Physical Layer Authentication via Fingerprint Embedding: Min-Entropy Analysis

Invited Presentation

Jake Bailey Perazzone[†], Paul L. Yu[*], Brian M. Sadler[*], Rick S. Blum[†]

Abstract— **One of the difficulties of implementing and analyzing algorithms that achieve information theoretic limits is adapting asymptotic results to the finite block-length regime. Results on secrecy for both regimes utilize Shannon entropy and mutual information as metrics for security. In this paper, we determine that Shannon entropy does not necessarily have equal utility for wireless authentication in finite block-length regimes with a focus on the fingerprint embedding framework. Then, we apply a new security performance metric to the framework that is linked to min-entropy rather than Shannon entropy and is similar to cheating probability used in the literature. The metric is based upon an adversary's ability to correctly guess the secret key over many observations using maximum likelihood decoding. We demonstrate the effect that system parameters such as the length of the key and the identification tag have on an adversary's ability to attack successfully. We find that if given a large key, it is better to use it all at once, than to use some and then renew the key with the remaining bits after a certain number of transmissions.**

I. Introduction

Authentication, in addition to secrecy and privacy, is a fundamental part of the security for any communication system. It is the process of verifying the source and integrity of a received message to protect communications between trusted parties. In wireless communications, authentication is doubly important due to the open nature of the medium which enhances an adversary's ability to eavesdrop on communications and transmit messages of their own. While existing cryptographic authentication methods are computationally secure[1], the wireless medium allows for the possibility of achieving information-theoretic security, which has much stronger guarantees, by taking advantage of channel differences between the legitimate and illegitimate parties. The concept of information-theoretic secrecy originates from the work of Shannon [1] and Wyner [2].

Their seminal works utilize Shannon's metric of entropy and mutual information to quantify the secrecy of the message being transmitted. More specifically, Wyner's work on the wire-tap channel and Csiszár and Körner's generalization [3] derive rate regions with a secrecy requirement in which the mutual information between an adversary's observation and the message must be made arbitrarily small as the block-length goes to infinity. Authentication, on the other hand, doesn't have a direct entropy metric that can be used to quantify a given scheme's performance. Instead, cheating probability, or the probability that an adversary can deceive the legitimate receiver, is used. Bounds on the cheating probability using entropy or mutual information, however, can be found in [4], [5], and [6] or in terms of an achievable rate region in [7]. The presence of entropy in these bounds and the importance of the shared key in authentication partially motivated our approach [8] of using key equivocation, i.e. conditional entropy of the key given an observation, to quantify authentication and key security.

The bounds, however, are of limited practical use for the fingerprint embedding authentication framework since one is for the noiseless case [4], one is only a lower bound [5], and one is for the asymptotic block-length regime [6]. In this paper, we investigate the utility of Shannon entropy in authentication and suggest using min-entropy [9] in an alternative metric. We also develop a means to measure an adversary's ability to impersonate a legitimate transmitter to show how much information-theoretic authentication security can be achieved in practice. More specifically, we directly calculate the cheating probability of an adversary with infinite computational resources over the course of multiple authenticated transmissions in the fingerprint embedding framework. The analysis is based on an adversary's ability to infer the key using a maximum likelihood decoder from their observations of legitimate communications and then sending a message authenticated using the most likely key.

This paper is organized as follows. Section II provides an overview of the fingerprint embedding authentication framework while Section III discusses the use of Shannon entropy and min-entropy as security metrics for authentication. Section IV presents the revised security analysis and results for the framework.

II. Fingerprint Embedding Framework Overview

The fingerprint embedding authentication framework [8] is designed to take advantage of the physical layer by protecting an authentication tag in noise to achieve a degree

[†] J.B. Perazzone and R.S. Blum are with Lehigh University, Bethlehem, PA. {jbp215,rblum}@lehigh.edu

[*] P.L. Yu and B.M. Sadler are with the US Army Research Lab, Adelphi, MD. {paul.l.yu.civ,brian.m.sadler6}@mail.mil

This material is based upon work partially supported by the U. S. Army Research Laboratory and the U. S. Army Research Office under grant number W911NF-17-1-0331 and by the National Science Foundation under grants ECCS-1744129 and CNS-1702555.

[1] Computational security refers to when a system is secure due to prohibitive computational complexity required to break them, that is, modern computers cannot efficiently infiltrate in a practical amount of time.

978-1-7281-1151-3/19/$31.00 ©2019 IEEE

of information theoretic secrecy. A block diagram of the authentication framework can be found in Figure 1.

In our model, a legitimate transmitter (Alice) wishes to have her communications with a legitimate receiver (Bob) authenticated in the presence of a single adversary (Eve) who may try to impersonate Alice by sending messages of her own to Bob. To facilitate authentication, Alice and Bob both agree upon a shared key $\boldsymbol{k}$ that is drawn uniformly from the set of all κ-bit keys $\mathcal{K}$ and is kept secret from Eve before communication begins. When Alice wishes to transmit a message $\boldsymbol{s}$ to Bob, she proceeds by first generating a tag $\boldsymbol{t} = g(\boldsymbol{s}, \boldsymbol{k})$ following the hash-based message authentication code (HMAC) protocol [10]. We assume that the tag generating function $g(\cdot, \cdot)$ is deterministic, but where outputs were selected uniformly over the tag space.

Next, Alice superimposes the tag on the message waveform

$$\boldsymbol{x} = p_s \boldsymbol{s} + p_t \boldsymbol{t} \ , \tag{1}$$

where the power allocation of the message and tag, p_s and p_t, respectively, are non-negative scalars that designed such that $p_s^2 + p_t^2 = 1$. We assume that $\boldsymbol{x}$ is a length L complex-valued vector of iid symbols with zero mean and unit variance in the form of a desired modulation scheme, e.g. QPSK, QAM. Finally, she sends $\boldsymbol{x}$ over two additive white Gaussian noise (AWGN) channels to Bob and Eve, respectively

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{w}_{\mathrm{b}} \tag{2}$$

$$\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{w}_{\mathrm{e}} \, , \tag{3}$$

where $\boldsymbol{w}_{\mathrm{b}}$ and $\boldsymbol{w}_{\mathrm{e}}$ are L-length random complex Gaussian noise vectors with zero mean and variance σ_{b}^2 and σ_{e}^2, respectively. Note here that, unlike for non-zero secrecy capacity in the wire-tap channel, we do not necessarily assume that $\sigma_{\mathrm{b}}^2 < \sigma_{\mathrm{e}}^2$. We do assume that Eve is not actively jamming or interfering with Alice's transmissions.

Bob receives $\boldsymbol{y}$ from the channel and first decodes the primary message $\hat{s}$ before determining its authenticity. Assuming he decodes without error, he obtains a noisy version of the tag $\tilde{\boldsymbol{t}}$ by removing the primary message's contribution to $\boldsymbol{y}$. Bob, knowing $\boldsymbol{k}$ and obtaining $\hat{s}$, computes the expected tag using the same function $g(\cdot, \cdot)$ as Alice which produces the same tag as in Eq. (1). He then uses a matched filter and threshold designed using the Neyman-Pearson lemma to test for the presence of the expected tag in $\tilde{\boldsymbol{t}}$. In the hypothesis test, H_0 corresponds to the expected tag not being present while H_1 corresponds to the expected tag being present. Authentication is successful when Alice sends an appropriately tagged message and Bob selects H_1. The false alarm α used to determine the threshold corresponds to the situation where Bob incorrectly authenticates a message containing an invalid tag.

For full details on the fingerprint embedding authentication framework, please refer to [8].

III. Entropy as a Security Metric in Authentication

Typical treatments of security for authentication systems approach performance in terms of the success probability of the adversary. In this case, success for the adversary occurs when a legitimate receiver erroneously accepts one of their false messages. The way in which the attack is performed is frequently broken down into two types, impersonation and substitution. The two attacks differ in that in an impersonation attack, the adversary sends a message hoping to get it authenticated before a legitimate source transmits whereas for a substitution attack, the adversary first intercepts a transmission, and then replaces it with their own message. The maximum success probability between the two attacks is frequently called the cheating probability which is used as the security performance metric for authentication.

Thus, an authentication system must protect against both attacks to minimize the cheating probability. Due to the similarity between the two attacks, we assume that the impersonation attack subsumes the substitution attack and consider just the former. In the case of the framework presented in Section II, impersonation attacks are first limited by the designed false alarm rate α and the size of the key space. This is due to the fact without knowledge of the key, the adversary is limited to uniformly guessing a key/tag to superimpose and attacks with which has a probability of $\max\{\alpha, 1/|\mathcal{K}|\}$ of succeeding. Once the adversary gains access to additional observations, they can be more intelligent with their attack strategy by making a better key/tag selection. Thus, attacks are limited by the amount of key leakage in each transmission.

In previous analysis [8], we measured the amount of key leakage using key equivocation which is simply the entropy of the key conditioned on the adversary's observations. Besides its use in the cheating probability bounds found in [4], [5], and [6], the direct link from key equivocation to the adversary's capabilities was never fully explored. Thus, we propose a new metric to quantify security of the fingerprint embedding authentication framework, or any HMAC-based authentication scheme, that is more in line with cheating probability in the literature. We define the metric as the probability that Eve successfully impersonates Alice when attacking with the most likely key based on her observations which is calculated as

$$P_S = P_K P_D + (1 - P_K)\alpha \, , \tag{4}$$

where P_K is Eve's probability of choosing the correct key and P_D is the probability that Bob accepts a correctly tagged message. Equation (4) takes into account the possibly that even if Eve does obtain the correct key, she is still limited by the power of Bob's hypothesis test P_D. Additionally, even if she guesses the wrong key, there is still a chance she is successful due to possible type I errors (limited by α) in Bob's test. With α chosen by Bob and the value for P_D obtained from the process outlined in Section II [11],

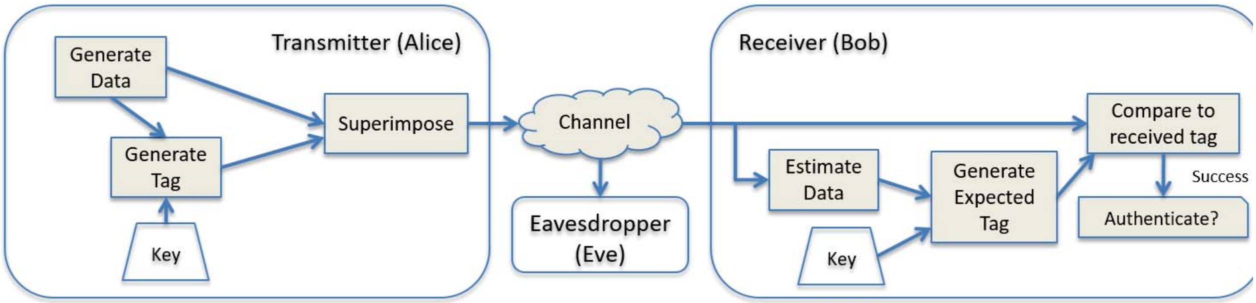

Fig. 1. System diagram of the physical layer fingerprinting authentication framework. A legitimate transmitter-receiver pair share the same secret key that is then used to create an identifying tag that enables authentication.

all that remains to calculate is P_K. In the next section, we demonstrate the merits of using min-entropy (defined in Eq. (8)) in order to define

$$P_K = 2^{-H_\infty(K|\boldsymbol{Y})}. \tag{5}$$

A. Min-entropy as a Metric

Key equivocation, or conditional Shannon entropy,

$$H(K|Y) = \sum_{x \in \mathcal{X}} P_Y(Y) H(K|Y = y). \tag{6}$$

allows users to examine the change in entropy of their shared secret key as an adversary intercepts an increasing number of transmissions between legitimate parties. Papers such as [8] and [12] use key equivocation to define the security of their authentication frameworks. Conditional Shannon entropy, however, is insufficient in fully characterizing the vulnerability of a key or the security of an authentication system [13][14]. The insufficiency partially comes from the lack of an intuitive explanation or practical meaning of the obtained entropy value and how it relates to an adversary's ability to successfully impersonate a legitimate party. While the two extremes of conditional Shannon entropy have clear meanings, maximum Shannon entropy indicates no leaked information (perfect secrecy) and zero entropy indicates that the key is fully leaked, the intermediate values' operational meanings are less apparent. Additionally, since Shannon entropy is an average measure of information, it can lead to misleading conclusions. For example, suppose we have the distribution of 128 keys conditioned on an observation where one key has a probability of 1/10 and the rest are equiprobable. Then, the Shannon entropy is 6.7588 which is close to the maximum 7 implying that the key is fairly secure when in reality one key stands out.

A similar information-theoretic metric that better considers the extrema of a distribution is min-entropy, which is defined as

$$H_\infty(X) \triangleq \min_{x \in \mathcal{X}} -\log_2 (P_X(x)) = -\log_2 \left(\max_{x \in \mathcal{X}} P_X(x) \right). \tag{7}$$

It is simply the negative logarithm of the highest probability in the distribution. As suggested in [13] and [14], min-entropy may be a better alternative measure for key secrecy due to the fact that $2^{-H_\infty(K)}$ is the probability of correctly determining a random variable, e.g. the key, on the first try using an optimal guessing scheme. This metric is applicable to authentication because it quantifies the probability of an adversary correctly guessing the key to attack with and thus directly affects the success of an impersonation attack.

Given an observation $\boldsymbol{y}$, the metric becomes $2^{-H_\infty(K|\boldsymbol{Y}=\boldsymbol{y})} = \max_K P(K|\boldsymbol{Y} = \boldsymbol{y})$. This is essentially equivalent to the success probability of using a maximum likelihood (ML) decoder to decode the key, something we will take advantage of in the subsequent section. We must be careful, though, about using conditional min-entropy $H_\infty(K|\boldsymbol{Y})$. It, as well as other conditional Rényi entropies, do not have generally accepted definitions [15]. For example, using the same formulation that is used for conditional Shannon entropy breaks the monotonicity property required for an information measure which is undesirable. Nonetheless, analysis in [13] and [14] uses the conditional Shannon entropy definition for conditional min-entropy. Here, though, we will follow the formulation suggested in [15]

$$H_\infty(Y|X) = -\log \left(\sum_{y \in \mathcal{Y}} P_Y(y) \left(\max_{x \in \mathcal{X}} P_{X|Y}(x|y) \right) \right). \tag{8}$$

This formulation is advantageous since $2^{-H_\infty(K|\boldsymbol{Y})}$ is the average performance of using an ML decoder to guess the key, giving us P_K. Additionally, it satisfies the monotonicity property and (weak) chain rule required for a measure (see [15]). The metric $2^{-H_\infty(K|\boldsymbol{Y})}$, as defined here, is equivalent to the substitution attack upper bound given in [6].

B. Guessing Entropy

Note that in any authentication system, the adversary may have more than one chance to guess the key. In such case, even if the first attack using the most likely key fails, they can try again using the next most likely key and so on until the correct key is reached and authentication is achieved (or the legitimate users recognize the failed attempts and act accordingly). Thus, min-entropy alone may not be fully sufficient. In [13] and [14] this situation is addressed by the proposal of using guessing entropy

$$E\left[G(X)\right] = \sum_{i=1}^{|\mathcal{X}|} i p_i^*$$

where p^* is the probability distribution on X ordered from highest probability to the lowest. Guessing entropy indicates the expected amount of guesses required before the correct value is obtained. This formulation assumes that there is no feedback at each guess which is typically valid in an authentication setting since the adversary usually does not know if a tag is accepted or not.

The conditional form of guessing entropy,

$$E\left[G(K|Y)\right] = \sum_{y \in \mathcal{Y}} P_Y(y) E\left[G(K|Y = y)\right],$$

is suggested in [16] to connect the idea of guessing entropy to security. With such a metric, a secure system requires the conditional guessing entropy to be sufficiently large. The users would design their system such that it must take the adversary many guesses to arrive at the correct secret key. We take a similar approach to the multiple key guess problem by comparing it to list decoding in Section IV-C.

The next section details how to determine P_K for both a single guess and a multiple guess attack. We determine P_K by examining the probability that an adversary can guess the key using a maximum likelihood decoding scheme via a random coding analogy.

IV. Security Analysis

A. Random Coding Analogy

Based on Eq. (5), we assume that Eve uses a maximum likelihood decoder to choose a key to attack with. In order to compute the probability P_K of choosing the correct key, we consider the key to tag mapping as a random code due to the uniform output and analyze the error performance. In this analogy, the tag generating function is an encoder that maps messages (keys) to codewords (tags). Eve wants to reliably decode the received codeword to determine which key is being used by Alice and Bob. Since the tag generating function is public, Eve has access to the codebook and can perform maximum likelihood decoding. Since we assume that Eve has unlimited computational abilities, the complexity of such a decoder is not considered. Analysis in [12] takes a similar approach, but models the tag generator as an equidistant code in order to achieve results for the case when an adversary has more than one observation.

When extending to multiple observations, Eve benefits from the fact that Alice repeatedly uses the same key. Even though the codeword changes with each transmission, Eve can modify her codebook to treat each observation as part of one long transmitted codeword from a single key. In the modified codebook, each codeword is a concatenation of tags corresponding to a specific key and all observed primary messages, i.e.

$$g(\hat{\boldsymbol{s}}_1, \boldsymbol{k}_i) || \cdots || g(\hat{\boldsymbol{s}}_{N_o}, \boldsymbol{k}_i) \, \forall \boldsymbol{k}_i \in \mathcal{K},$$

where N_o is the number of observations. The concatenated observed codeword is then

$$\tilde{\boldsymbol{t}}_1 || \cdots || \tilde{\boldsymbol{t}}_{N_o}.$$

She guesses a key by using maximum likelihood decoding on the new codebook.

B. Eve Performance Analysis

With the random code analogy in hand, we can use new and existing random coding results to analyze Eve's probability of correctly guessing the key using a maximum likelihood decoder averaged over all possible random codes. We show two approaches to calculating this probability. The first is a simplification of the channel to be binary symmetric where the transition probability is determined from the tag SNR $\frac{p_t^2}{\sigma_e^2}$ and the modulation scheme used. In this channel model, the decoder hard-decodes the received symbol and then selects the codeword in the codebook whose Hamming distance is the closest to the received codeword where ties are broken by random selection. Theorem 1 gives the average success probability for such a decoder of a random code over a binary symmetric channel (BSC).

Theorem 1 ([17]): The probability of correctly decoding the key over a BSC, averaged over all tag mappings, is

$$P_{K,\mathrm{BSC}} = \sum_{i=0}^{n} \binom{n}{i} p_e^i (1-p_e)^{n-i} \cdot \left(\sum_{t=0}^{|\mathcal{K}|-1} \frac{1}{t+1} \binom{|\mathcal{K}|-1}{t} \left(2^{-n}\binom{n}{i}\right)^t \left(2^{-n} \sum_{j=i+1}^{n} \binom{n}{j}\right)^{|\mathcal{K}|-1-t} \right), \tag{9}$$

where p_e is the BSC transition probability and n is the tag length in bits.

Proof: See [17, Theorem 2]. ■

This expression, however, is difficult to compute for large block-lengths due to the large number of binomial coefficients, although bounds are available. The probability of success for multiple observations is found by replacing all instances of n with nN_o to reflect the longer codeword lengths.

The second approach to analyzing Eve's decoding capabilities is to assume her maximum likelihood decoder is a bank of matched filters tuned to each key/tag and decodes by selecting the key that has the highest matched filter output. Theorem 2 gives the success probability of such a decoder over an AWGN channel.

Theorem 2: Assuming an AWGN channel, the probability that the correct key corresponds to the largest matched filter output is

$$P_{K,\mathrm{AWGN}} = \int_{-\infty}^{\infty} \phi\left(\frac{t-L}{\sqrt{\frac{L\sigma_e^2}{2p_t^2}}}\right) \Phi\left(\frac{t}{\sqrt{\frac{L}{2}+\frac{L\sigma_e^2}{2p_t^2}}}\right)^{|\mathcal{K}|-1} dt \tag{10}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the PDF and CDF of the standard normal distribution.

Proof: In [11], the distributions for the matched filter outputs for both a correct codeword and a random codeword are presented. Using those distributions and the fact that the codewords are pair-wise independent, we can simply calculate the probability that all of the $|\mathcal{K}|-1$ incorrect codewords matched filter outputs are less than the correct one's output to obtain (10). Unlike in the BSC, this formulation is continuous, so we do not need to consider the case where two matched filter outputs are equal. ■

The probability of success for multiple observations is found by replacing all instances of L with LN_o to reflect the longer codeword lengths. This formula is more efficient to compute than (9) since it does not contain any binomial coefficients.

The probability of correctly guessing the key using a maximum likelihood decoder (Equation (10)) versus the number of observations is plotted in Figure 2 over different key lengths and compared to the designed α. Markers on the plot show that the number of observations at which the probability of Eve guessing the key surpasses α is not linear with key length. This indicates that continual use of a longer key is better than using some of the key until Eve's probability reaches α and then switching to using the remaining key material. For example, suppose Alice has 1024 bits of secret key to use for authentication. If she uses the first 512 bits to authenticate, she can do so for about 13 transmissions before needing to switch to the other 512 bits for a total of 26 transmissions under α. If she instead uses all 1024 bits, then she can transmit approximately 29 times before reaching a compromising position. This is most likely due to the fact that some secrecy in the key remains when it is discarded in the former scheme and doing so twice instead of once is inefficient. In this case, the amount of bits wasted is $\log_2 \alpha \approx 13$ bits.

Figure 3 illustrates the number of observations Eve requires for her attack success probability (Eq. (4)) to exceed Alice and Bob's designed false alarm probability, as opposed to when her probability of guessing the key correctly exceeds as in Figure 2. Equation (4) is calculated using P_D from [11], which in this case is $P_D = .9958$ assuming the SNR for Alice to Eve is the same as from Eve to Bob. P_K is obtained from (10). Eve is clearly first limited by α since this probability is the default tolerance for accepting random tags chosen by Alice and Bob. Once she is able to guess the key with high probability, she is simply limited to P_D which is determined by her channel to Bob. The same conclusion of using all key material at once continues to hold for this plot.

C. ℓ-Key Attack

As first mentioned in Section III-A with the idea of guessing entropy, looking only at the probability that the correct key is the most likely is an incomplete analysis of an adversary's capabilities. In practice, Eve has more than one chance to impersonate Alice since she can send multiple messages each with a tag generated from a different key. Such an attack could have the strategy of using the top ℓ most likely keys in successive attacks. This changes the analysis to now look at the probability that the correct key is contained in the set of the ℓ most likely keys which is the basis for list decoding. The ℓ-list decoding performance for the BSC and AWGN regimes are given in Theorems 3 and 4, respectively.

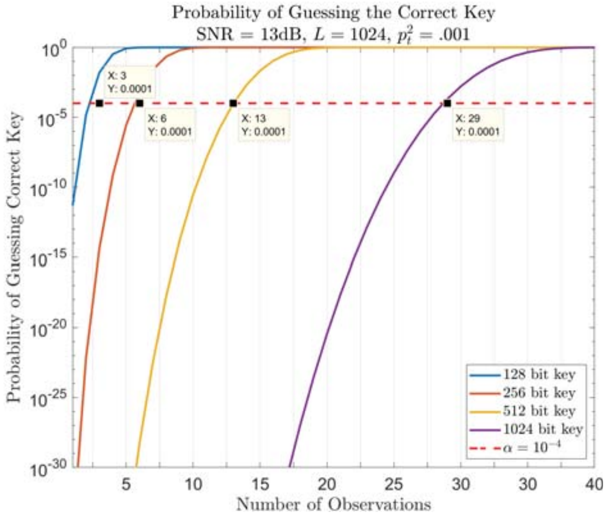

Fig. 2. As Eve obtains more observations, the probability of her guessing the key correctly increases. Adding key material naturally increases the security of the system. The increases show that it is better to use all of the key at once, rather than use some until it is compromised and then using the rest.

Theorem 3: For the BSC, the probability of that the correct key is contained in the top ℓ most likely keys, averaged over all tag mappings, is given by

$$P_{K,\mathrm{BSC}}(\ell) = \sum_{i=0}^{n} \binom{n}{i} p_e^i (1-p_e)^{n-i} \cdot \sum_{q=0}^{\ell-1} \sum_{t=0}^{|\mathcal{K}|-1-q} \min\left\{\frac{\ell-q}{t+1}, 1\right\} \binom{|\mathcal{K}|-1}{q, t, |\mathcal{K}|-1-q-t} \cdot \left(2^{-n} \sum_{k=0}^{i-1} \binom{n}{k}\right)^q \left(2^{-n}\binom{n}{i}\right)^t \left(2^{-n} \sum_{j=i+1}^{n} \binom{n}{j}\right)^{|\mathcal{K}|-1-q-t}. \tag{11}$$

Proof: See [18]. ■

Theorem 4: For the AWGN channel matched filter decoder model, the probability that the correct key is in the top ℓ largest outputs is

$$P_{K,\mathrm{AWGN}}(\ell) = \sum_{i=0}^{\ell-1} \binom{M-1}{i} \int_{-\infty}^{\infty} \phi\left(\frac{t-L}{\sqrt{\frac{L\sigma_e^2}{2p_t^2}}}\right) \Phi\left(\frac{t}{\sqrt{\frac{L}{2} + \frac{L\sigma_e^2}{2p_t^2}}}\right)^{M-1-i} \cdot \left(1 - \Phi\left(\frac{t}{\sqrt{\frac{L}{2} + \frac{L\sigma_e^2}{2p_t^2}}}\right)\right)^i dt. \tag{12}$$

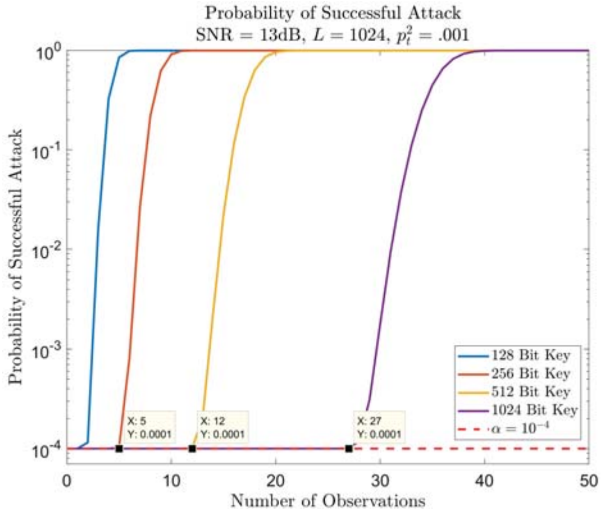

Fig. 3. As Eve obtains more observations, the probability of her successfully impersonating Alice increases. Similarly to Figure 2, it is better to use all of the key at once, rather than use some until it is compromised and then using the rest. The probability of a successful attack stays at the designed α until P_K approaches 1.

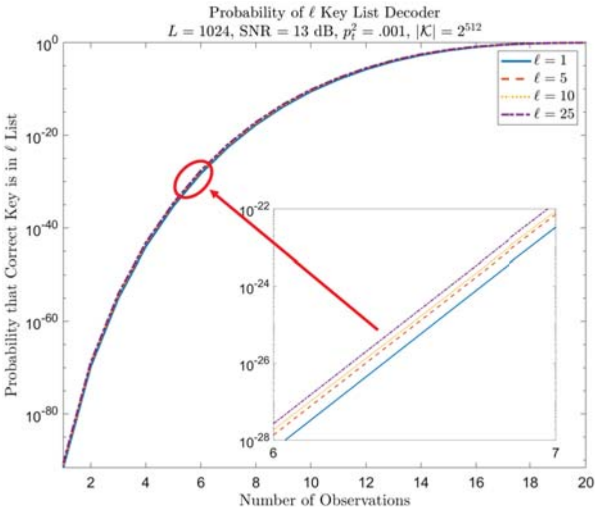

Fig. 4. Increasing the list size trivially increases the probability of the set containing the correct key. Alice and Bob must choose a list size ℓ to plan their key replenishment schedule on. This plot is for the single-key case.

Proof: In order to successfully decode, the correct codeword must be contained in the ℓ-list. This occurs when less than ℓ incorrect codewords have test statistics that are larger than the correct one. Given the distributions of the test statistics from [11], we sum the probabilities over all orderings in which the correct codeword is ranked ℓ or higher to get (12). ■

The probability that the correct key is contained in the top ℓ matched filter outputs is shown in Figure 4 with the same parameters as in Figure 2 and a 512 bit key. Naturally, larger list sizes have a higher probability of containing the correct key, but in this case, the performance gain is very minimal. In other words, if the correct key does not produce the largest matched filter output, then it is nearly equally likely to be in any of the remaining rankings. This shows that Eve will not benefit much from an ℓ-key attack, so Alice and Bob do not have to worry about protecting against such attacks.

REFERENCES

[1] C. E. Shannon, “Communication theory of secrecy systems,” *Bell Labs Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.

[2] A. D. Wyner, “The wire-tap channel,” *Bell Labs Technical Journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[3] I. Csiszár and J. Körner, “Broadcast channels with confidential messages,” *IEEE transactions on information theory*, vol. 24, no. 3, pp. 339–348, 1978.

[4] G. J. Simmons, “Authentication theory/coding theory.” in *Advances in Cryptology, Proceedings of CRYPTO ’84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, 1984, pp. 411–431.

[5] U. M. Maurer, “Authentication theory and hypothesis testing,” *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1350–1356, 2000.

[6] L. Lai, H. El Gamal, and H. V. Poor, “Authentication over noisy channels,” *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 906–916, 2009.

[7] J. Perazzone, E. Graves, P. Yu, and R. Blum, “Inner bound for the capacity region of noisy channels with an authentication requirement,” in *2018 IEEE International Symposium on Information Theory (ISIT)*, June 2018, pp. 126–130.

[8] P. L. Yu, B. M. Sadler, G. Verma, and J. S. Baras, “Fingerprinting by design: Embedding and authentication,” in *Digital Fingerprinting*, C. Wang, R. M. Gerdes, Y. Guan, and S. K. Kasera, Eds. Springer, 2016, pp. 69–88.

[9] A. Rényi, “On measures of entropy and information,” HUNGARIAN ACADEMY OF SCIENCES Budapest Hungary, Tech. Rep., 1961.

[10] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC press, 1996.

[11] G. Verma, P. Yu, and B. M. Sadler, “Physical layer authentication via fingerprint embedding using software-defined radios,” *IEEE Access*, vol. 3, pp. 81–88, 2015.

[12] X. Wu, Z. Yang, C. Ling, and X.-G. Xia, “Artificial-noise-aided message authentication codes with information-theoretic security,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1278–1290, 2016.

[13] G. Smith, “On the foundations of quantitative information flow,” in *International Conference on Foundations of Software Science and Computational Structures*. Springer, 2009, pp. 288–302.

[14] B. Espinoza and G. Smith, “Min-entropy as a resource,” *Information and Computation*, vol. 226, pp. 57–75, 2013.

[15] S. Fehr and S. Berens, “On the conditional rényi entropy,” *IEEE Transactions on Information Theory*, vol. 60, no. 11, pp. 6801–6810, 2014.

[16] C. Cachin, “Entropy measures and unconditional security in cryptography,” Ph.D. dissertation, ETH Zurich, 1997.

[17] S. J. MacMullan and O. M. Collins, “A comparison of known codes, random codes, and the best codes,” in *Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on*. IEEE, 1998, p. 217.

[18] P. Elias, “List decoding for noisy channels,” in *IRE WESCON Convention Record*, vol. 2, 1957, pp. 94–104.