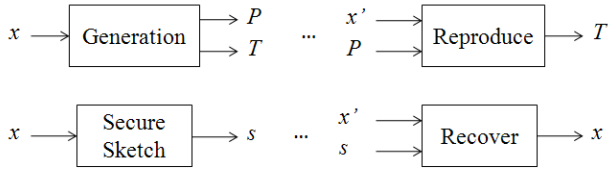

Figure 4. Fuzzy extractor (top); Secure sketch (bottom)

In this paper we follow the definition of fuzzy extractors in [10]. A fuzzy extractor works in two steps, illustrated in Figure 4. During calibration step, one runs an algorithm Gen in input $x \in M$ to generate a public value P and a location tag T, where M is a metric space of x. The public value P is stored for future use. An algorithm Rep is used to reproduce the tag T using P from noisy location vector x'. Fuzzy extractors are information-theoretically secure, thus we can use them for security applications without introducing additional assumptions [10]. A secure sketch also consists of two steps. A procedure SS produces s, called a sketch, using input x. Then given s and x' close to x, a procedure Rec can recover x. The sketch should not reveal much information about x. Unlike fuzzy extractors, a secure sketch recovers the original input x from noise while a fuzzy extractor reproduces location tag T from noisy input.

Definition 1. A fuzzy extractor is a tuple (M, t, Gen, Rep), where M is the metric space with a distance function dis, Gen is a generate procedure and Rep is a reproduce procedure, which has the following properties:

If Gen(x) outputs (T, P), then Rep(x', P) = T, whenever dis(x, x') $\le t$. If dis(x, x') $> t$, then there is no guarantee T will be outputted.

Definition 2. A secure sketch is a tuple (M, t, SS, Rec), where M is the metric space with a distance function dis, SS is a sketch generating procedure and Rec is a recover procedure, which has the following properties.

Rec(x', SS(x)) = x, if dis(x, x') $\le$ t. The sketch s is to be made public. We say the scheme is m-secure and the entropy loss of s is at most m. H(x) – H($x|s$) $\le m$. H denotes the entropy of a random variable.

In this paper we propose three fuzzy extractors based on Euclidean and Hamming metrics for inconsistent location parameters. We also introduce the secure sketch that Chang and Li [11] proposed for small set difference, which works the same as Hamming metric fuzzy extractor for location data. Although their approach was initially designed for biometric data, it can be adapted for location-based services with slight modifications.

Euclidean Metric Fuzzy Extractor

Let location vectors be n-dimensional in metric space M. We consider the distance measure for location-based parameters is L_∞ norm to be conservative. We normalized the measure using Δ, and the distance is defined as

$$dis(x, x') = \left(\max_i \frac{|x_i - x_i'|}{\Delta_i} \right)_{i=1}^{n} \quad (2)$$

The basic idea of this fuzzy extractor is to adjust the offsets between the continuous parameters and the discrete ones after quantization. The construction of the fuzzy extractor is shown as follows

$$\mathrm{Gen}(x) = \begin{pmatrix} T = hash(q_x) \\ P = \left(x_i - \Delta_i \left\lfloor \frac{x_i}{\Delta_i} \right\rfloor \right)_{i=1}^{n} \end{pmatrix} \quad (3)$$

$$\mathrm{Rep}(q_x', P) = \begin{pmatrix} q_x' = \left\lfloor \frac{x'_i - P_i + \frac{\Delta_i}{2}}{\Delta_i} \right\rfloor_{i=1}^{n} \\ T' = hash(q_x') \end{pmatrix} \quad (4)$$

If $dis(x, x') < \frac{1}{2}$, then quantized location vector q_x' can be reproduced, that is, T' = T. This claim defines the reproducibility of location tag. The quantization step Δ is a design parameter. The bigger the step, the more errors can be tolerated using this fuzzy extractor.

Shannon entropy is used to measure entropy loss of fuzzy extractors mathematically. We estimate the entropy loss or the mutual information between the conditional H($x|P$) and unconditional H(x) entropies. They are statistically independent if the mutual information is zero. Given $x = q_x + P$, let x' = $q_x + P - \delta$, where δ is the Euclidean difference between x and x' due to noises and biases. Our objective is to determine an upper bound on H($x|P$). By using the definition of conditional entropy [12], we obtain

$$H(x \mid P) = H(x) - H(\delta) \quad (5)$$

Thus, the entropy loss of public value P is H(δ). It depends on the probability distribution of x and the quantization step Δ. For the case n number of different location parameters, the total information leakage is

$$H(\delta) \le \sum_{i=1}^{n} \log(\Delta_i) \quad (6)$$

This equation assumes the location parameters are uniformly and independently distributed and provides an upper bound on the entropy loss. In practice, the entropy loss is small in comparison with $H(x)$. The measured entropy in a location tag also quantifies the amount of uncertainty from an attacker's point of view. The entropy in a location tag computed from quantized parameters is equal to $H(q_x|P)$. By the definition of q_x, q_x is independent of P; thus, P does not leak any information on q_x. Intuitively, this makes sense that knowing the offsets between x and Δ_x q_x, one cannot figure out the

user's quantization level exactly without further information.

Reed-Solomon Based Fuzzy Extractor

The approach achieves robustness against noises and biases by making use of error-correcting codes to recover changes measured by Hamming distance. Hamming distance, defined in Equation (7), measures the number of different elements between two strings or vectors. In addition, this fuzzy extractor deals with the problem caused by offline transmitters. Location tag can be reproduced even when there are missing parameters.

$$dis(x, x') = \sum_{i=1}^{n} x_i \oplus x'_i \quad (7)$$

We use Reed-Solomon (RS) error-correcting code to construct a fuzzy extractor to recover the changes of the quantized location parameters. Reed-Solomon coding is a well known forward error correction coding method that potentially for burst errors [13]. The key idea of the construction is to first create a polynomial by encoding the secrets, which is the tag in location-based security system. The next step is to project the quantized location parameters on the polynomial and randomly create chaff points to hide the polynomial. At last, the secrets can be recovered from the chaff points with adequate location parameters. The detailed construction is described as follows.

Calibration. Given $q_x = \{q_1, \ldots, q_n\}$,

1. A secret message is computed from a random generator.
2. The secret message can be hashed to get tag T.
3. The tag T is encoded to a vector c using Reed-Solomon code. The vector c has size n. The RS encoder (n, k) is chosen based on design criteria that the total number of errors t can be corrected is determined by $(n\text{-}k)/2$.
4. Construct mapping matrix or public information P. P has a size of $N \times n$, where N is the number of quantization levels of location parameters and determined by chosen quantization steps. For each column of P, locate the element of vector c based on each quantized location parameter. For instance, if $q_i = 20$, then $P(20, i) = c_i$. Figure 5 illustrate the formation of mapping matrix P. Populate the rest of the matrix using random numbers. This mapping matrix is then saved for future use.

$$\mathrm{Gen}(q_x, m) = \begin{pmatrix} T = rand \\ c = \mathrm{RS\,encode}(T) \\ P = mapping(c, q_x, T) \end{pmatrix} \quad (8)$$

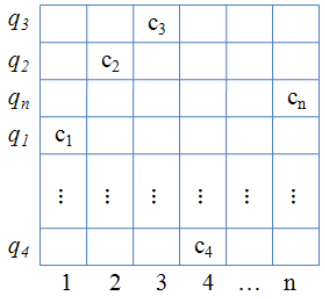

Figure 5. Mapping matrix construction

Verification. Given q_x' is a location parameter vector that has t or less than t elements different from q_x.

1. Given the mapping matrix P generated previously.
2. Obtain a vector c' using P and q_x'. If q_x' and q_x are identical, c' has the same elements as c. If attackers have no information on location parameters q_x, it is difficult to guess a vector c' that satisfies $\mathrm{dis}(c, c') \le t$ due to the large search space of mapping matrix. It is equivalent to brute-force attack.
3. Apply Reed-Solomon decode to compute T from c'. If $\mathrm{dis}(c, c') \le t$, the secret message can be recovered correctly; otherwise, the output would not be the same as T.

$$\mathrm{Rep}(q_x', P) = \begin{pmatrix} c' = mapping^{-1}(P, q_x') \\ T' = \mathrm{RS\,decode}(c') \end{pmatrix} \quad (9)$$

This approach makes use of the property of Reed-Solomon codes to tolerant t errors in the quantized location parameters. It is not fault-detective since users would not be able to find out whether the errors in received location parameters can be tolerated or not until computation of the tag. The entropy loss of this construction is $t\log N$. This results in the effective tag length is $(n\text{-}t)\log N$. Thus Hamming metric fuzzy extractors improve location tags' reproducibility at the expense of their entropy.

Secret Sharing Based Fuzzy Extractor

The third construction of fuzzy extractor is based on the idea of secret sharing. The scheme is a method of sharing secret S among a set of n participants. For any subset of k $(k \le n)$ participants, the secret S can be reconstructed. But a subset of less than m participants will fail to reconstruct S.

The distance metric in this construction is also Hamming. The input to the fuzzy extractor is quantized location vector q_x. The first step of construction is to create a polynomial $f(x)$, such that $f(i) = q_i$, $\forall\, i = 1, 2, \ldots, n$. The generation and reproduction procedures are as follows

$$\mathrm{Gen}(x) = \begin{pmatrix} f(i) = T + a_1 x + a_2 x^2 + \ldots + a_k x^k, \\ \text{where } a_1, a_2, \ldots, a_k \text{ are random numbers} \\ P = < i_1, \ldots, i_k >, s.t.\ f(i_j) = q_j \end{pmatrix}, (10)$$

$$\mathrm{Rep}(x', P) = \begin{pmatrix} \text{Reconstruct } f(i) \text{ using } P \text{ and } q'_x \\ T' = < f(0) > \end{pmatrix} \quad (11)$$

If $dis(q_x, q'_x) \le n - k$, the polynomial $f(x)$ can be reconstructed with the assistance of P thus tag T can be reproduced $T'=T$. The effective tag length is $k\log N$.

Fuzzy Extractor Modified from Chang and Li's Secure Sketch

Unlike fuzzy extractor, the secure sketch recovers the input at calibration using a sketch. The main security requirement is that the published sketch s should not reveal essential information on the inputs; otherwise, it will be compromised to attackers. Chang and Li proposed small secure sketch for point set difference [11], which can be applied to location-based security system with modification. Location data that is always contaminated by noises is not possible to recover exactly; therefore, we modify the secure sketch to fuzzy extractor. The distance measure is also Hamming metric in this approach. The constructions of the modified approach is as follows

Calibration. Given $q_x = \{q_1, \ldots, q_n\}$,

1. Construct a monic polynomial $p_1(x) = \prod_{i=1}^{n}(x - q_i)$.
2. Publish $P = <p_1(0), p_1(1), \ldots, p_1(2t\text{-}1)>$.

Verification. Given $P = <p_1(0), p_1(1), \ldots, p_1(2t\text{-}1)>$ and $q_x' = \{q'_1, \ldots, q'_n\}$,

1. Construct a new polynomial $q_1(x) = \prod_{i=1}^{n}(x - q'_i)$.
2. Compute $q_1(0), q_1(2), \ldots, q_1(2t\text{-}1)$.
3. Let $p_2(x) = x^t + \sum_{j=0}^{t-1} a_j x^j$, $q_2(x) = x^t + \sum_{j=0}^{t-1} b_j x^j$ be polynomials of degree t. Construct the linear equations with a_j's and b_j's as unknowns, and satisfy the following condition,

$$q_1(i)\, p_2(i) = p_1(i) q_2(i), \qquad \text{for } 0 \le i \le 2t - 1$$

4. Find one solution of the linear system.
5. Solve for the roots of the two polynomials $p_2(x)$ and $q_2(x)$. Let the roots be x' and y' respectively.
6. The recovered location parameter is $q_x' = (y \cup x') \setminus y'$.

The lemma 1 in Chang and Li's paper states the entropy loss due to enrollment is at most $2t\log N$ when $x \cap \{0, \ldots, 2t-1\} = \emptyset$. The assumption is that x does not contain any element from $\{0, \ldots, 2t\text{-}1\}$.

Combination Use of Fuzzy Extractors

We design the Euclidean metric fuzzy extractor to adjust the errors introduced by random noises and seasonal biases. The RS and secret sharing based fuzzy extractors can be used to reproduce location tag while location parameters are missing due to offline transmitters.

As noises and biases are always presented in RF signals, Euclidean fuzzy extractor should be applied all the time to minimize the impact of signal temporal variations and guarantee the reproducibility of location tags. Unlike noises and biases, errors due to missing parameters are infrequent. Users have their choices to use which fuzzy extractor. A combination use of Euclidean metric and Hamming metric fuzzy extractors can achieve more robustness in tags but the tradeoff is more entropy loss.

REPRODUCIBILITY ANALYSIS

In this section we examine and compare the performance of three fuzzy extractor constructions. The evaluation is based on user's FRR, attacker's successful rate FAR, and entropy loss.

All the three constructions improve consistency of location parameters thus reduce the FRR. Users' false reject depends on the variations of the parameters, the selected quantization step Δ and the quantization offset that is, how far off are the received parameters from the center of the quantization grid. The most desired scenario is the distribution of the parameter is exactly in the middle of the quantization grid (offset = 0) while the worst case is that the distribution lies on the boundary of the grid (offset = 0.5Δ), shown in Figure 6.

Figure 6. Quantization scenarios: best (left); worst(right)

Euclidean Metric Fuzzy Extractor

We first examine how the reproducibility of location tag improves using the Euclidean metric fuzzy extractor. The analysis is shown in Figure 7. The x-axis is the quantization steps in terms of standard deviation σ and the y-axis is the estimated FRR. The tag is computed from time difference (TD), signal strength (SS), and ECD (envelope-to-cycle difference) using the seasonal data from four west coast stations. As a result, there are 11 different location parameters.

To estimate FRR we take the first day of the 90-day data as calibration to compute a tag and the data from the rest

of 89 days for verification. The experimental FRR is the number of days, in which the tags are matched with the computed tag on day one, divided by 89. We observe that the estimated FRR is reduced by 84% after applying the Euclidean metric fuzzy extractor.

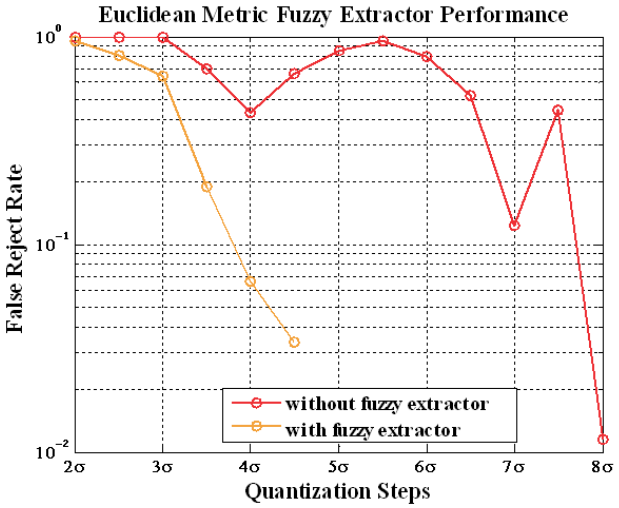

Figure 7. Euclidean metric FE performance improvement

From mathematical analysis the Euclidean metric fuzzy extractor rounds off the location measurements at verification step to the measurements at calibration step. A tag can be reproduced when the offset between the two measurements is less than a threshold, $\Delta/2$.

Reed-Solomon Error Correcting Review

This section provides a short review on Reed-Solomon codes as the FRR of multiple parameters using Hamming metric fuzzy extractor depends the RS error correcting performance. Let q be the code alphabet, that is, $q = 2^b$ the number of possible symbols, where b is the number of binary bits in a symbol. RS codes are non-binary codes. The decoding algorithm of RS codes is defined as bounded distance decoder, that is, only received sequences within a fixed designed bound of a valid codeword can be decoded and no errors can be corrected over the bound [13]. The representation of decoder operation is illustrated in Figure 8. Each white dot corresponds to actual RS codeword. The black dots enclosed by the circles are the possible received sequences, and can be mapped to the closest codeword.

The decoder can make two types of errors: the received sequence is decoded in an incorrect codeword, called undetected error; the received sequence is not decoded to any of the codewords, considered as a decode failure. Let the purple circle in Figure 8 represent the correct codeword. If the received sequence is within any other orange circle, the output codeword is incorrect and we consider it is undetected error. If the received sequence is in the gray region and not bounded by the circles, we consider it is a failure. For a RS(n, k), the minimum spacing between different codewords is n-k. The decoder can correct errors up to $t \le \lfloor (n-k)/2 \rfloor$. Let p be the symbol error or the error rate of one location parameter. The probabilities of incorrect decoding [13] are

$$\Pr\{\text{error or decorder failure}\} = \sum_{i=t+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \qquad (12)$$

$$\Pr\{\text{undetected error}\} = \frac{(q^k - 1)\sum_{i=0}^{t} \binom{n}{i} (q-1)^i}{q^n} \qquad (13)$$

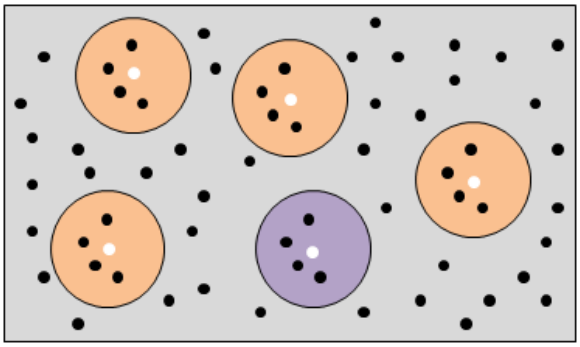

Figure 8. RS decoder representation

The evaluation of error detection can be classified into three different scenarios to compute three probabilities [14]: that the decided codeword is correct P_C, that the decoded codeword is an undetected error P_{UE}, and that fails to decode P_F. For RS(n, k) code, the minimum distance between codewords can be defined as $d_{min} = n - k + 1$. Let u represent the number of symbol errors, and $0 \le u \le n$. The three probabilities can be computed as follows

- For $0 \le u \le t$, $P_C = 1$, $P_{UE} = 0$ and $P_F = 0$ since the decoder can correct error up to t.
- For $t \le u \le d_{min}$-t, $P_C = 0$, $P_{UE} = 0$ and $P_F = 1$ since the received sequence is too far from any of the possible codeword thus considering as a decode failure.
- For $u \ge d_{min}$-t, $P_C = 1$, and $P_F = 1$-P_{UE}. The undetected error probability can be computed using Equation (13).

RS Based Hamming Metric Fuzzy Extractor

In practice, multiple parameters are used for robustness and security strength of location tag. More location parameters provide more information entropy, better precision, and increase the difficulty in predicting a tag. However, one drawback is that the FRR of the system is increased. The reproducibility comparison with and without a Hamming metric fuzzy extractor is illustrated in

Figure 9. Both cases use Euclidean metric fuzzy extractor to ensure data lying in the middle of quantization grids. We use 15 parameters to estimate FRR in this analysis thus $n = 15$. The overall FRR of Euclidean metric fuzzy extractor can be estimated as $1-\prod_{i=1}^{n}(1-p_i)$, where p_i is the error rate of one parameter or symbol error. For the combination of fuzzy extractors, overall FRR, shown in orange color, is estimated using Equation (12). We choose the number of errors t can be corrected in Hamming metric fuzzy extractor as 2. This results in that $k = 11$. The solid lines represent the analytical analysis while the dots are estimated using the same seasonal data mentioned in the previous section.

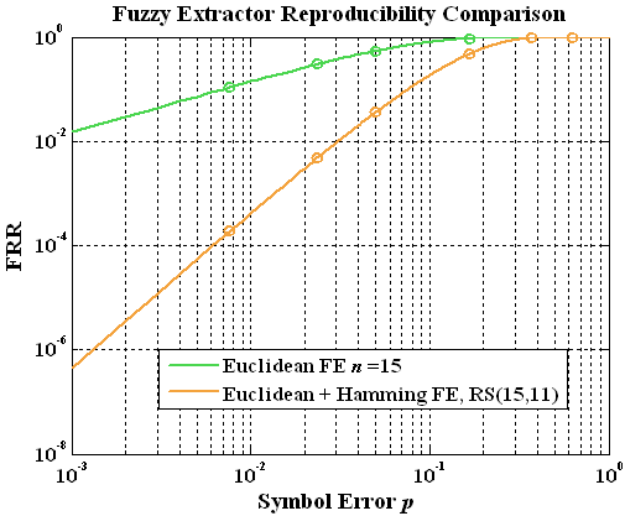

Figure 9. Performance of RS-based fuzzy extractor

Secret Sharing Based Hamming Metric Fuzzy Extractor

In this section we use Wi-Fi data shown in Figure 3 to evaluate the performance of secret sharing based Hamming metric fuzzy extractor. Only APs are used to map into a location tag for simplicity. Other location dependent parameters such as received signal strength (RSS) and response rate can also be used to derive Wi-Fi location tag. Euclidean metric fuzzy extractor should be applied if RSS is used for the geotag generation. The performance analysis is shown in Figure 10. The blue line illustrates the FRR of the derived geotag without using any fuzzy extractor while the red line represents the improved FRR after using secret sharing based Hamming metric fuzzy extractor. Figure 3 shows that there is only one AP, which has 100% response rate. It is understandable that the tag computed from 2 or more than 2 APs has low FRR or low reproducibility, as illustrated in Figure 10. From the analysis, we observe that the FRR is reduced by 85% when the number of APs used to derive a geotag is 4. The geotag has high FRR when the number of APs is greater than 4 even with secret sharing based fuzzy extractor. Wi-Fi tag reproducibility is location-dependent as the coverage of Wi-Fi APs is different from one location to another.

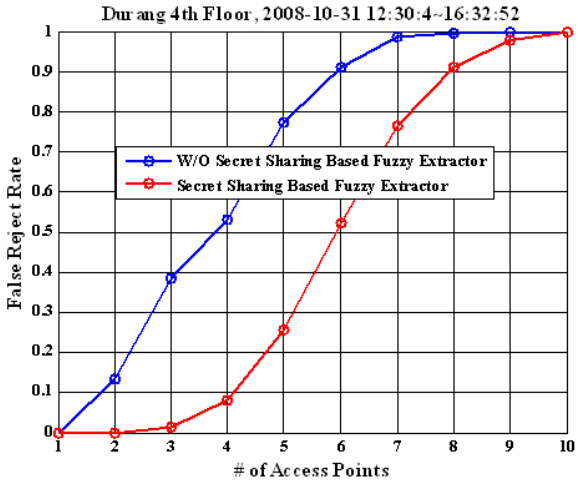

Figure 10. Performance of SS based fuzzy extractor

SECURITY ANALYSIS

An important measure of security in location-based services is false accept rate, which is probability of an attacker who achieves a desired location tag successfully in white-listing applications. In this section we study how fuzzy extractors affect the security performance in a location-based security system.

Euclidean Metric Fuzzy Extractor

FAR values depend on the distance between the physical locations of authorized user and attacker, variance and decorrelation of location parameters, and quantization steps selected by a user.

The Euclidean metric fuzzy extractor does not always increase or reduce attacker's false accept rate. The key idea of this fuzzy extractor is to round off user's measurements to a specific quantization level. For all the measurements x', if $\text{dis}(x\text{-}x') < \Delta/2$, quantized parameter q_x can be recovered. This claim is also true for attackers. For any attacker whose measurements are within $\Delta/2$ distance from x, the measurements can map into a correct grid. The following diagram explains the claim using two scenarios.

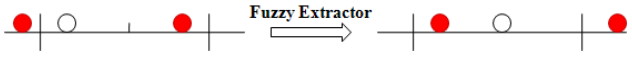

Figure 11. Security performance of Euclidean FE

Let the white ball represent authorized user, and two red balls represent two attackers. The two long vertical lines bound a quantization grid with a step Δ. The absolute distance between the white ball and the center of the grid

is the public value P. After applying fuzzy extractor, the white ball is shifted to the center, and the left red ball is also moved inside the grid while the other red ball is moved outside. From this diagram we see the probability that an attacker achieves a correct quantized value depends on the closeness of the user and attacker. In addition, the more parameters used to compute a tag, the higher the difficulty to break the tag for attackers.

RS Based Hamming Metric Fuzzy Extractor

The false accept rate using RS based fuzzy extractor depends on the RS decoder performance. As mentioned earlier, the decoder replies on a bounded distance to map into the closest codeword and the bound is determined by n and k. We consider the FAR is a type of undetected error. From an authorized user's point of view, undetected error is that the received sequence is mapped into any of the incorrect codewords, which are the orange circles in Figure 8. From an attacker's point of view, undetected error is that the received sequence is mapped into the user's codeword, the purple circle in Figure 8. The total FAR of multiple parameters using RS based fuzzy extractor should be analyzed in two conditions: $\mathrm{dis}(q_x, q_y) \le \mathrm{t}$ and $\mathrm{dis}(q_x, q_y) > \mathrm{t}$, where q_x is the quantized location vector of an authorized user and q_y is the vector of an attacker. The FAR estimation is derived from Equation (13), shown as follows

$$\mathrm{FAR}_{\mathrm{RS}} = \begin{cases} 1 & \mathrm{dis}(q_x, q_y) \le t \\ \dfrac{q^k \sum_{i=0}^{t} \binom{n}{i}}{q^n} & \mathrm{dis}(q_x, q_y) > t \end{cases} \quad (14)$$

When the Hamming distance between user and attacker's vectors is less than or equal to t, both user and attacker can reproduce the desired codeword. However, when the Hamming distance is greater than t, the probability of error is fixed, and depends on n and t only. For instance, if we use 15 location parameters $n = 15$ and allow 2 errors $t = 2$, the probability that an attacker receives a desired codeword is approximately 0.00185.

SS Based Hamming Metric Fuzzy Extractor

This FAR analysis of secret sharing and secure sketch based fuzzy extractors is similar to that of the RS based. Desired location tag can be achieved if the number of errors is less than a certain threshold. For the secret sharing based, the threshold is n-k; for Chang and Li's approach, the threshold is t, which is a design parameter. If attacker's location parameters have more errors than the threshold, the false accept rate is low.

TRADEOFF

As mentioned in the previous section, multiple location parameters provide more security strength in the derived location tags; however, it reduces reproducibility of tags and reliability of authorized users. The security strength of each parameter is different in terms of information entropy, reproducibility and false accept rate.

We first study the tradeoff between false reject rate of authorized users and false accept rate of attackers as the number of location parameters increases. Loran data is used for this study. The analysis is illustrated in Figure 12. The error rates also depend on the selected quantization steps. The two orange curves represent the FARs with quantization steps of 3σ and 6σ while the two green curves indicate the corresponding FRRs. The error rates of 3σ quantization step is shown in circle marker and solid line. The triangle marker and dashed lines indicate the error rates of 6σ step. The seven selected parameters are TD from George, Middletown and Searchlight, and the signal strength from all four stations in GRI 9940. The error rates are calculated using the seasonal data shown in Figure 2. To estimate FRR, we think ourselves as authorized users. We take the first day of seasonal data as calibration measurements the rest days of data as verification measurements, and estimate the percentage of time that the location tags at calibration and verification steps match. On the other hand, to estimate FAR, we think ourselves as attackers, whose parameters are 1σ off from the authorized user's parameter values. We observe that the tradeoff between false reject rate of users and false accept rate of attackers with varying number of location parameters and quantization step sizes.

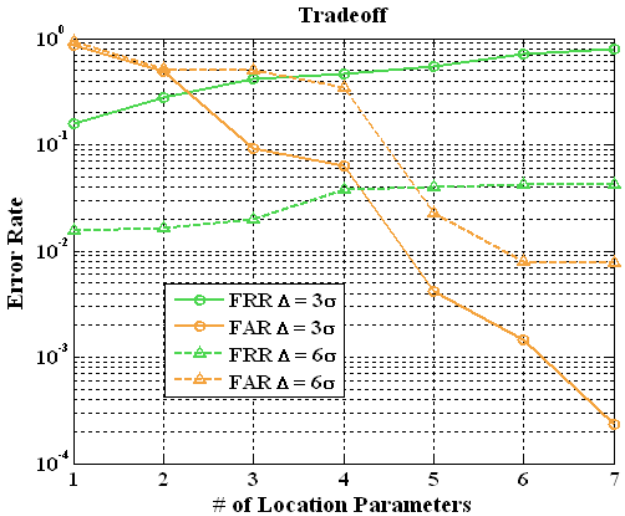

Figure 12. Tradeoff between FAR and FRR

The second tradeoff study, illustrated in Figure 13, is between reproducibility and information entropy. The more entropy in the location parameters, the longer the

Table 1. Comparison of fuzzy extractors

	FRR	FAR	Entropy Loss (bits)
Euclidean Metric	$1-\prod_{i=1}^{n}(1-p_x)$	$\prod_{i=1}^{n} p_y$	No loss on location tag $\sum_{i=1}^{n}\log\Delta_i$
RS-based	$\sum_{i=t+1}^{n}\binom{n}{i}p_x{}^i(1-p_x)^{n-i}$	$\begin{cases}1 & dis(q_x,q_y)\le t\\ \dfrac{q^k\sum_{i=0}^{t}\binom{n}{i}}{q^n} & dis(q_x,q_y)>t\end{cases}$	$\sum_{i=1}^{t}\log N_i$
Secret sharing based	$1-\prod_{i=1}^{k}(1-p_x)$	$\prod_{i=1}^{k} p_y$	$\sum_{i=1}^{n-k}\log N_i$
Chang and Li's	$\sum_{i=t+1}^{n}\binom{n}{i}p_x{}^i(1-p_x)^{n-i}$	$\begin{cases}1 & dis(q_x,q_y)\le t\\ 0 & dis(q_x,q_y)>t\end{cases}$	$\sum_{i=1}^{2t}\log N_i$

tag we can derive. The longer tag means it is harder to use brute-force attack to break. Brute-force attack is a method to defeat a cryptographic scheme by trying all the possible combinations of a binary key. With a long enough tag, brute-force attack is not threatening to the system.

The left axis is the FRR of authorized user and the right axis is the sum of information entropy of location parameters. The quantization step used in this analysis is 6σ. We observe that 56-bit tag can be achieved from 7 parameters with this particular quantization step. The time to break a 56-bit tag with 1 operation per 1 μsec is 1142 years. With a supercomputer that performs 10^6 operations per 1 μsec, it takes only 10 hours to break a 56-bit tag [15].

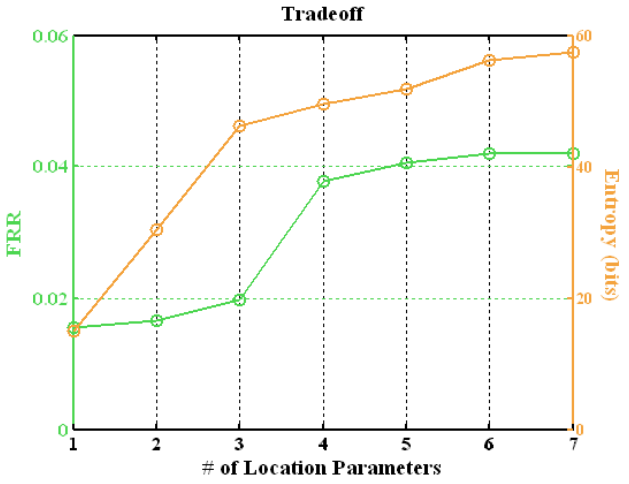

Figure 13. Tradeoff between FRR and entropy

The geotag FRR and average cell size can be traded off against each other by varying the number of parameters used to compute the getoag, shown in Figure 14. Wi-Fi data is used in this analysis. A cell size is defined as the dimension of quantized spatial cells, which all have different geotag. The cell size depends on the separation between the calibrated locations and the spatial decorrelation of location parameters.

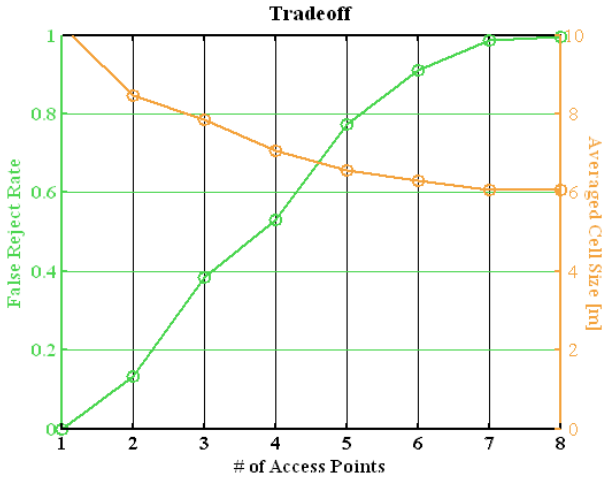

Figure 14. Tradeoff between FRR and average cell size

CONCLUSION

We proposed location-based security services using RF signals: in which location is used as a validation to restrict or deny certain action in security applications. Verification tags are computed from location information that is obtained from a location sensor. The location tag is not a replacement but builds on the conventional authentication schemes. This location-based service can be applied to many applications, such as DMP, inventory control and data access control.

Euclidean metric fuzzy extractor was developed for noise, biases and quantization errors to achieve high reproducibility of location tag. Reed-Solomon based and secret sharing based fuzzy extractors were designed for the scenario that RF transmitter is offline. The entropy loss of Hamming metric fuzzy extractor is more than the Euclidean metric one. The performance comparison of proposed fuzzy extractors is shown in Table 1.

Adequate quantization step should be chosen to achieve reasonable tag reproducibility. FAR, FRR and information entropy can be traded off by varying the quantization steps and number of location parameters. A more secure system requires smaller quantization step and more number of parameters while a more convenient system prefers larger step and less parameters. Users of location-based security service have the flexibility to select appropriate design parameters based on their applications and performance requirement.

ACKNOWLEDGMENTS

The authors would like to thank Mitch Narins of the FAA, Loran Program Office for supporting this effort. In addition, we would like to thank Dr. Greg Johnson, Ruslan Shalaev and Christian Oates from Alion Science & Technology for providing us the Stanford Seasonal Monitor data collection equipments. Finally, thanks go to US Coast Guard (USCG) Loran Support Unit (LSU) for the use of their equipment in our testing.

REFERENCES

[1] J. Hruska. "Microsoft patent brings Miss Manners into the digital age". June 11, 2008.

[2] B. Schneier. "Kill Switches and Remote Control". A blog covering security and security technology. July 1, 2008.

[3] L. Scott and D. Denning, "Location Based Encryption & Its Role In Digital Cinema Distribution", *Proceedings of ION GPS/GNSS 2003*, pp288-297.

[4] D. Qiu, S. Lo, and P. Enge, "Geoencryption Using Loran", *Proceeding of ION NTM 2007*.

[5] D. Qiu, "Security Analysis of Geoencryption: A Case Study using Loran", *Proceeding of ION GNSS 2007*.

[6] S. Lo, R. Wenzel, G. Johnson, and P. Enge, "Assessment of the Methodology for Bounding Loran Temporal ASF for Aviation". *Proceeding of ION NTM 2008*.

[7] P. Swaszek, G. Johnson, R. Hartnett, and S. Lo, "An Investigation into the Temporal Correlation at the ASF Monitor Sites". *Proceedings of ILA 36th Annual Meeting 2007*.

[8] D. Qiu, S. Lo, and P. Enge. "A Measure of LORAN Location Information", *Proceeding of ION PLANS 2008*.

[9] A. Juels and M. Wattenberg. "A Fuzzy Commitment Scheme". *Proceedings of ACM Conf. on Computer and Communications Security*, pp28-36, 1999.

[10] Y. Dodis, L. Reyzin, and A. Smith. "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data". *Eurocrpt'04*, vol. 3027 of LNCS, pp523-540. Springer-Verlag, 2004.

[11] E. Chang and Q. Li. "Small Secure Sketch for Point-set Difference". *Cryptology ePrint Archive, Report 2005/145 (2005).*

[12] T. Cover, Elements of Information Theory. John Wiley & Sons, Inc. 2001.

[13] A. Daraiseh and C. Baum. "Decoder Error and Failure Probabilities for Reed-Solomon Codes: Decodable Vectors Methods". *IEEE Trans. Commun. 46(7), July 1998, pp. 857-859*.

[14] K. Carroll, A. Hawes, B. Peterson, K. Dykstra, P. Swaszek, and S. Lo. "Differential Loran-C". *Proceeding of the European Navigation Conference GNSS 2004, Rotterdam, The Netherlands*.

[15] L. Williams, "A Discussion of the Importance of Key Length in Symmetric and Asymmetric Cryptography". *SAN Institute 2000-2002*.