Energy obfuscation for compressive encryption and processing

Energy Obfuscation for Compressive Encryption and Processing

Matteo Testa, Tiziano Bianchi and Enrico Magli
Department of Electronics and Telecommunication engineering
Politecnico di Torino, Italy

***Abstract*—Compressed Sensing enables both computationally secure encryption and signal processing in the compressed domain. Even though these characteristics have always been considered in separate fashion, in this paper we propose a novel method that takes into account these features jointly. As a result we obtain provable secrecy guarantees and enable fast signal processing. In more detail, we show that it is possible to perform anomaly detection relying on the measurements information leakage. At the same time, we can prevent attackers trying to obtain confidential data by obfuscating the information leakage. We show the effectiveness of such method through theoretical bounds and numerical experiments.**

I. Introduction

Compressed Sensing (CS) has become increasingly popular in recent years thanks to its ability to perform signal acquisition and compression in a single operation by means of random projections.

The acquisition, which is a key aspect of CS, can be either performed in software or hardware. Software acquisition can be easily modeled as matrix-vector multiplication, where the sensing matrix is a fat matrix, i.e. it has more columns than rows, made of i.i.d. random entries. However, while general, this approach fails to take full advantage of the CS properties since it still requires to first sample the original signal in a conventional way and only later apply CS. On the other hand, hardware (e.g., optical [1]) acquisition is able to fully exploit CS and allows to either reduce the number of required sensing elements or their power consumption [2]–[4]. This latter aspect makes the CS framework an excellent candidate for the class of low-energy devices which form the Internet of Things (IoT). While a rising demand of IoT devices is foreseeable, leading to increased amounts of data and hence a stronger need for compression, it is unclear how to guarantee the data confidentiality. As highlighted in [5], typical low-energy sensors seem to fail to meet the computational requirements needed to perform standard data encryption operations.

Along the same line, oftentimes sensors are required to perform basic signal processing operations on confidential data, e.g. detect anomalies. While this is a desirable characteristic, in typical settings this operation would require to decrypt the ciphertext before performing signal processing operations, thus requiring even higher computational capabilities. Interestingly, both of the aforementioned characteristics can be provided by means of CS. Indeed, not only it allows to perform signal processing operations in the compressed domain at low cost, e.g. [6], [7], but it can also provide secrecy. In fact, if the sensing matrix is assumed to be a *secret* (only known at trusted parties) and it is re-generated at each acquisition, then the ill-posed inverse problem of CS can be cast as the decryption stage of a private key cryptosystem [8], [9].

At this point, it is important to highlight that the secrecy and processing requirements are orthogonal, i.e. as more processing is needed, more secret information has to be to shared with the processing unit. While different design choices are available, we focus on architectures that need fast processing operations, thus relying on the information that is leaked by the measurements themselves.

Different sensing matrix structures are related to different kinds of information leakage and, as such, authors addressed this problem for Gaussian [9], Bernoulli [10] or circulant [11] sensing matrices. In the best case (i.i.d. Gaussian random entries) the leakage is only related to the energy of the original signal [9], namely an attacker can obtain an estimate of the energy of the original plaintext. Assuming a signal can belong to different classes, each of them having a different energy, it is evident that this leakage, while desirable at the processing side, can help an attacker to gain deeper information about the nature of the encrypted signal. Literature contains works that deal with this problem by normalizing the signal energy in order to avoid this leakage and achieve perfect secrecy (in asymptotic sense) [9]. However, in case of HW acquisition, additional computational hardware may be needed to compute the signal energy and to normalize the measurements. Moreover, since the energy is a confidential information which needs to be transmitted to trusted parties in order to guarantee a correct recovery, encryption schemes such as [12] for this additional quantity need to be taken into account.

In this paper we propose a novel method which allows to bypass the shortcomings of the energy normalization method and also enables fast anomaly detection, allowing to jointly consider secrecy and processing in CS. We show that a multiplicative random gain is able to obfuscate the leakage of information through the measurements when using Gaussian sensing matrices and, in the asymptotic case, also that of generic sensing matrices. For the sake of clarity, in the rest of the paper we will refer to this method as *energy obfuscation*. This approach not only does not require to know the original signal energy but it also does not require any additional information to be transmitted to trusted parties. Indeed, trusted

WIFS'2017, December, 4-7, 2017, Rennes, France. 978-1-5090-6769-5/17/$31.00

parties which can efficiently de-obfuscate the measurements can perform fast anomaly detection before performing any recovery operations. The result is an efficient obfuscation scheme that can be efficiently implemented in a compressive cryptosystem architecture at very low cost.

II. Compressive cryptosystem architecture

Before discussing the architecture of the compressive cryptosystem, it is important to state the security model we will adopt in the remainder of this paper. We assume that the attacker not only has access to ciphertext, but also to arbitrary plaintext-ciphertext couples. Moreover, the ciphertext has to be protected against any attacker trying to either decrypt the signal or to estimate the original signal's energy. Driven by this security model, we consider the one time sensing scheme which requires to re-generate the sensing matrix at each acquisition in order to make the cryptosystem resistant to known and chosen plaintext attacks. By employing this strategy, as depicted in Fig. 1, the secret is a key which is shared among trusted parties and is used to generate the full sensing matrix by means of a generating function Gen_k. At encryption side the measurements y are acquired either via hardware or software and sent to the post-acquisition processing unit. This block will output the actual ciphertext z by applying, if needed, additional processing. In particular, for the specific case we are considering throughout this paper this block will handle the energy obfuscation. At the decryption side, the sensing matrix is generated from the same shared key k and used along with the ciphertext z by the decryption block Dec_Φ to produce the recovered plaintext $\hat{x}$. This is done by inverting the post-processing operation and using any available CS recovery algorithm, e.g. LASSO.

In the architecture we also include a processing block which can perform basic signal processing operations in the encrypted domain based on partial or no knowledge of the sensing matrix entries. This block will be mainly considered in Sec. IV, where an energy based anomaly detector is used to show the performance of the proposed method. Is it worth noting that the largest computational burden a sensor will have to handle is the cost of the sensing matrix regeneration. One may argue that at the same cost a block cipher scheme could be implemented on the same sensor. However, the advantage of the proposed scheme is a fast processing in the encrypted domain. As described in Sec. IV, the processing block only needs the ciphertext and the first sample of the sensing matrix to perform signal processing, e.g. anomaly detection and potentially raise alarms. Conversely, a block cipher-based encryption scheme would require the processing block to generate the full key stream to decrypt the ciphertext before performing any processing; this would also require to disclose the whole plaintext to the entity performing the processing stage.

Lastly, we recall that in this paper we mainly focus on the post-acquisition processing of the measurements in order to increase the security of the whole cryptosystem.

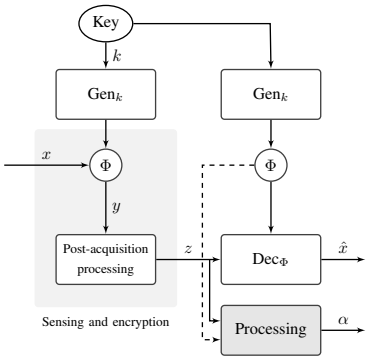

Fig. 1: Compressive cryptosystem architecture scheme

III. Main results

In this section we describe how to obfuscate the energy leakage of a Gaussian CS cryptosystem through multiplicative blinding. Let us start with the encryption model we consider throughout this paper which is given by

$$z = ay = a\Phi x, \tag{1}$$

where $z \in \mathbb{R}^{m\times 1}$ is the encrypted signal, $x \in \mathbb{R}^{n\times 1}$ is the original signal, the entries of $\mathbf{\Phi} \in \mathbb{R}^{m\times n}$ are i.i.d. and follow $\phi_{i,j} \sim \mathcal{N}(0, \sigma_\Phi^2)$ and $a \in \mathbb{R}$ is drawn from a log-Normal distribution $a \sim \ln\mathcal{N}(0, \sigma_a^2)$. It is known from [9] that $\varepsilon_x = \mathbf{x}^\top\mathbf{x}$, i.e. the energy of the original signal, is leaked by the measurements y.

In order to reduce the leakage, quantified by the mutual information $I(x; y)$, we propose to scale the energy according to a random value. It is important to highlight that we consider a single scalar multiplication instead of a vector-wise product with a random vector. The reason behind this choice comes from the fact that it is known from [8] that, if the sensing matrix is Gaussian and unknown, the spherical angle of the original vector cannot be determined (whereas its magnitude can indeed be estimated). Thus, a scalar multiplication which only modifies the magnitude of the signal is preferred, differently from other transformations which can also affect its spherical angle.

In more detail, we quantify the improved secrecy due to the random scalar multiplication in terms of lower mutual information and increased η mean square error (MSE). This latter metric, as defined in [9], quantifies the normalized minimum MSE that can be obtained by an estimator seeking an estimate of ε_x from the encrypted signal z, and can be defined as:

Definition III.1. *The measurements are said to be η-MSE secret with respect to the signal's energy if for every possible estimator $\hat{\varepsilon}_x(z)$ of ε_x, we have that*

$$\eta_{\hat{\varepsilon}_x} \triangleq \frac{\mathbb{E}[\|\varepsilon_x - \hat{\varepsilon}_x(z)\|_2^2]}{\sigma_{\varepsilon_x}^2} \geq \eta,$$

where $\sigma_{\varepsilon_x}^2$ is the variance of ε_x.

Let us start with an equivalence which we will use in the remainder of the paper.

Lemma III.2. *Assuming the model considered in* (1)*, the mutual information between x and z is equivalent to the mutual information between their energies as $I(z;x) = I(\varepsilon_z;\varepsilon_x)$.*

Proof. The proof is presented in the Appendix. □

The result of the above Lemma allows us to consider $I(\varepsilon_z;\varepsilon_x)$ instead of $I(z;x)$ which makes the problem easier to tackle. In the next lemma we present an upper bound for $I(\varepsilon_z;\varepsilon_x)$.

Lemma III.3. *If we consider a CS cryptosystem as defined in* (1) *and $p(\varepsilon_x = 0) = 0$ and $p(a = 0) = 0$, then the leakage of information of x through z is bounded by*

$$I(z;x) = I(\varepsilon_z;\varepsilon_x) \leq \frac{1}{2}\ln\left(1 + \frac{\psi_1\left(\frac{m}{2}\right) + \operatorname{var}(\ln\varepsilon_x)}{4\sigma_a^2}\right),$$

where $\psi_1(z) = \frac{d^2}{dz^2}\ln\Gamma(z)$ is the trigamma function.

Proof. The proof is presented in the Appendix. □

From the above lemma we can see the fundamental role of the term σ_a^2; as its value increases, the upper-bound and hence $I(z;x)$ goes towards zero. It important to note that, as shown in the next section, small values of σ_a^2 are able to significantly increase the secrecy of the system. Suitable choices for σ_a are discussed in Sec. IV. In order to obtain the value of the η-MSE metric when energy obfuscation is employed, we state the following

Lemma III.4. *Obfuscated measurements are at least η-MSE secret with respect to ε_x, where*

$$\eta = \frac{e^{h(\varepsilon_x|z)-1}}{2\pi\sigma_{\varepsilon_x}^2}.$$

Proof. As for Lemma 2 in [9], by employing Theorem 8.6.6 in [13] we have that $\mathbb{E}[\|\varepsilon_x - \hat{\varepsilon}_x(z)\|_2^2] \geq \frac{1}{2\pi}e^{2h(\varepsilon_x)-2I(\varepsilon_x;z)-1} = \frac{1}{2\pi}e^{2h(\varepsilon_x|z)-1}$, the result then follows from the definition of η-MSE secrecy. □

Next, employing Lemma III.3 and Lemma III.4 we can state the following

Corollary III.4.1. *If we consider a CS cryptosystem as defined in* (1)*, and x is an exactly k-sparse signal with i.i.d Gaussian non-zero entries, the minimum MSE obtainable by any estimator seeking an estimate of ε_x from the encrypted signal z is given by*

$$\eta = \frac{e^{2\xi\left(\frac{k}{2}\right) - \ln\left(1 + \frac{\psi_1\left(\frac{m}{2}\right) + \psi_1\left(\frac{k}{2}\right)}{4\sigma_a^2}\right) - 1}}{\pi k},$$

where $\xi(z) = z + \ln\Gamma(z) + (1-z)\psi(z)$ and $\psi(z)$ is the digamma function,

where we used the fact that for k-sparse signals with i.i.d. Gaussian non-zero entries $\operatorname{var}(\ln\varepsilon_x) = \psi_1\left(\frac{k}{2}\right)$. We now extend the previous result for generic sensing matrices and finite power signals. If we consider the asymptotic setting, that is $n \to \infty$, we can state the following

Proposition III.5. *If we consider X to be a random process whose realizations x_j have finite power $W_x = \lim_{n\to\infty}\frac{1}{n}\sum_{j=0}^{n-1}x_j^2$ and are mapped to Y_j according to* (1) *to a finite m where the entries of Φ are i.i.d. from a subgaussian distribution, then as $n \to \infty$ Lemma III.3 and Corollary III.4.1 hold for generic sensing matrices with strict less-than sign.*

Proof. The proof is presented in the appendix. □

It is worth to highlight that this important result can be used to guarantee the secrecy of efficient sensing matrices such as Bernoulli ones for sufficiently large values of n. Moreover, since practical Gaussian sensing matrix entries are drawn from truncated distributions and represented with finite precision, this result can also be used to provide secrecy for this class of sensing matrices.

A. Numerical simulations

We now present some experiments showing the behavior of the bounds discussed in the previous section to gain a better understanding. In Fig. 2 the values of the bound on $I(z;x)$ as a function of the number of measurements m are depicted. We compare it with the bound obtained in [9] to validate the effects of the energy obfuscation. Since for this experiment no distribution for ε_x is specified and it is known that $c_0 = \log\mathbb{E}[\varepsilon_x] - \mathbb{E}[\log\varepsilon_x] \geq 0$, we fix its value to a positive constant $c_0 = 0.1$. For the same reason we fix $c_1 = \operatorname{var}(\ln\varepsilon_x) = 0.2$. As can be seen, the proposed method is able to decrease the mutual information $I(\varepsilon_z,\varepsilon_x)$; this means that the energy leakage, as seen by an attacker, is greatly reduced. Conversely, if no energy obfuscation is applied the mutual information is higher and increases with the number of measurements. It is important to highlight that the bound in Fig. 2 behaves in a counter-intuitive way. The mutual information decreases as the number of measurements becomes larger, in contrast to what one may expect. This may be explained since the depicted mutual information is actually an upper bound value whereas as m increases the upper bound becomes tighter. A similar behavior is shown in Fig. 3 where the minimum obtainable MSE on ε_x is considered for signals which are exactly k-sparse with i.i.d. non-zero Gaussian entries. Moreover, in this experiment different values of m are considered, but still keeping fixed the ratio $k/m = 0.5$. In this experiment we also consider the theoretical performance of a linear

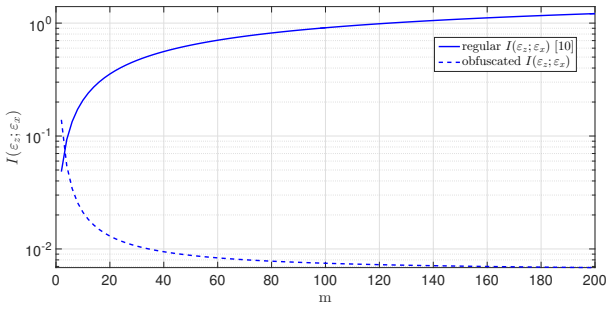

Fig. 2: Mutual information, $\sigma_a = 2$

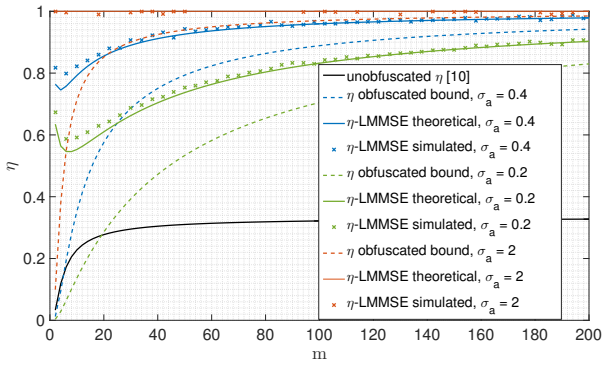

Fig. 3: η-MSE information for different σ_a in function of m

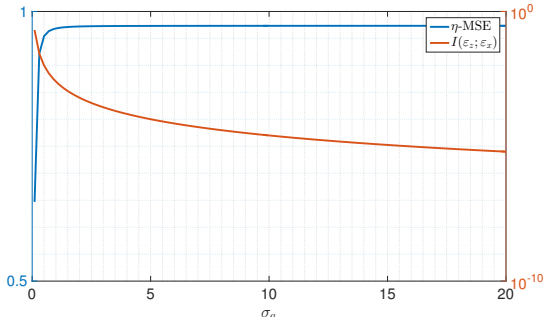

Fig. 4: Behavior of I and η as a function of the variance σ_a

minimum mean square error (LMMSE) estimator which can be obtained as $\hat{\varepsilon}_x(\varepsilon_z)_{LMMSE} = \frac{2\varepsilon_z}{\sigma_\Phi^2 h} + \frac{k\sigma_x^2(h-2me^{2\sigma_a^2})}{h}$, where $h = 2(2+k+m)(g+e^{2\sigma_a^2}) + mkg$ and $g = (e^{4\sigma_a^2}-1)e^{4\sigma_a^2}$. Then, the MSE achieved by this estimator can be shown to be $\eta_{LMMSE} = 1 - \frac{2me^{4\sigma_a^2}}{h}$. It can be seen in Fig. 3 that the MSE of the LMMSE estimator approaches the bound as m increases, however there is still a gap between these two quantities. This is due to the fact that during the derivation of the η-MSE in Corollary III.4.1 the entropy of $\log \varepsilon_z$ is upper bounded by that of a Gaussian distribution. For the sake of completeness, Fig. 3 also depicts the simulated η-LMMSE, obtained averaging over 10^5 experiments, that is very close to the theoretical η-LMMSE curve.

In a last experiment, depicted in Fig. 4, we show the behavior of both mutual information and η with respect to different values of σ_a: increasing the variance of $\log(a)$ reduces the mutual information and increases the η-MSE.

IV. Application to Anomaly Detection

In this section we show an experiment aimed at showing the effectiveness of the energy obfuscation method in a practical setting. We consider an energy based anomaly detector which can be of interest in applications such as infrared camera fire detectors, where the acquisition is performed by means of CS. The goal here is that of being able to perform a fast detection (avoiding the full signal decryption chain) by exploiting the information leakage, if an anomaly is detected then the original image should be recovered. Nevertheless, we need to provide

(a) (b)

Fig. 5: Set of two 64×64 images employed in the experiment: (a) regular image (b) anomaly image.

confidentiality by avoiding an attacker who has access to the ciphertext to correctly detect the anomaly. Let us define the problem as a threshold-based detection where $\mathcal{H}_0$ corresponds to no anomaly and $\mathcal{H}_1$ means that an anomaly has occurred. Namely if $\varepsilon_x > \tau$ the detector will consider $\mathcal{H}_1$ to be true, and $\mathcal{H}_0$ in the other case. Both legitimate and attacker detectors have no access to the true ε_x but rather they can estimate its value given ε_y and ε_z for the legitimate and attacker detectors respectively.

If we consider the best estimator in terms of MSE, which is the minimum MSE estimator, it has to be highlighted that it requires to specify a prior distribution on the plaintext energy ε_x. However, considering the problem we described in this section, no informative prior can be reasonably chosen as the best one. Thus, we suppose that both attacker and legitimate detectors employs a maximum-likelihood (ML) estimation strategy, which can be shown to asymptotically achieve the minimum MSE among all consistent estimators. For what concerns the attacker side, since a ML estimator requires the knowledge of the conditional probability $p(\varepsilon_z|\varepsilon_x)$ which cannot be obtained in closed form for this specific case, we propose to use a two step ML estimation. At first the ML estimator in (2) is employed to estimate the value of ε_y.

$$\hat{\varepsilon}_y(\varepsilon_z) = \arg\max_{\varepsilon_y} \frac{-\log^2 \varepsilon_y + \log \varepsilon_z \log \varepsilon_y}{8\sigma_a^2}. \tag{2}$$

This estimate is then considered as a given observation and used by the estimator in (3) which outputs an estimate of ε_x.

$$\hat{\varepsilon}_x(\varepsilon_y) = \frac{\varepsilon_y}{m\sigma_\Phi^2}. \tag{3}$$

It is important to note that, if m is large enough, the variance of $p(\varepsilon_y|\varepsilon_x)$ is small and thus, the knowledge of ε_y implies the knowledge of ε_x with no or little uncertainty. Under these circumstances, the assumption we made to justify the two-step ML estimation strategy is legitimate. Based on this assumption the attacker uses the two-step ML estimator previously described in order to estimate $\hat{\varepsilon}_x$ and perform the anomaly detection. Conversely, the legitimate detector uses the secret key to generate the first random sample in order to obtain the obfuscation random scalar a. At this point it can obtain $y = z/a$ and use (3) to estimate $\hat{\varepsilon}_x$. These estimates are then used to discriminate between $\mathcal{H}_0$ and $\mathcal{H}_1$ based on the result of the thresholding.

For this anomaly detection experiment, we employ two synthetic images (shown in Fig. 5), namely regular image and image with anomaly. In Fig. 6 we show the receiver operating characteristic (ROC) curve for both the attacker and the legitimate detectors trying to detect the anomaly. We consider

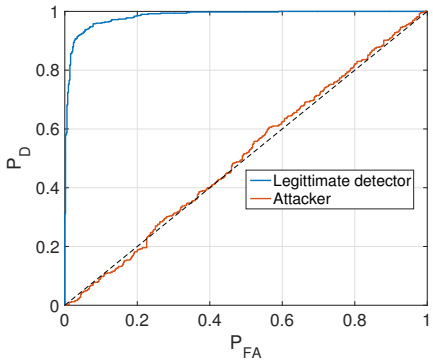

Fig. 6: Probability of detection in function of the probability of false alarm for both the attacker and the legitimate detector.

1000 experiments with a fixed compression ratio $m/n = 0.3$ with $n = 4096$, Gaussian i.i.d entries with $\sigma_\Phi^2 = 1$ for the sensing matrix and $\sigma_a^2 = 2$. Additionally we add AWGN noise to the acquired images with standard deviation 0.95. As can be seen, the proposed method obfuscates the energy to the attacker making the energy thresholding detection more difficult as it flattens the ROC curve towards the perfect secrecy which corresponds to $P_D = P_{FA}$. On the other hand, the legitimate detector can detect the anomaly with very high probability.

V. Practical implementation strategies

In this section we briefly discuss a possible implementation of the sensing matrix generation block. If we consider the sensing matrix having i.i.d. entries distributed according to $\mathcal{N}(0,1)$ we need to securely generate mn Gaussian entries. Different approaches to solve this problem include the Box-Muller transform [14] and the Ziggurat method [15]. Because of its simplicity and efficient implementation we consider the Box-Muller transform method which generates two Gaussian distributed entries given two samples uniformly distributed in $(0,1]$. To generate these uniform samples we can rely on trusted cryptographic primitives which generate uniformly distributed random bits. In particular, as suggested in [16] we choose to employ the SHA3 *Skein* [17] or *Keccak* [18] algorithms which include a Key Derivation Mode which can generate an arbitrary number of uniformly distributed bits with a single call to the pseudorandom generation function (PRF). For their representation we use a Qb representation which is a fixed point representation which employs b-bits for the significand. If we then choose the b-bits to be the output of a secure hashing algorithm, the result is a uniformly distributed number in $[0,1)$, from which we map the 0 value to 1 in order to limit the interval to $(0,1]$. We now recall that the value of b directly influences the smallest number which can be represented and hence the maximum output value of the Box-Muller transformation. As example, if we consider $b = 32$ which leads to the smallest representable number to be 2^{-32}, the Box-Muller transformation will always output values smaller than 6.6604. Consequently, the Gaussian values are sampled, in practice, from a Gaussian distribution with truncated tails for probabilities smaller than 1.3654×2^{-36}. In particular, by employing $b = 78$ the tails are truncated for probabilities smaller than 2^{-83} which is considered a negligible value for cryptographic applications.

Nevertheless, here we need to take into account that compressive cryptosystem schemes are suited for low-power weak-secrecy applications. This means that for this range of applications, we are not interested in reaching perfect secrecy. Consequently, the number of required bits can be reduced depending on the specific application requirements in terms of secrecy and computational capabilities.

Lastly, the generation of the energy obfuscation parameter relies on the same strategy as described above, except for the fact that the Gaussian sample has to be scaled and exponentiated in order to be distributed accordingly to a log-Normal distribution with the desired variance. Regarding this latter aspect, it is important to note that $\mathrm{var}(a) = (e^{\sigma_a^2} - 1)e^{2\mu_a + \sigma_a^2}$ and the secrecy bounds we obtained depend on the value of σ_a^2. This suggests that one can have arbitrarily large σ_a^2 by keeping bounded the value of $\mathrm{var}(a)$, e.g. by setting $\mu_a = -\sigma_a^2$ which results in $\mathrm{var}(a) = 1 - e^{-\sigma_a^2}$. However, this trick does not take into account the fact that with finite precision we have a limit on how large σ_a^2 could be. More in detail, if we consider a floating point number representation, we have that the smallest number which can be represented is approximately 1.2×10^{-38} for single precision and 5×10^{-324} for double precision respectively. At this point we obtain the maximum σ_a which can be used in practice by recalling that $a = e^{\mu_a + \sigma_a X}$ where $X \sim \mathcal{N}(0, \sigma_x^2)$. If we consider that $\sigma_x^2 = 1$ and that the distribution of x is truncated at $10\sigma_x$, we obtain that $\sigma_{a,\mathrm{MAX}}^{\mathrm{single}} = 15.6$ and $\sigma_{a,\mathrm{MAX}}^{\mathrm{double}} = 32.74$ for single and double precision respectively. To conclude, even tough this trick has some limitations in practical applications and σ_a cannot be chosen to be arbitrarily large, it is important to consider that small values of σ_a are sufficient to provide secrecy guarantees.

VI. Conclusions

In this paper we considered the existing trade-off between the processing capabilities and the secrecy of a compressive cryptosystem. In fact, we showed that it is possible to design a compressive cryptosystem which offers a good balance between these two characteristics. More in detail, relying on the measurements information leakage, the proposed method enables fast anomaly detection and increased secrecy through a simple random multiplication. Its effectiveness is proven from both theoretical and experimental points of view. To conclude, it is worth noting that, while the literature of CS encryption schemes is increasing, there is still a open gap towards practical implementations, and this paper presents a few possible solutions.

Appendix

Proof of Lemma III.2. At first we show that $I(z;x) = I(z;\varepsilon_x)$. From [9] we have that $p(y|x) = p(y|\varepsilon_x)$, thus

$p(z|x) = \int p(z|y)p(y|x)dy = \int p(z|y)p(y|\varepsilon_x)dy = p(z|\varepsilon_x)$. Then, we can write $I(z;x) = I(z;x,\varepsilon_x) = I(z;\varepsilon_x) + I(x;z|\varepsilon_x) = I(z;\varepsilon_x)$. This is due to the fact that since $p(z|x) = p(z|\varepsilon_x)$, we have that $I(x;z|\varepsilon_x) = I(x;z|x) = h(x|x) - h(x|z,x) = 0$. Lastly, let us define $u_z = z/\varepsilon_z$ which allows us to write $I(\varepsilon_x;z) = I(\varepsilon_x;\varepsilon_z,u_z) = I(\varepsilon_x;\varepsilon_z) + I(\varepsilon_x;u_z|\varepsilon_x) = I(\varepsilon_z;\varepsilon_x)$ since u_z is uniformly distributed on a hypersphere and it is independent on ε_z and ε_x. □

Proof of Lemma III.3. To prove this lemma, in a similar fashion to [19], we start the derivation obtaining a bound on $I(\varepsilon_z;\varepsilon_y)$ which, under the assumptions of the Lemma and since the logarithm is a deterministic and invertible function, it is equal to $I(\bar{\varepsilon}_z;\bar{\varepsilon}_y)$ with $\bar{\varepsilon}_z = \ln\varepsilon_z$ and $\bar{\varepsilon}_y = \ln\varepsilon_y$. For this reason we carry on the derivation starting from the equation $\bar{\varepsilon}_z = 2\bar{a} + \bar{\varepsilon}_y$ where $\bar{a} = \ln a$. By definition we have

$$I(\bar{\varepsilon}_z;\bar{\varepsilon}_y) = h(\bar{\varepsilon}_z) - h(\bar{\varepsilon}_z|\bar{\varepsilon}_y) = h(\bar{\varepsilon}_z) - h(2\bar{a}) \qquad (4)$$

Since the variance of a normal distributed r.v. with variance σ^2 is given by $\frac{1}{2}\log(2\pi\sigma^2)$, focusing on the second term, we have that $h(2\bar{a}) = \frac{1}{2}\ln\left(8\pi e\sigma_a^2\right)$. For what concerns the first terms, since it is difficult to obtain its entropy in closed form, we will bound it using its maximum. We choose to bound it with the entropy of a Gaussian distribution having the same variance as $\bar{\varepsilon}_z$ since they share the same support $(-\infty,+\infty)$. Since $p(\varepsilon_y|\varepsilon_x) = \mathrm{Gamma}\left(\frac{m}{2}, 2\sigma_\phi^2 m\varepsilon_x\right)$, by the law of total variance, we have that $\mathrm{var}(\ln\bar{\varepsilon}_y) = \mathop{\mathbb{E}}_{p(\varepsilon_x)}\left[\mathop{\mathrm{var}}_{p(\varepsilon_y|\varepsilon_x)}(\bar{\varepsilon}_y)\right] + \mathrm{var}\left(\mathop{\mathbb{E}}_{p(\varepsilon_y|\varepsilon_x)}[\bar{\varepsilon}_y]\right) = \psi_1\left(\frac{m}{2}\right) + \mathrm{var}\left(\psi\left(\frac{m}{2}\right) + \ln(2\sigma_\Phi^2 m\varepsilon_x)\right) = \psi_1\left(\frac{m}{2}\right) + \mathrm{var}(\ln\varepsilon_x)$. Thus, the variance of $\bar{\varepsilon}_z$ is given by $\mathrm{var}(\bar{\varepsilon}_z) = 4\sigma_a^2 + \psi_1\left(\frac{m}{2}\right) + \mathrm{var}(\ln\varepsilon_x)$. Hence we can bound $h(\bar{\varepsilon}_z)$ with the entropy of a gaussian distritbution having the same variance as $h(\bar{\varepsilon}_z) \leq \frac{1}{2}\ln\left(2\pi e\left[4\sigma_a^2 + \psi_1\left(\frac{m}{2}\right) + \mathrm{var}(\ln\varepsilon_x)\right]\right)$. If we now put together the two terms in (4), we can obtain $I(\varepsilon_z;\varepsilon_y) = I(\bar{\varepsilon}_z;\bar{\varepsilon}_y) \leq \frac{1}{2}\ln\left(1 + \frac{\psi_1\left(\frac{m}{2}\right) + \mathrm{var}(\ln\varepsilon_x)}{4\sigma_a^2}\right)$ In order to link the $I(\varepsilon_z;\varepsilon_y)$ with $I(\varepsilon_z;\varepsilon_x)$ we can note that the energies follow a Markov chain $\varepsilon_z \to \varepsilon_y \to \varepsilon_x$. Thus, if we apply the data processing inequality, we get $I(\varepsilon_z;\varepsilon_x) \leq I(\varepsilon_z;\varepsilon_y)$. □

Proof of Proposition III.5. Here we give an intuitive explanation of the validity of the bounds in the asymptotic case. From Proposition 2 in [10], we have that $p(y|x) \sim \mathcal{N}\left(0, \sigma_\Phi^2\varepsilon_x\mathbb{I}_m\right)$ as $n \to \infty$. Moreover, we have that the bound in Lemma III.3 will only hold with the equal sign *iff* ε_z is distributed as a log-Normal random variable. However, under the assumption of a being distributed as a log-Normal, $\varepsilon_z = a^2\varepsilon_y$ can not be log-Normal since such distribution can only be obtained as the product of log-Normal distributed RVs which would imply ε_y is log-Normal, which is not the case. This means that the bound holds for strict less-than sign. Given these considerations, and recalling that in the asymptotic sense $p(y|x)$ tends to a Normal distribution, there must be an n starting from which the asymptotic mutual information approaching the true one is smaller than the value of the upper bound. This verifies the proposition. □

ACKNOWLEDGMENT

This work results from the research cooperation with the Sony Technology Center Stuttgart (Sony EuTEC). We would especially like to thank Lev Markhasin and Oliver Erdler from Sony Technology Center Stuttgart for their feedback to this work and all the fruitful discussions.

REFERENCES

[1] M. F. Duarte, M. A. Davenport, D. Takbar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[2] M. A. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE transactions on signal processing*, vol. 57, no. 6, pp. 2275–2284, 2009.

[3] C. Quinsac, A. Basarab, J.-M. Girault, and D. Kouamé, "Compressed sensing of ultrasound images: Sampling of spatial and frequency domains," in *Signal Processing Systems (SIPS), 2010 IEEE Workshop on*. IEEE, 2010, pp. 231–236.

[4] D. Gangopadhyay, E. G. Allstot, A. M. Dixon, K. Natarajan, S. Gupta, and D. J. Allstot, "Compressed sensing analog front-end for bio-sensor applications," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 2, pp. 426–438, 2014.

[5] W. Trappe, R. Howard, and R. S. Moore, "Low-energy security: Limits and opportunities in the internet of things," *IEEE Security & Privacy*, vol. 13, no. 1, pp. 14–21, 2015.

[6] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, "Signal processing with compressive measurements," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 445–460, 2010.

[7] M. Testa and E. Magli, "Compressive estimation and imaging based on autoregressive models," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5077–5087, 2016.

[8] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*. IEEE, 2008, pp. 813–817.

[9] T. Bianchi, V. Bioglio, and E. Magli, "Analysis of one-time random projections for privacy preserving compressed sensing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 313–327, 2016.

[10] V. Cambareri, M. Mangia, F. Pareschi, R. Rovatti, and G. Setti, "Low-complexity multiclass encryption by compressed sensing," *IEEE Transactions on Signal Processing*, vol. 63, no. 9, pp. 2183–2195, 2015.

[11] T. Bianchi and E. Magli, "Analysis of the security of compressed sensing with circulant matrices," in *Information Forensics and Security (WIFS), 2014 IEEE International Workshop on*. IEEE, 2014, pp. 173–178.

[12] R. Fay, "Introducing the counter mode of operation to compressed sensing based encryption," *Information Processing Letters*, vol. 116, no. 4, pp. 279–283, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.ipl.2015.11.010

[13] T. M. Cover and J. A. Thomas, "Elements of information theory 2nd edition," 2006.

[14] D. W. Scott, "Box–muller transformation," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 3, no. 2, pp. 177–179, 2011.

[15] G. Marsaglia, W. W. Tsang *et al.*, "The ziggurat method for generating random variables," *Journal of statistical software*, vol. 5, no. 8, pp. 1–7, 2000.

[16] R. Fay and C. Ruland, "Compressive sensing encryption modes and their security," 2016.

[17] N. Ferguson, S. Lucks, B. Schneier, D. Whiting, M. Bellare, T. Kohno, J. Callas, and J. Walker, "The skein hash function family," *Submission to NIST (round 3)*, vol. 7, no. 7.5, p. 3, 2010.

[18] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "The keccak sha-3 submission," *Submission to NIST (Round 3)*, vol. 6, no. 7, p. 16, 2011.

[19] T. Bianchi, A. Piva, and M. Barni, "Analysis of the security of linear blinding techniques from an information theoretical point of view," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 5852–5855.