

A Lower Bound on the Key Length of Information-Theoretic Forward-Secure Storage Schemes

Stefan Dziembowski*

Department of Computer Science
University of Rome, *La Sapienza*

Abstract. *Forward-Secure Storage* (FSS) was introduced by Dziembowski (CRYPTO 2006). Informally, FSS is an encryption scheme (Encr, Decr) that has the following non-standard property: even if the adversary learns the value of some function h of the ciphertext $C = \text{Encr}(K, M)$, he should have essentially no information on the corresponding plaintext M , even if he knows the key K . The only restriction is that h is *input-shrinking*, i.e. $|h(R)| \leq \sigma$, where σ is some parameter such that $\sigma \leq |C|$.

We study the problem of minimizing the length of the secret key in the IT-secure FSS, and we establish an almost optimal lower bound on the length of the secret key. The secret key of the FSS scheme of Dziembowski has length $|M| + O(\log \sigma)$. We show that in every FSS the secret key needs to have length at least $|M| + \log_2 \sigma - O(\log_2 \log_2 \sigma)$.

1 Introduction

Forward-Secure Storage (FSS) was introduced by Dziembowski in [5]. Informally, FSS is an encryption scheme (Encr, Decr) that has the following non-standard property: if the adversary has only partial information about the ciphertext $C = \text{Encr}(K, M)$, he should have essentially no information on the corresponding plaintext M , even if he learns the key K . Here, “partial information” means that the adversary knows some value $U = h(C)$, where h is chosen by him. The only restriction is that h is *input-shrinking*, i.e. $|U| \leq \sigma$, where σ is some parameter such that $\sigma \leq |C|$. In the security definition one assumes that h has to be chosen *before* the adversary learns K (as otherwise he could simply choose h to be the function that decrypts M from C). Since usually one wants to construct schemes that are secure for large values of σ , and since obviously $\sigma < |C|$, therefore normally $\text{Encr}(K, M)$ is much longer than M .

Originally FSS was proposed in the context of the so-called *Bounded-Storage Model (BSM)*¹ [4,3,5,7,10,2] as a tool for increasing security of data stored on

* The European Research Council has provided financial support under the European Community’s Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement no 207908.

¹ In [5] this model was called a *Limited Communication Model*.

the machines that can be attacked by internet viruses. In this model one assumes that the ciphertext C is stored on a PC on which the adversary can install a virus. The virus may perform any computation on C but he can communicate to the adversary only a value $|h(C)| \leq \sigma$. The practical relevance of this assumption comes from the fact that in many cases it may be hard to retrieve large amounts of data from an infected machine. Since in practice the length of C needs to be huge (several gigabytes) it is often required that it should be possible to decrypt M just by reading a small number of the bits of C .

Another application of FSS is to use it for storing data on hardware that can leak information via the so-called side-channel attacks, which are the attacks based on measuring the power consumption, electromagnetic radiation, timing information, etc. As before, one can model such an attack by allowing the adversary to compute some input-shrinking function on ciphertext (this method was also used, in a different context in [8,1,11]). The only difference is that usually the size of the secret data stored on the device is much smaller, and hence there is no need to require that only a small portion of C has to be read to decrypt the message.

In this paper we study the problem of constructing FSS schemes that are information-theoretically (IT) secure, which means that the computing power of the adversary is not limited, and there is no restriction on the computational complexity of the function h . Such an IT-secure FSS scheme was already constructed in [5] (besides of this, [5] considers also computationally-secure and so-called hybrid-secure schemes).

Our contribution: In this paper we revisit the IT-secure FSS construction of [5], and establish an almost optimal lower bound on the length of the secret key. The secret key of the FSS scheme of [5] has length $|M| + O(\log \sigma)$ (if built using an appropriate randomness extractor). Obviously, since FSS has to be secure as an information-theoretically encryption scheme, by Shannon's theorem the length of the key has to be at least $|M|$, one may ask, however, if the $O(\log \sigma)$ term is necessary. In this paper we show that that the construction of [5] is essentially optimal, by proving (cf. Corollary 1) that in every secure FSS the secret key needs to have length at least $|M| + \log_2 \sigma - O(\log_2 \log_2 \sigma)$.

2 FSS — The Formal Definition

Formally, a *Forward-Secure Storage* (FSS) scheme is a pair of randomized algorithms $\Phi = (\text{Encr}, \text{Decr})$. The algorithm Encr takes as input a *key* $K \in \mathcal{K}$ and a *plaintext* $M \in \mathcal{M}$ and outputs a *ciphertext* $C \in \mathcal{C}$. The algorithm Decr takes as input a key K and a ciphertext C , and it outputs a string M' . The following correctness property has to be satisfied with probability 1: $\text{Decr}(K, \text{Encr}(K, M)) = M$.

To define the security of an FSS scheme consider a σ -*adversary* \mathcal{A} (that we model as a Turing Machine), that plays the following game against an oracle Ω .

FSS - distinguishing game

1. The adversary produces two messages $M^0, M^1 \in \{0, 1\}^\mu$ and sends them to Ω .
2. Ω selects a random key $K \in \{0, 1\}^\kappa$, a random bit $b \in \{0, 1\}$ and computes $C = \text{Encr}(K, M^b)$.
3. The adversary gets access to C and can compute an arbitrary value $U = h(C)$ such that $|U| \leq \sigma$. The adversary can store U , but he is not allowed to store any other information.
4. The adversary learns K and has to guess b .

We say that an adversary \mathcal{A} *breaks the scheme Φ with an advantage ϵ* if his probability of winning the game is $1/2 + \epsilon$. We say that an FSS scheme Φ is (ϵ, σ) -*IT-secure* if every σ -adversary \mathcal{A} breaks Φ with advantage at most ϵ . Without loss of generality we can assume that \mathcal{A} is deterministic. This is because a computationally-unlimited deterministic adversary can always compute the optimal randomness for the randomized adversary.²

3 FSS — The Construction of [5]

3.1 Probability-Theoretic Preliminaries

Let random variables X_0, X_1, X_2 be distributed over some set \mathcal{X} and let Y be a random variable distributed over \mathcal{Y} . Define the *statistical distance between X_0 and X_1* as $\delta(X_0; X_1) = \frac{1}{2} \sum_{x \in \mathcal{X}} |P(X_0 = x) - P(X_1 = x)|$. If X is distributed over \mathcal{X} then let $d(X) := \delta(X; U_{\mathcal{X}})$ denote the *statistical distance of X from a uniform distribution (over \mathcal{X})*. Moreover, $d(X_0|X_1) = \delta((X_0, X_1); (U_{\mathcal{X}}, X_1))$ denotes the statistical distance of X_0 from a uniform distribution *given X_1* . It is easy to verify that

$$d(X_0|X_1) = \sum_x d(X_0|X_1 = x) \cdot P(X_1 = x), \quad (1)$$

and that the triangle inequality ($\delta(X_0, X_1) \leq \delta(X_0, X_2) + \delta(X_2, X_1)$) holds. We will overload the symbols δ and d and sometimes apply them to the probability distributions instead of the random variables. A *min-entropy \mathbf{H}_∞ of a random variable R* is defined as

$$\mathbf{H}_\infty(R) := \min_r \log_2(P(R = r)).$$

A function $\text{ext} : \{0, 1\}^\rho \times \{0, 1\}^\kappa \rightarrow \{0, 1\}^\mu$ is an (ϵ, n) -extractor if for any R with $\mathbf{H}_\infty(R) \geq n$ and K distributed uniformly over $\{0, 1\}^\kappa$ we have that $d(\text{ext}(R, K)|K) \leq \epsilon$ (see e.g. [14] for an introduction to the theory of extractors).

² More precisely suppose that \mathcal{A} takes some random input $\varrho_{\mathcal{A}}$ and the oracle takes some random input ϱ_{Ω} . Let p denote the probability (taken over $\varrho_{\mathcal{A}}$ and ϱ_{Ω}) that $\mathcal{A}(\varrho_{\mathcal{A}})$ wins the game. Then there has to exist randomness r such that $\mathcal{A}(r)$ wins with probability p . A computationally-unlimited adversary can find this r .

3.2 The Construction

The construction of the IT-secure FSS scheme of [5] used as a building-block a special type of randomness extractors called BSM-secure-functions, where BSM stands for the *Bounded-Storage Model* (see [13,6,12,15]). The need to use this special type of extractors came from the fact that originally FSS was proposed as a primitive in the Bounded-Retrieval Model, where it is crucial that the decryption function does not need to read the entire ciphertext. To be more general, in this paper we drop this assumption, and build an FSS scheme using any randomness extractor.

For completeness, in this section we review the construction [5], and prove that it is secure (this security argument appeared already implicitly in [5]). Let μ denote the length of the plaintext M and let $\text{ext} : \{0, 1\}^\rho \times \{0, 1\}^\kappa \rightarrow \{0, 1\}^\mu$ be an $(\epsilon, \rho - \sigma - \alpha)$ -extractor (for any parameter α). The key for an FSS scheme is a pair (K_0, K_1) , where $|K_0| = \kappa$ and $|K_1| = \mu$, and the encryption procedure is defined as $\text{Encr}((K_0, K_1), M) := (R, \text{ext}(R, K_0) \oplus K_1 \oplus M)$, where $R \in \{0, 1\}^\rho$ is uniformly random. The decryption is defined as $\text{Decr}((K_0, K_1), (R, X)) = \text{ext}(R, K_0) \oplus K_1 \oplus X$.

Lemma 1. *The $(\text{Encr}, \text{Decr})$ scheme constructed above is $(2\epsilon + 2^{-\alpha}, \sigma)$ -IT-secure.*

Before proving this lemma we show the following.

Lemma 2. *Modify the distinguishing game from Sect. 2 in the following way. The adversary (that we will call a weak adversary), instead of getting access to the entire ciphertext $C = (R, \text{ext}(R, K_0) \oplus K_1 \oplus M)$ (in Step 3) gets only access to R , and then in Step 4 he gets K_0 and $\text{ext}(R, K_0) \oplus M^b$. Then any σ -adversary wins this game (i.e. guesses b correctly) with probability at most $1/2 + 2^{-\alpha} + 2\epsilon$.*

Proof. Let $y = h(R)$ be the value that the adversary retrieves in Step 3. We first show that

$$P(\mathbf{H}_\infty(R|h(R=y)) \leq \rho - \sigma - \alpha) \leq 2^{-\alpha}. \quad (2)$$

Since $|h(R)| \leq \sigma$, hence the number of all y 's is at most equal to 2^σ . Therefore the number of r 's for which there exists some y such that

$$|\{r : h(r=y)\}| \leq 2^{\rho - \sigma - \alpha} \quad (3)$$

is at most $2^{\rho - \sigma - \alpha} \cdot 2^\sigma = 2^{\rho - \alpha}$. Hence the probability that it exists for a *random* $r \in \{0, 1\}^\rho$ is at most $2^{\rho - \alpha} / 2^\rho = 2^{-\alpha}$. Clearly, since R is distributed uniformly, we have that if y is such that (3) holds then

$$\mathbf{H}_\infty(R|h(R=y)) \leq \rho - \sigma - \alpha. \quad (4)$$

Thus (2) is proven. Now, since ext in an $(\epsilon, \rho - \sigma - \alpha)$ -extractor, we have that if y is such that $\mathbf{H}_\infty(R|h(R=y)) \leq \rho - \sigma - \alpha$ then $d(\text{ext}(R, K_0)|K, h(R)=y) \leq \epsilon$. Therefore in this case from the point of view of the adversary M^b is simply encrypted with a one-time pad $X = \text{ext}(R, K_0)$ such that $d(X) \leq \epsilon$. In [6] (Lemma 7) it is shown that if this is the case then the adversary can distinguish

between the ciphertexts $M^0 \oplus X$ and $M^1 \oplus X$ (for any messages M^0 and M^1) with an advantage at most $2d(X)$. Therefore the total advantage of the adversary is at most

$$\begin{aligned} P(\mathbf{H}_\infty(R|h(R=y)) \leq \rho - \sigma - \alpha) \cdot 1 + 2d(X) \\ \leq 2^{-\alpha} + 2\epsilon. \end{aligned}$$

We are now ready for the proof of Lemma 1.

Proof (of Lemma 1). We show that if there exists an adversary \mathcal{A} that breaks (Encr, Decr) with probability ξ then there exists a weak adversary \mathcal{A}' that breaks (Encr, Decr) with probability ξ . Clearly by Lemma 2, showing this will finish the proof.

The adversary \mathcal{A}' simulates \mathcal{A} in the following way. First, he starts \mathcal{A} and forwards to the oracle the messages M^0 and M^1 that \mathcal{A} produces. Then, when he gets access to R he chooses a uniformly random string $Z \in \{0, 1\}^\mu$ and gives (R, Z) to \mathcal{A} . Later (in Step 4), when he receives K and $X = \text{ext}(R, K_0) \oplus M^b$ he sets $K_0 = K$ and $K_1 = X \oplus Z$ (hence: $K_1 = \text{ext}(R, K_0) \oplus M^b \oplus Z$) and gives (K_0, K_1) to \mathcal{A} . At the end \mathcal{A}' outputs the bit b that \mathcal{A} outputs.

Set $T := \text{ext}(R, K_0) \oplus M^b$ and observe that in the original game \mathcal{A} can see the following random variables

$$R, K_0, K_1, T \oplus K_1 \tag{5}$$

(where K_0, K_1, R are uniformly random and independent) and in our simulation we have

$$R, K_0, T \oplus Z, Z \tag{6}$$

(where K_0, R, Z are uniformly random and independent). Obviously the variables in (5) and (6) have an identical joint distribution, and therefore the simulated \mathcal{A} guesses b correctly with the same probability as \mathcal{A} in a normal execution. Hence the probability that \mathcal{A} wins is equal to the probability that \mathcal{A}' wins. \square

Since randomness extractors with seed of length $O(\log k)$ are known (see e.g. [14]), in particular the non-explicit extractor that extracts almost all the entropy has seed of length $\log k + O(1)$, therefore we can conclude that there exists a (δ, σ) -IT-secure FSS scheme with key of length $|M| + O(\log |R|)$ and δ being a small constant. Since one can also construct extractors where σ is a constant fraction of $|R|$ we get that one can construct a (δ, σ) -IT-secure FSS scheme with key of length $|M| + O(\log \sigma)$.

4 The Lower Bound

In this section we present the main result of the paper. We start with the following lemma.

Lemma 3. *Let $\Phi = (\text{Encr}, \text{Decr})$ be an FSS scheme. Suppose the set \mathcal{K} of the keys is equal to $\{0, 1\}^\kappa$, for some parameter κ . There exists a σ -adversary \mathcal{A} that breaks Φ with advantage at least $1/4$, for*

$$\sigma = \frac{\kappa \cdot 2^{\kappa+1}}{|\mathcal{M}|} + 1. \quad (7)$$

Proof. We construct \mathcal{A} as follows. For every message M and a ciphertext C let

$$\mathcal{K}_{M,C} := \{K : P(\text{Encr}(K, M) = C) > 0\}.$$

Of course a computationally-unlimited machine can always compute $\mathcal{K}_{M,C}$ for given M, C , by just examining all possible K 's and all possible random inputs of the Encr algorithm. Clearly, from the correctness of the decryption, for any C and any two distinct messages M^0 and M^1 we have that

$$\mathcal{K}_{M^0,C} \cap \mathcal{K}_{M^1,C} = \emptyset. \quad (8)$$

Set $x := (\sigma - 1)/\kappa$. Therefore from (7) we have

$$x = 2^{\kappa+1} / |\mathcal{M}|. \quad (9)$$

The strategy of \mathcal{A} is as follows. First, he chooses two messages M^0 and M^1 (such that $M^0 \neq M^1$) uniformly at random. He sends M^0, M^1 to the oracle. After receiving $C = \text{Encr}(K, M^b)$ the adversary determines $\mathcal{K}_{M^0,C}$ and $\mathcal{K}_{M^1,C}$ and checks if for some $b' \in \{0, 1\}$ it is the case that $|\mathcal{K}_{M^{b'},C}| \leq x$ (if it holds for both $b' = 0, 1$ then he chooses b' arbitrarily). Denote this even with \mathcal{E} . If such b' does not exist then he sets U to be equal to an empty string. Otherwise he sets U to be equal to (\tilde{U}, b') where \tilde{U} is the binary representation of $\mathcal{K}_{M^{b'},C}$. Clearly, $\mathcal{K}_{M^{b'},C}$ can be represented (just by listing all its elements) with $|\mathcal{K}_{M^{b'},C}| \cdot \kappa = \sigma - 1$ bits, so U has length at most σ .

After learning K the adversary does the following:

1. if \mathcal{E} did not occur, i.e. U is an empty string then he outputs b uniformly at random,
2. otherwise suppose $U = (\tilde{U}, b')$. The adversary checks if K is a member of the set that \tilde{U} represents. If yes, then he outputs b' , otherwise he outputs $1 - b'$.

Clearly in the first case the probability that the adversary guesses b correctly is equal to $1/2$. It follows from (8) that in second case the probability that he guesses b correctly is equal to 1. Hence, the total probability that the adversary guesses b correctly is equal to

$$\begin{aligned} & 1/2 \cdot (1 - P(\mathcal{E})) + 1 \cdot P(\mathcal{E}) \\ &= 1/2 + 1/2 \cdot P(\mathcal{E}) \end{aligned}$$

Therefore he wins the game with advantage $1/2 \cdot P(\mathcal{E})$. Thus it remains to give a bound on the probability of \mathcal{E} , or in other words, to bound the following probability:

$$P\left(\text{there exists } b' \text{ such that } |\mathcal{K}_{M^{b'},C}| \leq x\right). \quad (10)$$

From (8) it follows that for every C we have that

$$\sum_{M \in \mathcal{M}} |\mathcal{K}_{M,C}| = \left| \bigcup_M \mathcal{K}_{M,C} \right| \leq 2^\kappa.$$

Hence, for a randomly chosen M the probability that $|\mathcal{K}_{M,C}| \geq x$ is at most equal to $2^\kappa / (x \cdot |\mathcal{M}|)$, which, from (9) is at most equal to $1/2$. We now observe that M^{1-b} is distributed completely uniformly given C (since C is a function of M^b and K)³. Therefore the probability that $|\mathcal{K}_{M^{1-b},C}| \geq x$ is at most equal to $2^\kappa / (x \cdot |\mathcal{M}|)$. This implies that the (10) is at least $1/2$. Hence, the adversary wins the game with advantage at least $1/4$. \square

Corollary 1. *For every σ consider a family of FSS schemes that is $(1/4, \sigma)$ -secure. Suppose $\mathcal{M} = \{0, 1\}^\mu$ (where μ is constant) and $\mathcal{K} = \{0, 1\}^\kappa$. Then*

$$\kappa \geq \mu + \log_2 \sigma - O(\log_2 \log_2 \sigma). \quad (11)$$

Proof. From Lemma 3 we get that

$$\sigma \leq \frac{\kappa \cdot 2^{\kappa+1}}{2^\mu} + 1.$$

This implies that:

$$\kappa \geq \mu + \underbrace{\log_2(\sigma - 1) - 1}_{\log_2(\sigma) + O(1)} - \underbrace{\log_2 \kappa}_{(*)} \quad (12)$$

Since we can assume that $\kappa \leq \mu + \log_2 \sigma$ (as otherwise (11) is proven), we get that $(*)$ is $O(\log_2 \log_2 \sigma)$. Hence (11) is proven. \square

References

1. Akavia, A., Goldwasser, S., Vaikuntanathan, V.: Simultaneous hardcore bits and cryptography against memory attacks. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 474–495. Springer, Heidelberg (2009)
2. Cash, D., Ding, Y.Z., Dodis, Y., Lee, W., Lipton, R.J., Walfish, S.: Intrusion-resilient key exchange in the bounded retrieval model. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 479–498. Springer, Heidelberg (2007)
3. Di Crescenzo, G., Lipton, R.J., Walfish, S.: Perfectly secure password protocols in the bounded retrieval model. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 225–244. Springer, Heidelberg (2006)
4. Dziembowski, S.: Intrusion-resilience via the bounded-storage model. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 207–224. Springer, Heidelberg (2006)
5. Dziembowski, S.: On forward-secure storage. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 251–270. Springer, Heidelberg (2006)

³ Note that this does not necessarily hold for M^b , since M^b can slightly depend on C .

6. Dziembowski, S., Maurer, U.M.: On generating the initial key in the bounded-storage model. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 126–137. Springer, Heidelberg (2004)
7. Dziembowski, S., Pietrzak, K.: Intrusion-resilient secret sharing. In: FOCS, pp. 227–237 (2007)
8. Dziembowski, S., Pietrzak, K.: Leakage-resilient cryptography. In: 49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, Philadelphia, PA, USA, October 25–28, pp. 293–302. IEEE Computer Society, Los Alamitos (2008)
9. Halevi, S., Rabin, T. (eds.): TCC 2006. LNCS, vol. 3876. Springer, Heidelberg (2006)
10. Alwen, Y.D.J., Wichs, D.: Leakage resilient public-key cryptography in the bounded retrieval model. In: Halevi, S. (ed.) Advances in Cryptology - CRYPTO 2009. LNCS, vol. 5677, pp. 36–54. Springer, Heidelberg (2009)
11. Katz, J.: Signature schemes with bounded leakage resilience. Cryptology ePrint Archive, Report 2009/220 (2009), <http://eprint.iacr.org/>
12. Lu, C.-J.: Encryption against storage-bounded adversaries from on-line strong extractors. *J. Cryptology* 17(1), 27–42 (2004)
13. Maurer, U.M.: Conditionally-perfect secrecy and a provably-secure randomized cipher. *J. Cryptology* 5(1), 53–66 (1992)
14. Shaltiel, R.: Recent developments in explicit constructions of extractors. *Bulletin of the EATCS* 77, 67–95 (2002)
15. Vadhan, S.P.: Constructing locally computable extractors and cryptosystems in the bounded-storage model. *J. Cryptology* 17(1), 43–77 (2004)