LE MATEMATICHE
Vol. LXXIX (2024) – Issue II, pp. 477–503
doi: 10.4418/2024.79.2.13

ALGORITHM SUBSTITUTION ATTACKS ON SYMMETRIC ENCRYPTION: A SURVEY

D. CARNEMOLLA - M. DI RAIMONDO

In 2014, Bellare, Paterson, and Rogaway suggested formalizing Algorithm Substitution Attacks (ASAs), a new type of attack against symmetric encryption methods. These attacks replace the conventional encryption algorithm with a subverted one, enabling the attacker, known as Big Brother, to decrypt messages without the user's collaboration. The formal definitions of these attacks highlight the user's capacity to identify the subversion (i.e., the replacement of regular encryption with a malicious one) and the Big Brother's capacity to gather data about encrypted messages. In recent years, the cryptographic community has developed several definitions, attacks, and possible defenses to increase its awareness of this potential issue. In this paper, we will explore the Algorithm Substitution Attack concepts and assaults available in the literature, comparing them with a critical eye.

1. Introduction

Following the public release of Snowden's revelations [14, 15], concerns have emerged regarding the potential methods government agencies (the Big Brother)

Received on September 29, 2024

AMS 2010 Subject Classification: 94A60, Cryptography
Keywords: Mass surveillance, algorithm-substitution attack, symmetric encryption

could employ for mass surveillance. The methods used appear to be various, including network router sabotage, network traffic manipulation, malware injection, and so on. There are also attempts to compromise cryptographic standards in order to insert potential backdoors into parameter generation [22]. From a theoretical standpoint, it is intriguing to consider whether they could manipulate the cryptographic tools we regularly use without our knowledge, potentially revealing confidential information such as messages and keys. Let us begin by assuming that cryptographic experts have carefully examined and confirmed the theoretical security of the chosen cryptographic scheme on paper. This is a reasonable assumption for many of the most current specific standards. A risk may arise during the transition to deployment on computers and devices: has everything been done correctly? Has an unexpected mechanism been introduced? Code accessibility often restricts verification capabilities. This is a clear limit for a commercial closed-source project. On the other hand, many reference implementations (such as OpenSSL, OpenSSH, etc.) are open-source projects, and this should provide more transparency. Regrettably, the software is often distributed as pre-compiled binaries through repositories or app stores, which suggests the potential for tampering during the distribution phase. Finally, as a general observation, these projects are often very complex, so only a small group of selected experts can understand and really review the code. This implies that a malicious developer could potentially introduce a modification or backdoor into the source code, which could remain undetected if well-obfuscated. These considerations allow us to state that this subversion issue can become relevant for open-source software too.

This poses intriguing questions. If a hypothetical Big Brother compromises our cryptographic tools, will we be able to detect such manipulations through regular usage (for example, by studying their output and/or behavior)? Are there cryptographic algorithms that, by design, cannot be compromised in any way? All of this has led to studies on Algorithm Substitution Attacks (ASAs) [3, 27, 28]. In these scenarios, users are unaware of the replacement of regular cryptographic tool implementations with altered versions (malware). Because such a replacement may not be global, or there may be other unaffected implementations, malicious software must still ensure a certain level of interoperability (i.e., the sabotaged encrypted messages should look regular and be decryptable with the standard routine). Questions like these can be posed concerning any cryptographic tool (symmetric or asymmetric encryption, digital signatures, key exchange protocols, etc.). For practical reasons, various studies, including this survey, have chosen to focus on the specific case of symmetric encryption. Indeed, this option favors a greater focus on the problem and provides more meaningful results. Additionally, they are frequently combined with other cryptographic primitives and are widely utilized in a variety of scenarios, especially for the provided high efficiency.

A proper approach is required to thoroughly investigate a problem like this. To assess if something is safe or not, we must first define what it means to be secure. In other words, it is crucial to have a suitable definition. Bellare *et al.* (hereinafter BPR) were among the first in [3] to address the definitional issue of ASA for symmetric encryption. They designed a model that formally demonstrated several intriguing results. They pioneered the first generic subversion methods, enabling blackbox modification of an entire class of encryption algorithms into something beneficial for Big Brother. Such results also allow us to show, under certain conditions (e.g., use of initialization vectors or, more generally, randomization), that regular users will not notice the subversion at all. Unfortunately, the traits exploited by such attacks are quite common in the existing cryptographic standards. This may make us think there are no safe ways to avoid such threats. However, the BPR investigation identified encryption classes that are difficult to subvert, at least not without easy detection during use. This, at least theoretically, provides some hope to the common user. At first glance, designing a definition appears to be a simple task. Correctly defining the scheme and associated attacks is critical: what is the attacker's goal? What powers does this enemy possess? How do the various parts interact with the others? As a symptom of this difficulty, in the specific case of ASA for symmetric encryption, there are at least two other works that have led to a revision of what BPR initially proposed. In 2015, Bellare addressed various problems of the prior model in [2], along with a different working group (BJK from now on). This alternative model also allowed for an expansion of the class of available attacks and improved the already known subversion schemes. In the same year, Degabriele *et al.* (DFP hereafter) critiqued the BPR's model, proposed their own, and finally presented a simple yet effective new method of subversion that had not previously been explored. The presence of several works on a given topic, some of which are inconsistent and not necessarily inclusive, is often symptomatic of the need for further research. The present survey aims to collect, document, unify, and compare the major findings on the subject: definitions, subversion schemes, and immune cryptographic schemes. This type of contribution intends to aid other researchers in swiftly understanding the topic, potentially leading to a unified solution that consolidates existing know-how.

It is also important to understand the potential limitations of a formal approach of this kind. Consider how a typical user could spot anomalies in their system's behavior. The definitions do not account for the potential of the user reverse-engineering the binary executable code or conducting advanced studies on so-called side-channels, such as execution time or the number of requests to the system's RNG (Random Number Generator). Although this reduces the practical value of some stated results, they remain important milestones for a deeper understanding of the problem.

1.1. Related Works

The origins of ASA can be traced back to Simmons' [23–25] research on the leakage of secret keying material via *subliminal channels* in blackbox implementations. This notion is further investigated by Young and Yung's *Kleptography* thread [27–31], which focuses on public key encryption and signature techniques. Unlike the symmetric setting addressed in this paper, such a scheme would allow Big Brother's subversion key to be preserved even in case of reverse-engineering of the subverted code. Other research, including [26] and [1], have further refined the asymmetric case. Even PRNGs (Pseudo Random Number Generators) have been scrutinized, with speculation on potential ASA assaults [20]. To validate the concrete risk that ASA represents, various works have targeted real protocols such as SSL/TLS, SSH, Wireguard, and Signal [6, 13, 32]. The ongoing research on the topic has shown several connections between ASA and other cryptographic primitives. Anamorphic Encryption and ASA have a tight link, as proven by [26]. In [5], the authors demonstrated that ASAs and steganographic systems against primitives are equivalent.

2. Preliminaries

Notation. If n is a positive integer, we denote with $\{0,1\}^n$ the set of all binary strings of length n and with $\{0,1\}^*$ the set of all finite binary strings. The symbol ε is used to represent the empty string. We denote with $x||y$ the concatenation of two strings x and y and with $|x|$ the length of a string x. Vectors are indicated with bold fonts. For any vector $\mathbf{X}$, we denote with $\mathbf{X}[i]$ its i-th component. Finally, if A is a set we denote with $|A|$ its size and with $y \overset{\$}{\leftarrow} A$ the process of extracting an element from A uniformly at random and assigning it to y. We denote with $\lambda \in \mathbb{N}$ a security parameter. A function $f : \mathbb{N} \to \mathbb{R}^+$ is called *negligible* if for every positive polynomial $p(\cdot)$ there exists a λ_0 such that, for every $\lambda > \lambda_0$ it holds that $f(\lambda) < 1/p(\lambda)$. We use $\mathsf{negl}(\lambda)$ to denote a generic negligible function.

Definition 1 (Pseudo-random function)**.** Let $F : \{0,1\}^\ell \times \{0,1\}^* \to R$ be a function taking a key $L \in \{0,1\}^\ell$ and input $c \in \{0,1\}^*$ to return an output $F(L,c) \in R$. Let $\mathcal{A}$ an adversary against the game $\mathrm{PRF}_F^{\mathcal{A}}$ shown in Figure 1. The advantage of $\mathcal{A}$ is defined as follows

$$\mathsf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) = 2 \cdot \Pr\left[\mathrm{PRF}_F^{\mathcal{A}} = 1\right] - 1.$$

```
Game $\mathsf{PRF}_F^{\mathcal{A}}$
    $L \xleftarrow{\$} \{0,1\}^\ell; b \xleftarrow{\$} \{0,1\}; C \leftarrow \emptyset$
    $b' \xleftarrow{\$} \mathcal{A}^{\textsc{Fn}}$
    return $(b = b')$

$\textsc{Fn}(c)$
    if $(b = 1)$ then $y_c \leftarrow F(L,C)$
    else if $c \notin C$ then
        $y_c \xleftarrow{\$} R$
        $C \leftarrow C \cup \{c\}$
    return $y_c$
```

Figure 1: PRF security game.

We say that F is a pseudorandom function if $\mathsf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) \leq \mathsf{negl}(\lambda)$ for all probabilistic polynomial-time (PPT) attackers $\mathcal{A}$.

2.1. Symmetric Encryption

In this section, we briefly recall the fundamental definitions of symmetric cryptography. In general, we denote with $\mathcal{M}$ the message space and with $\mathcal{AD}$ the associated data space. The associated data typically represent additional public information sent with the private message; in the case of authenticated encryption, they are included in the integrity checks.

Definition 2 (Symmetric Encryption Scheme)**.** A symmetric encryption scheme is a triple $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, as follows:

- $\mathcal{K}$ is the key space; it is a finite nonempty set of strings of some fixed length.
- $\mathcal{E}_K(M,A,\sigma) \xrightarrow{\$} (C,\sigma')$ is the encryption algorithm. This algorithm may be randomized, stateful, or both. On input a key $K \in \mathcal{K}$, a message $M \in \mathcal{M}$, an associated data $A \in \mathcal{AD}$ and a (possible) state σ, it returns a ciphertext C (or $\perp$ in case of error) and a (possible) new state σ'.
- $\mathcal{D}_K(C,A,\sigma) \rightarrow (M,\sigma')$ is the decryption algorithm. It is a deterministic algorithm which, on input a key $K \in \mathcal{K}$, a ciphertext C, an associated data $A \in \mathcal{AD}$ and a (possible) state σ, returns a message $M \in \mathcal{M}$ (or $\perp$ in case of error) and a (possible) new state σ'.

The encryption and decryption states are always initialized to ε.

Definition 3 (Stateless Encryption (resp. Decryption) Algorithm)**.** An encryption algorithm $\mathcal{E}$ (resp., a decryption algorithm $\mathcal{D}$) is said to be stateless if for all $K \in \mathcal{K}$, $M \in \mathcal{M}$, $A \in \mathcal{AD}$ the updated state returned by $\mathcal{E}_K(M,A,\varepsilon)$ (resp., $\mathcal{D}_K(C,A,\varepsilon)$) remains ε.

Definition 4 (Stateless Symmetric Encryption Scheme)**.** A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said stateless if both $\mathcal{E}$ and $\mathcal{D}$ are stateless.

In a stateless scheme the notation can omit the state σ. For short, for any $\ell \in \mathbb{N}$, any vector of messages $\mathbf{M} = [M_1, \ldots, M_\ell] \in \mathcal{M}^\ell$ and any vector of additional data $\mathbf{A} \in \mathcal{AD}^\ell$, $(\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon)$ denotes the sequential encryption of messages in $\mathbf{M}$. A similar notation, $(\mathbf{M}, \sigma_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon)$, is intended for the sequential decryption of a vector of ciphertexts.

Definition 5 (Correctness)**.** A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to be (q, δ)-correct if for all $\ell \le q$, all $\mathbf{M} \in \mathcal{M}^\ell$, and all $\mathbf{A} \in \mathcal{AD}^\ell$,

$$\Pr\left[K \leftarrow^{\$} \mathcal{K}; (\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon); (\mathbf{M}', \rho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon) : \mathbf{M} \neq \mathbf{M}'\right] \le \delta.$$

If Π satisfies correctness with $\delta = 0$ for all $q \in \mathbb{N}$ then it is said to be perfectly correct.

The security definition of Indistinguishability against Chosen-Ciphertext Attack (IND-CPA) follow.

Game IND-CPA$_\Pi^\mathcal{A}$	LR(M_0, M_1, A)
$b \leftarrow^{\$} \{0,1\}$	if $\lvert M_0 \rvert \neq \lvert M_1 \rvert$ then return $\perp$
$\sigma \leftarrow \varepsilon; K \leftarrow^{\$} \mathcal{K}$	$(C, \sigma) \leftarrow \mathcal{E}_K(M_b, A, \sigma)$
$b' \leftarrow \mathcal{A}^{\mathrm{LR}}$	**return** C
return $(b = b')$	

Figure 2: IND-CPA security game.

Definition 6 (Security)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\mathcal{A}$ be an adversary. The adversary's advantage against IND-CPA$_\Pi^\mathcal{A}$ (described in Figure 2) is defined as follows

$$\mathsf{Adv}_\Pi^{\text{ind-cpa}}(\mathcal{A}) = 2 \cdot \Pr\left[\text{IND-CPA}_\Pi^\mathcal{A}\right] - 1.$$

Π is said to be IND-CPA secure if the above advantage is negligible for all PPT adversaries $\mathcal{A}$.

Definition 7 (Coin Injectivity)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Π is said coin injective if for each fixed key $K \in \mathcal{K}$, message $M \in \mathcal{M}$ and associated data A, $\mathcal{E}_K(M, A; r)$ is injective for each r.

Definition 8 (Min-Entropy)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a scheme for symmetric encryption. The min-entropy $\mathbf{H}_\infty(\Pi)$ of the scheme Π is defined as follows

$$2^{-\mathbf{H}_\infty(\Pi)} = \max_{K,M,C} \Pr\left[\mathcal{E}_K(M; r) = C\right]$$

3. Algorithm-Substitution Attacks

In an Algorithm Substitution Attack (ASA) scenario, an adversary (Big Brother) is able to replace the standard encryption routine $\mathcal{E}$ with a custom implementation $\widetilde{\mathcal{E}}$. This subverted implementation can leverage additional inputs, directly embedded within the code, known only to the adversary: a subversion key $\widetilde{K}$. The strategy aims to create an alternative method for extracting plaintexts from users' ciphertexts, represented by a specific plaintext-recovery algorithm $\widetilde{\mathcal{D}}$. This could involve some methodology, more or less hidden, to transmit the user's key K to Big Brother (*Key Recovery*) and then proceed to standard decryption. More formally:

Definition 9 (Subversion [3]). Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. A subversion of Π is a triple $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$, as follows:

- $\widetilde{\mathcal{K}}$ is the *subversion key space*; it is a finite nonempty set.
- $\widetilde{\mathcal{E}}_{\widetilde{K},K}(M, A, \sigma, i) \to (C, \sigma')$ is the *subverted encryption algorithm*. This algorithm may be randomized, stateful or both. On input a subversion key $\widetilde{K} \in \widetilde{\mathcal{K}}$, a user's secret key $K \in \mathcal{K}$, a message $M \in \{0,1\}^*$, an associated data string $A \in \{0,1\}^*$, a (possible) state σ and a user identifier i, it returns a ciphertext C (or $\perp$ in case of error) and a (possible) new state σ'.
- $\widetilde{\mathcal{D}}_{\widetilde{K}}(\mathbf{C}, \mathbf{A}, i) \to \mathbf{M}$ is the *plaintext-recovery algorithm*. It is a deterministic algorithm which, on input a subversion key $\widetilde{K} \in \widetilde{\mathcal{K}}$, a vector of ciphertexts $\mathbf{C}$, a vector of associated date $\mathbf{A}$ and an identifier i for the key K used by the attacked user, it returns a vector of messages $\mathbf{M}$.

The user identifier i used above provides support for the multi-user scenario and formalizes the possibility of the adversary to identify each user by means of some kind of public unique information (e.g., MAC address, IP address). It is important to note that the plaintext-recovery algorithm $\widetilde{\mathcal{D}}$ is not meant to replace the normal decryption algorithm $\mathcal{D}$ that regular users employ. Instead, it encodes the strategy adopted by the adversary to recover, exploiting the knowledge of the subversion key $\widetilde{K}$, the plaintexts from a batch of eavesdropped ciphertexts. The multi-message approach is useful to formalize some recovery strategies that require the systematic collection of a few bits of some kind of sensible information (e.g., the user key K) in order to gain the knowledge necessary to invert the encryption of any ciphertext.

In order to be effective in the long term, the subverting strategy must be undetectable by regular users when using $\widetilde{\mathcal{E}}$ instead of $\mathcal{E}$. In this sense, the minimum condition is that the regular decryption algorithm $\mathcal{D}$, using the regular key K, continues to work on the subverted ciphertexts. The proposed formal definition follows.

Definition 10 (Decryptability [10])**.** Let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of a symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. $\widetilde{\Pi}$ satisfies (q, δ)-decryptability if $(\widetilde{\mathcal{K}} \times \mathcal{K}, \widetilde{\mathcal{E}}, \mathcal{D}')$ is a (q, δ)-correct encryption scheme where $\mathcal{D}'$ is defined as $\mathcal{D}'_{\widetilde{\mathcal{K}},\mathcal{K}}(C, A, \sigma) = \mathcal{D}_{\mathcal{K}}(C, A, \sigma)$.

If $\widetilde{\Pi}$ is $(q, 0)$-decryptable with respect to Π for all $q \in \mathbb{N}$, it is said to be perfectly decryptable. This is the only case considered by BPR in their first model [3]: this, as shown later, will represent a limitation and it does not permit to formalize some specific effective attacks.

3.1. Detectability and Surveillance

The ability to continue using the regular decryption procedure to open a ciphertext generated by a compromised encryption algorithm ensures interoperability and protects one from being swiftly identified by the compromised victim. On the other hand, this is not enough to formalize other anomalies that a regular user could observe in the altered algorithm's behavior. As a result, the detection and surveillance advantages were introduced in first security model in [3]. Intuitively, the detection advantage models the user's (i.e., victim's) ability to detect that a subversion occurred, while the surveillance advantage models the attacker's effectiveness in stealing user data using it. There are different variants of these definitions in the specific literature, each intended to capture different attacker/user capabilities. We report and discuss these definitions, highlighting the differences in the following sections.

3.1.1. BPR Security Model [3]

The former definitions of detectability and surveillance advantages were proposed by BPR in [3]. The security games for their BPR model are depicted in Figure 3. Let Π be a symmetric encryption scheme and $\widetilde{\Pi}$ its subversion. In the game $\mathrm{DETECT}^{\mathcal{U}}_{\Pi,\widetilde{\Pi}}$, the user $\mathcal{U}$ (the detector) is challenged to guess whether it is in the real or in the subverted world. In the real world, the standard encryption algorithm $\mathcal{E}$ is used, whereas in the subverted world, the adversarial altered encryption algorithm $\widetilde{\mathcal{E}}$, jointly with a random subversion key $\widetilde{K}$, is employed. At the beginning of the game, a uniformly chosen random bit b is selected; it determines if the user $\mathcal{U}$ will run against the real encryption algorithm or the subverted one. If the output bit b' matches b, $\mathcal{U}$ wins the game. During its execution, the user $\mathcal{U}$ has access to an oracle KEY, which models its ability to obtain the keys of any regular user he wants to impersonate; indeed this is a multi-user game. Obviously, the user does not have access to the adversary's subversion key $\widetilde{K}$. The encryption oracle ENC permits $\mathcal{U}$ to interact with the encryption functionality whose behavior depends on the bit b. This means that the detection advantage of $\mathcal{U}$ measures the user's ability to know if he is receiving ciphertexts produced by $\mathcal{E}$ or by $\widetilde{\mathcal{E}}$. Their formal definition of detection advantage is given in Definition 11. In the game suggested by BPR [3], it is important to note that the detector $\mathcal{U}$ is modelled so that it has the decryptor's point of view. This is demonstrated by the fact that the encryption oracle does not return the encryption state σ to $\mathcal{U}$. This means that the detector can't inspect this state; as noted in BJK [2], this can be very limiting in modeling the ways a regular user could detect the anomalies introduced by the subversion. Given this, the aim of the BJK model, reported in Section 3.1.2, is to overcome such a limit, adopting as a point of view the one of the encryptor.

The other proposed game, $\text{SURV}^{\mathcal{B}}_{\Pi,\widetilde{\Pi}}$, aims to measure the ability of a Big Brother $\mathcal{B}$ to extract information from subverted ciphertexts; the obvious goal would be full decryption, but the game, following the standard style of semantic security, attempts to capture the leakage of any bit of information. For this reason, similarly to the previous one, it challenges $\mathcal{B}$ to recognize the nature of the encryption functionality (real or subverted) behind the oracle ENC while giving him access only to its subversion key $\widetilde{K}$. An oracle KEY is also present, formally useful for triggering on demand the generation of user keys, but it always returns ε. The surveillance advantage is formally defined in Definition 13.

```
Game $\text{DETECT}^{\mathcal{U}}_{\Pi,\widetilde{\Pi}}$
$b \leftarrow^{\$} \{0,1\}; \widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}; b' \leftarrow \mathcal{U}^{\text{KEY},\text{ENC}}$
return $(b = b')$

KEY$(i)$
if $(K_i = \perp)$ then $K_i \leftarrow^{\$} \mathcal{K}; \sigma_i \leftarrow \varepsilon$
return $K_i$

ENC$(M, A, i)$
if $(K_i = \perp)$ then return $\perp$
if $(b = 1)$ then
    $(C, \sigma_i) \leftarrow^{\$} \mathcal{E}_{K_i}(M, A, \sigma_i)$
else $(C, \sigma_i) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K},K_i}(M, A, \sigma_i, i)$
return $C$
```

```
Game $\text{SURV}^{\mathcal{B}}_{\Pi,\widetilde{\Pi}}$
$b \leftarrow^{\$} \{0,1\}; \widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}; b' \leftarrow \mathcal{B}^{\text{KEY},\text{ENC}}$
return $(b = b')$

KEY$(i)$
if $(K_i = \perp)$ then $K_i \leftarrow^{\$} \mathcal{K}; \sigma_i \leftarrow \varepsilon$
return $\varepsilon$

ENC$(M, A, i)$
if $(K_i = \perp)$ then return $\perp$
if $(b = 1)$ then
    $(C, \sigma_i) \leftarrow^{\$} \mathcal{E}_{K_i}(M, A, \sigma_i)$
else $(C, \sigma_i) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K},K_i}(M, A, \sigma_i, i)$
return $C$
```

Figure 3: BPR security model.

Definition 11 (Detection Advantage [3])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{U}$ be an algorithm executed by users to detect the subversion. The advantage of $\mathcal{U}$ is defined as follows

$$\mathsf{Adv}^{\text{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) = 2 \cdot \Pr\left[\text{DETECT}^{\mathcal{U}}_{\Pi,\widetilde{\Pi}} = 1\right] - 1$$

where the game DETECT is the one shown in Figure 3.

Definition 12 (Undetectability)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. We say that $\widetilde{\Pi}$ is undetectable if for any PPT adversary $\mathcal{U}$, holds

$$\mathsf{Adv}^{\text{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq \mathsf{negl}(\lambda)$$

Definition 13 (Surveillance Advantage [3])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{B}$ an adversary representing the Big Brother. The advantage of $\mathcal{B}$ is defined as follows

$$\mathsf{Adv}^{\text{srv}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}) = 2 \cdot \Pr\left[\text{SURV}^{\mathcal{B}}_{\Pi,\widetilde{\Pi}} = 1\right] - 1$$

where the game SURV is the one shown in Figure 3.

These definitions were provided by BPR in [3] jointly with the notion of (perfect) decryptability, where each ciphertext produced by $\widetilde{\mathcal{E}}$ can be correctly decrypted by the user using $\mathcal{D}$ with his key K. To prove that an encryption scheme cannot be subverted, it is necessary to show that for any possible subversion $\widetilde{\Pi}$, the subverter $\mathcal{B}$ has a not-significant surveillance advantage. Note that if the quantification does not restrict to (perfectly) decryptable subversion schemes, finding a scheme with this level of resilience may become impossible. Indeed, it is inevitably possible to construct an artificious subversion scheme $\widetilde{\Pi}$ that is always distinguishable by $\mathcal{B}$ but not perfectly decryptable. Consider a subversion in which an additional bit 0 is tied to the ciphertext; $\mathcal{B}$'s task would become trivial, but the regular decryption routine in Π would fail (for the unexpected extra bit). The formal definition of subversion resistance follows.

Definition 14 (Subversion Resistance)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Π is said to be subversion resistant if for any perfectly decryptable subversion $\widetilde{\Pi}$ and any adversary $\mathcal{B}$, holds

$$\mathsf{Adv}^{\text{surv}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}) \leq \mathsf{negl}(\lambda).$$

3.1.2. BJK Security Model [2]

As noted above, in the BPR security model, the subversion detector is essentially the decrypter. This, as noted by BJK in [2], can limit the detector's investigating capabilities. The encryptor, the entity sending encrypted messages, is another actor who is undoubtedly interested in detecting any potential subversion of the employed tools. He possesses certain advantages over the decryptor: indeed, in the case of stateful schemes, he has the ability to inspect the state of the encryption algorithm (whether subverted or not) by inspecting the device memory but also to apply on it some ordinary operations (e.g., reset or clone virtual machines). This can lead to concrete detection strategies: for example, upon reset, a stateful subverted algorithm could anomaly behave (see attack in Section 4.1.1, where the first outputted IV is always the encryption of the subverted key). For this reason, BJK [2] introduced a stronger notion of undetectability, called strong undetectability. In their BJK security model, the subversion distinguisher $\mathcal{U}$ has access to an oracle ENC which takes not only the message M to be encrypted but also the key K to use. This allows us to model the detector's ability to pick the key K as it wants (as well as knowing it). Furthermore, the oracle returns not only the ciphertext C but also the updated encryption state σ. This models the possibility for the encryptor to observe anomalies and, jointly with a new RESET oracle, to reset it. The authors observe that the possibility to use the RESET oracle implies that an effective subversion strategy, with respect to their stronger notion of undetectability, can't be stateful. Indeed, a detector always has the possibility to systematically nullify such state using the reset capability before every ENC invocation. Finally, we stress that strong undetectability implies undetectability; this can be easily proven by reduction.

Another notable change in the BJK model is to focus on the effective meaning of a successful subversion attack. Instead of the indistinguishability-based notion for Definition 11, they adopt a Key Recovery game where the adversarial goal is to recover the user key K. From the perspective of the subverter, this is a more robust goal, but it results in a weaker security definition. This can be useful in modeling concrete subversion attacks. At the beginning of the new game KR, a user key K to guess and a subversion key $\widetilde{K}$ are chosen. Then, the attacker $\mathcal{B}$ is executed by running its $\mathcal{B}$.EXT key extraction algorithm, which returns a guess $\hat{K}$ for the user key K. During its execution, the attacker has access to an encryption oracle ENC with state σ. This oracle does not take as input a given message to be encrypted but this is queried from a new oracle $\mathcal{M}$, with a different state σ', for message sampling. Specifically, the oracle $\mathcal{M}$ models the user's requests for encryption of specific messages; in fact, in a real scenario, the authors consider reasonable that the Big Brother cannot choose the messages to be encrypted. If the attacker's guess is correct ($\widetilde{K} = K$), it wins.

It is significant to note that while BPR attacks target multiple users, these new games focus on a single user. However, this arrangement does not limit the detector's functionality in SDETECT game since he can utilize any key without the limitation imposed by the KEY oracle. In Figure 4, we report the pseudocodes of the security games. Then we recall the formal definitions for this security model.

```
Game SDETECT$^{\mathcal{U}}_{\Pi,\widetilde{\Pi}}$
$\widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}$ ; $b \leftarrow^{\$} \{0,1\}; \sigma \leftarrow \varepsilon$
$b' \leftarrow \mathcal{U}^{\text{ENC,RESET}}$
return $(b = b')$

ENC$(K, M)$
if $(b = 1)$ then $(C, \sigma) \leftarrow^{\$} \mathcal{E}_K(M, A, \sigma)$
else $(C, \sigma) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K},K}(M, A, \sigma)$
return $(C, \sigma)$

RESET()
$\sigma \leftarrow \varepsilon$
```

```
Game KR$^{\mathcal{B}}_{\Pi,\mathcal{M}}$
$K \leftarrow^{\$} \mathcal{K}; \widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}$ ; $\sigma \leftarrow \varepsilon; \sigma' \leftarrow \varepsilon$
$\hat{K} \leftarrow^{\$} \mathcal{B}.\text{EXT}^{\text{ENC}}(\widetilde{K})$
return $(K = \hat{K})$

ENC()
$(M, \sigma') \leftarrow^{\$} \mathcal{M}(\sigma')$
$(C, \sigma) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K},K}(M, \sigma)$
return $C$
```

Figure 4: BJK security model.

Definition 15 (Strong Detection Advantage [2])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{U}$ be a detection adversary. The advantage of $\mathcal{U}$ is defined as follows

$$\mathsf{Adv}^{\text{sdet}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) = 2 \cdot \Pr\left[\text{SDETECT}^{\mathcal{U}}_{\Pi,\widetilde{\Pi}} = 1\right] - 1$$

where the game SDETECT is the one shown in Figure 4.

Definition 16 (Strong Undetectability)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. We say that $\widetilde{\Pi}$ is undetectable if for any PPT adversary $\mathcal{U}$, holds

$$\mathsf{Adv}^{\text{sdet}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq \mathsf{negl}(\lambda)$$

Definition 17 (Key Recovery Advantage [2])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{B}$ an adversary against the game KR in Figure 4. The advantage of $\mathcal{B}$ is defined as follows

$$\mathsf{Adv}^{\text{kr}}_{\Pi,\mathcal{M}}(\mathcal{B}) = \Pr\left[\text{KR}^{\mathcal{B}}_{\Pi,\mathcal{M}} = 1\right].$$

Definition 18 (Key Recovery Subversion Resistance)**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Π is said to be key recovery subversion resistant if for any perfectly decryptable subversion $\widetilde{\Pi}$ and any adversary $\mathcal{B}$, holds

$$\mathsf{Adv}^{\mathrm{kr}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}) \leq \mathsf{negl}(\lambda).$$

3.1.3. DFP Security Model [10]

In [10], DFP criticizes the BPR model, highlighting some limitations in the proposed definitions. As stated in Section 3.1.1, BPR's subversion resistance requires perfect decryptability. BPR also present it as the minimum level of undetectability that any concrete subversion attack should provide. On the other hand, their notion of undetectability is not very comparable to perfect decryptability; in fact, the former does not imply the latter. As a counter-example, consider an input-triggered attack proposed by DFP in [10] and fully reported in Section 4.3. Here, we illustrate the basic idea in a simplified attack: this subversion $\widetilde{\Pi}$ makes use of a regular encryption $\mathcal{E}$ for all messages except for a special message $\bar{M}$ (the trigger), upon which the corresponding ciphertext would be $\bar{C} = \widetilde{\mathcal{E}}_{\widetilde{K},K}(\bar{M}) = K$. Even though this particular ciphertext $\bar{C}$ couldn't be inverted using the standard decryption routine, it would provide a clean transmission channel to pass the leaked key K. In other words, without knowledge of the special trigger message $\bar{M}$, this subversion would be undetectable, but on the other hand, it could not meet the condition of perfect decryptability. DFP considers this a limitation of the prior model, as it excludes some effective attacks that do not meet the criticized strong correctness condition.

As a first attempt to fix the issue, DFP relaxed the decryptability condition, introducing the more general definition of (q, δ)-decryptability (reported as Definition 10), where δ is small but not null. Unfortunately, this is not sufficient as the resulting model would become unsatisfiable with a total lack of any subversion-resistant schemes; indeed, the previous input-triggered attack can be applied to any (randomized, deterministic, or stateful) encryption scheme Π, and the resulting subversion would be (q, δ)-decryptable (with negligible δ) but undetectable. This would leave us with no hope of building frameworks that can offer resistance. Their novel security model (DFP from now on), depicted in Figure 5, required a deeper rework on the underlying games, trying to formalize the capacity of the detector $\mathcal{U}$ to catch anomalies in the (subverted) framework during the active exploitation by the Big Brother $\mathcal{B}$. In the designed $\overline{\text{DETECT}}$ game, unlike the BPR's DETECT, the detector $\mathcal{U}$ loses direct access to the KEY and ENC oracles but gains the ability to consult a full transcript T of $\mathcal{B}$ during an active subverting session. In this game, the Big Brother $\mathcal{B}$ runs before the detector $\mathcal{U}$, and it is challenged to distinguish between the real and the subverted world (as in the $\overline{\text{SURV}}$ game), making use of the KEY and ENC oracles. Needless to say, $\mathcal{B}$ has sole access to the subversion key $\widetilde{K}$ and not the user key K. Whenever $\mathcal{B}$ interacts with an oracle, the typescript T is updated: in the case of the KEY oracle, the pair (K_i, i) is added to T, while in the case of ENC, the triple (M, A, C) is appended. It is important to note that the provided bit guess b' returned by $\mathcal{B}$ is discharged as we are only interested in the transcript. The detector $\mathcal{U}$, who has access to the transcript T, is asked to provide his own guess b''. He wins when $b = b''$. The $\overline{\text{SURV}}$ game remains similar to BPR model and is presented for completeness. When comparing the games of this new model to the two previous ones, we can observe how this, similar to what was seen in BJK, is defined as single-user (note the restriction on the invocability of the KEY oracle). This is not a limitation: the advantage can still be calculated considering a reduction by a polynomial factor based on the number of users. Like in SURV of BPR, Big Brother $\mathcal{B}$ can apply a chosen-message attack in DFP's $\overline{\text{SURV}}$ game: indeed it represents an essential aspect in an input-triggered attack. The BJK's KR game lacks such freedom, preventing $\mathcal{B}$ from using the trigger mechanism to recover the key (no advantage). Finally, we see how the detector $\mathcal{U}$ in SDETECT, like in DETECT, acts again as a decryptor, without the possibility to verify the encryption algorithm's possible state.

The adversarial advantages and related definitions are almost identical to the DFP model with the exception of surveillance resistance (Definition 22 below), where the two advantages are made related. This allows us to reinforce the idea that Big Brother is a *cautious-but-malicious* attacker, indeed, in this definition, a scheme is intuitively considered unsubvertible if the detector has a greater advantage than the Big Brother, even if the latter is not negligible.

Definition 19 (Detection Advantage [10]). Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{U}$ be an algorithm executed by users to detect the subversion aided by the transcript produced by $\mathcal{B}$. The advantage of $\mathcal{U}$ is defined as follows

$$\mathsf{Adv}^{\overline{\text{det}}}_{\Pi,\widetilde{\Pi}}(\mathcal{B},\mathcal{U}) = 2 \cdot \Pr\left[\overline{\text{DETECT}}^{\mathcal{B},\mathcal{U}}_{\Pi,\widetilde{\Pi}} = 1\right] - 1$$

where the game $\overline{\text{DETECT}}$ is the one shown in Figure 5.

Definition 20 (Undetectability). Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. We say that $\widetilde{\Pi}$ is undetectable if for any PPT adversary $\mathcal{U}$ and any Big Brother $\mathcal{B}$, holds

$$\mathsf{Adv}^{\overline{\text{det}}}_{\Pi,\widetilde{\Pi}}(\mathcal{B},\mathcal{U}) \leq \mathsf{negl}(\lambda)$$

```
Game $\overline{\text{DETECT}}^{\mathcal{B},\mathcal{U}}_{\Pi,\widetilde{\Pi}}$
$b \leftarrow^{\$} \{0,1\}; \widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}$
$b' \leftarrow \mathcal{B}^{\text{KEY},\text{ENC}}(\widetilde{K})$ ; $b'' \leftarrow \mathcal{U}(T)$
return $(b = b'')$

KEY$(i)$        // called once
if $(K_i = \perp)$ then $K_i \leftarrow^{\$} \mathcal{K}; \sigma_i \leftarrow \varepsilon$
$T \leftarrow (K_i, i)$
return $\varepsilon$

ENC$(M, A, i)$
if $(K_i = \perp)$ then return $\perp$
if $(b = 1)$ then $(C, \sigma_i) \leftarrow^{\$} \mathcal{E}_{K_i}(M, A, \sigma_i)$
else $(C, \sigma_i) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K}, K_i}(M, A, \sigma_i, i)$
$T \leftarrow T || (M, A, C)$
return $C$
```

```
Game $\overline{\text{SURV}}^{\mathcal{B}}_{\Pi,\widetilde{\Pi}}$
$b \leftarrow^{\$} \{0,1\}; \widetilde{K} \leftarrow^{\$} \widetilde{\mathcal{K}}$
$b' \leftarrow \mathcal{B}^{\text{KEY},\text{ENC}}(\widetilde{K})$
return $(b = b')$

KEY$(i)$        // called once
if $(K_i = \perp)$ then $K_i \leftarrow^{\$} \mathcal{K}; \sigma_i \leftarrow \varepsilon$
return $\varepsilon$

ENC$(M, A, i)$
if $(K_i = \perp)$ then return $\perp$
if $(b = 1)$ then
    $(C, \sigma_i) \leftarrow^{\$} \mathcal{E}_{K_i}(M, A, \sigma_i)$
else
    $(C, \sigma_i) \leftarrow^{\$} \widetilde{\mathcal{E}}_{\widetilde{K}, K_i}(M, A, \sigma_i, i)$
return $C$
```

Figure 5: DFP security model.

Definition 21 (Surveillance Advantage [10])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be a subversion of it. Let $\mathcal{B}$ an adversary representing the Big Brother. The advantage of $\mathcal{B}$ is defined as follows

$$\mathsf{Adv}^{\overline{\text{srv}}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}) = 2 \cdot \Pr\left[\overline{\text{SURV}}^{\mathcal{B}}_{\Pi,\widetilde{\Pi}} = 1\right] - 1$$

where the game $\overline{\text{SURV}}$ is the one shown in Figure 5.

Definition 22 (Subversion Resistance [10])**.** A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to be subversion resistant with respect to a (universal) detection algorithm $\mathcal{U}$ if for all efficient adversaries $\mathcal{B}$ and all efficient subversions $\widetilde{\Pi}$,

$$\mathsf{Adv}^{\overline{\text{srv}}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}) \leq \mathsf{Adv}^{\overline{\text{det}}}_{\Pi,\widetilde{\Pi}}(\mathcal{B}, \mathcal{U}).$$

4. Attacks and Analysis

In this section we report the most representative black-box ASAs and their practical impact. For brevity, we have omitted the proofs of the theorems. For details, see the original papers.

4.1. The IV-replacement Attacks [3]

The IV-replacement attack proposed by BPR in [3] is a generic attack that can be deployed against any randomized stateless encryption scheme that surface/expose its IV (i.e., initialization vector). More formally, an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to surface its IV if there exists an efficient algorithm $\mathcal{X}$ such that $\mathcal{X}(\mathcal{E}_K(M,A;IV)) = IV$ for all K,M,A,IV. This formalizes the possibility to extract the original IV, used during the encryption phase, just by observing the resulting ciphertext. This condition is quite common and is satisfied by some very widely used schemes, such as CBC$ (CBC mode with random IV) and CTR$ (CTR mode with random starting point). The authors have put forth two variations of this attack, differing primarily in whether the subverted encryption algorithm maintains a state.

4.1.1. Stateful Variant

The first proposed attack, depicted in Figure 6, requires an internal state for the subverted encryption algorithm. The main idea here is to hide the user key K inside the IV value used during the encryption of the first message (with respect to the message sequence tracked by the state). Instead of a random value, an encryption of the value K, using the subversion key $\widetilde{K}$, is employed. Verifying that the resulting subversion scheme satisfies perfect decryptability (recovering all messages regularly) is straightforward. It also satisfies the undetectability notion (Definition 13) as the subverted IV is the output of a PRF and it is indistinguishable from a random value. A formal treatment of this statement is given in Theorem 1.

As previously stated, stateful subversion cannot provide strong undetectability. This is easily shown here by observing that the detector could notice that all of the first ciphertexts share the same IV after any reset action.

```
$\widetilde{\mathcal{E}}_{\widetilde{K},K}(M,A,\sigma,i)$
**if** $\sigma = 0$ **then** $IV \leftarrow E_{\widetilde{K}}(K)$
**else** $IV \leftarrow^{\$} \{0,1\}^n$
$C \leftarrow \mathcal{E}_K(M,A;IV)$
$\sigma \leftarrow \sigma + 1$
**return** $(C,\sigma)$
```

```
$\widetilde{\mathcal{D}}_{\widetilde{K}}(\mathbf{C},\mathbf{A},i)$
$IV \leftarrow \mathcal{X}(\mathbf{C}[1])$
$K \leftarrow E^{-1}_{\widetilde{K}}(IV)$
$\mathbf{M} \leftarrow \mathcal{D}_K(\mathbf{C},\mathbf{A})$
**return M**
```

Figure 6: IV-replacement stateful attack.

Theorem 1 ([3])**.** Let $\Pi = (\{0,1\}^n, \mathcal{E}, \mathcal{D})$ be a randomized, stateless symmetric encryption scheme that surfaces an IV of length n. Let $E : \widetilde{K} \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a blockcipher. Let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be the subversion of Π as defined in Figure 6. Let $\mathcal{U}$ be a test that makes q queries to its KEY oracle. Then we can construct an adversary $\mathcal{A}$ that makes q oracle queries and its running time is that of $\mathcal{U}$ such that

$$\mathsf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq q^2/2^n + \mathsf{Adv}^{\mathrm{prf}}_{E}(\mathcal{A})$$

The $q^2/2^n$ term represents the possibility that two users will have the same key, in which case their subverted IVs will be the same. Given this, the subversion $\widetilde{\Pi}$ is undetectable and the detector $\mathcal{U}$'s final advantage is negligible if the blockcipher E behaves as a good PRF.

4.1.2. Stateless Variant

A second variant of the earlier attack leverages the IV-based transport channel to continually transmit the bits of the user key K. This systematic leakage of keying material becomes compatible with a scenario in which subverted encryption cannot use a state. Let k represent the length of the user key K, $v = \lceil \log_2 k \rceil$, and $E : \widetilde{K} \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a blockcipher, where n stands for the length of the IV in Π. A uniformly random index $\ell \in [1..k]$ is chosen by the subverted encryption algorithm $\widetilde{\mathcal{E}}$ (shown in Figure 7); this index will act as the position to the key bit $K[\ell]$ that will be encoded. Here, the IV is picked as the encryption employing the subversion key $\widetilde{K}$ of the message $K[\ell]||\ell||R$, where R is a randomly drawn padding string of size $n-v-1$, ℓ is its position encoded as a v-bit string, and $K[\ell]$ is the leaked bit of K. We expect that every $\ell \in [1..k]$ has been selected after roughly $k\log_2(k)$ encryptions, at which point the entire user key K will be acquired by the subverter. Once again, this subversion scheme fulfills perfect decriptability and the undetectability comes from the fact that E is a PRF, as formally pointed out in Theorem 2.

```
$\widetilde{\mathcal{E}}_{\widetilde{K},K}(M,A,i)$
$\ell \leftarrow^{\$} [1..k]$
$R \leftarrow^{\$} \{0,1\}^{n-v-1}$
$IV \leftarrow E_{\widetilde{K}}(K[l]||l||R)$
$C \leftarrow \mathcal{E}_K(M,A;IV)$
return $C$
```

```
$\widetilde{\mathcal{D}}_{\widetilde{K}}(\mathbf{C},\mathbf{A},i)$
for $j \in 1,\ldots,|C|$ do
    $b||\ell|R \leftarrow E^{-1}_{\widetilde{K}}(\mathcal{X}(\mathbf{C}[j]))$
    $K[\ell] \leftarrow b$
$\mathbf{M} \leftarrow \mathcal{D}_K(\mathbf{C},\mathbf{A})$
return $\mathbf{M}$
```

Figure 7: IV-replacement stateless attack.

Theorem 2 ([3])**.** Let $\Pi = (\{0,1\}^k, \mathcal{E}, \mathcal{D})$ be a randomized, stateless symmetric encryption scheme that surfaces an *IV* of length n. Let $E : \widetilde{K} \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher. Let $v = \lceil \log_2(k) \rceil$. Let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ be the subversion of Π as defined in Figure 7. Let $\mathcal{U}$ be a test that makes q queries to its Enc oracle. Then we can construct an adversary $\mathcal{A}$ that makes q oracle queries and its running time is that of $\mathcal{U}$ such that

$$\mathsf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq q^2/2^{n-v-1} + \mathsf{Adv}^{\mathrm{prf}}_{E}(\mathcal{A})$$

The BPR's paper [3] did not take into account the notion of strong undetectability. Nonetheless, several considerations lead to the conclusion that this scheme also satisfies the stronger variant. The fact that the game SDETECT allows you to inspect and reset the state σ is irrelevant here because both the original and subverted schemes are stateless. Furthermore, the ability to generate keys without using the missing oracle Key shouldn't be beneficial for the detector's task.

4.2. The Biased-Ciphertext Attacks

Many encryption schemes, such as CBC2 [19], IACBC [16], and XCBC$ [12], do not surface their *IV*, which is required for the attacks presented above. To address this issue, BPR [3] introduced a new stateful "universal" *biased-ciphertext attack* that applies to any generic randomized encryptions with few extra requirements. The subsequent improving stateless variant is due to BJK in [2].

4.2.1. Stateful Variant [3]

This attack can be used on any randomized, stateless, and coin-injective encryption scheme. It has been demonstrated that this attack satisfies the undetectability in the BPR model; on the other hand, the authors of the BJK model suggest it can be proven to be strongly undetectable in their model too. The subversion scheme functions intuitively in the following way: consider the encryption of a message M with associated data A and key K as a ciphertext $C \leftarrow \mathcal{E}_K(M,A;\delta)$, where δ is a coin drawn at random from a space D. We refer to it as a random coin to emphasize that it does not represent extractable information, unlike the previous IV. The subversion scheme $\widetilde{\mathcal{E}}$ works similarly to the original $\mathcal{E}$, with the exception that δ is chosen so that $F(\widetilde{K},C) = K[j]$ the j-th bit of K, where F is a PRF. At this point, given C, the key extraction algorithm $\widetilde{\mathcal{D}}$ can compute $K[j]$ using $F(\widetilde{K},C)$. More in detail, the subverted encryption algorithm $\widetilde{\mathcal{E}}$ makes use of an integer state σ, initialized to zero and incremented on each encryption. If j is the position of the key bit $K[j]$ to encode, the procedure randomly selects a coin δ such that the output of $F(\widetilde{K},\mathcal{E}_K(M,A;\delta)||\sigma||i)$

is equal to the $K[j]$. To formalize this, given the coin space $D \subseteq \{0,1\}^*$, a bit $b \in \{0,1\}$, and two functions $g : D \to R$ and $f : \{0,1\}^* \to \{0,1\}$, we consider the set $S^{f,g}(b,D) = \{\delta \in D : f(g(\delta)) = b\}$. The wanted set of coins corresponds to $S^{F(\widetilde{K},\cdot),g(\cdot)}(K[j],D)$ where $F : \widetilde{K} \times \{0,1\}^* \to \{0,1\}$ is a PRF and g is an auxiliary function with embedded parameters (K,M,A,j,σ,i) that, on input the coin δ, outputs $\mathcal{E}_K(M,A;\delta)||\sigma||i$. Finally, the message M is encrypted using a coin δ randomly extracted from such a set. Then, $\widetilde{\mathcal{D}}$ extracts the j-bit of the user key as $F(\widetilde{K},\mathbf{C}[j]||j-1||i)$, where $\mathbf{C}$ is the vector of ciphertexts produced by $\widetilde{\mathcal{E}}$. If $k = |K|$ is the user key length, after k calls to the encryption algorithm, the attacker will be able to extract the user's key and thus decrypt the all the messages. The following theorem formally states the undetectability of the resulting subversion scheme.

```
$\widetilde{\mathcal{E}}_{\widetilde{K},K}(M,A,\sigma,i)$
$j \leftarrow \sigma \bmod |K|; j \leftarrow j+1$
$g(\cdot) \leftarrow \mathcal{E}_K(M,A;\cdot)||\sigma||i$
$\delta \leftarrow^{\$} S^{F(\widetilde{K},\cdot),g(\cdot)}(K[j],D)$
$C \leftarrow \mathcal{E}_K(M,A;\delta)$
$\sigma \leftarrow \sigma+1$
return $(C,\sigma)$

$\widetilde{\mathcal{D}}_{\widetilde{K}}(\mathbf{C},\mathbf{A},i)$
for $j = 1,\ldots,|K|$ do
    $K[j] \leftarrow F(\widetilde{K},\mathbf{C}[j]||j-1||i)$
$\mathbf{M} \leftarrow \mathcal{D}_K(\mathbf{C},\mathbf{A})$
return $\mathbf{M}$
```

Figure 8: The stateful biased-ciphertext attack.

Theorem 3 ([3])**.** Let $\Pi = (\{0,1\}^k,\mathcal{E},\mathcal{D})$ be a randomized, stateless, coin-injective symmetric encryption scheme with randomness-length r, and let $d = 2^r$. Let $F : \widetilde{K} \times \{0,1\}^* \to \{0,1\}$ be a PRF. Let $\widetilde{\Pi} = (\widetilde{\mathcal{K}},\widetilde{\mathcal{E}},\widetilde{\mathcal{D}})$ be the subversion of Π as defined in Figure 8. Let $\mathcal{U}$ be a test that makes q queries to its ENC oracle. Then we can construct an adversary $\mathcal{A}$ that makes q oracle queries and its running time is that of $\mathcal{U}$ such that

$$\mathsf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq q/2^d + \mathsf{Adv}^{\mathrm{prf}}_{E}(\mathcal{A})$$

4.2.2. Stateless Variant [2]

The strong undetectability definition was proposed by BJK in [2], but they also enhanced the earlier biased-ciphertext attack. They can apply their variant to any encryption scheme with a non-trivial level of randomization, eliminating the need for both the coin-injective property and, crucially, the statefulness. This new stateless subversion scheme has been demonstrated to be strongly undetectable in the BJK model and allows a full key recovery. This clearly implies that it is undetectable in the original BPR model too. Let be a PRF $F : \{0,1\}^k \times \{0,1\}^* \to \{0,1\} \times [k]$, where k is the length of the user key. The idea is still to encode the user key bit by bit in the ciphertext using rejection sampling. The algorithm runs a loop, shown in Figure 9, where a random coin $\delta \in \{0,1\}^k$ is chosen at each iteration. This coin is used as randomness for the standard encryption scheme $\mathcal{E}$, which produces a ciphertext C. A pair (v,t) is employed to store the output of the PRF $F(\widetilde{K},C)$ after each attempt. The loop breaks if either an iteration counter j reaches an out-of-time flag s or if the user key's t-th bit matches v. On the other side, the key recovery algorithm $\mathcal{B}.\textsc{Ext}$ works as follows. Given q the number of queries allowed to the attacker, for each i in range $[1,\ldots,q]$, the Big Brother $\mathcal{B}$ invokes the $\textsc{Enc}$ oracle, which returns a ciphertext C of a message chosen by the user. Then, $(v,t) = F(\widetilde{K},C)$ is computed and the t-th bit of the key is set to v. Then, in Theorems 4 and 5 we report the strong detection and key recovery advantages, respectively.

```
$\widetilde{\mathcal{E}}_{\widetilde{K},K}(M,A)$
$j \leftarrow 0$
do
    $j \leftarrow j+1$
    $\delta \leftarrow^{\$} \{0,1\}^k$
    $C \leftarrow \mathcal{E}_K(M,A;\delta)$
    $(v,t) \leftarrow F(\widetilde{K},C)$
while $(K[t] \neq v \wedge j \neq s)$
return $C$
```

```
$\mathcal{B}.\textsc{Ext}_{\widetilde{K}}(\mathbf{C},\mathbf{A},i)$
for $i = 1,\ldots,q$ do
    $C \leftarrow^{\$} \textsc{Enc}()$
    $(v,t) \leftarrow F(\widetilde{K},C)$
    $K[t] \leftarrow v$
return $K$
```

Figure 9: The BJK's stateless variant of the biased-ciphertext attack.

Theorem 4 ([2]). Let Π be a randomized, stateless scheme for symmetric encryption and let k be the user key length. Let $F : \{0,1\}^k \times \{0,1\}^* \to \{0,1\} \times [1,\ldots,k]$ be a PRF. Let $q,s \geq 1$ and let $\widetilde{\Pi}$ the subversion in Figure 9. Let $\mathcal{M}$ be an arbitrary message distribution. Then we can build a PRF adversary $\mathcal{A}$ such that

$$\mathsf{Adv}^{\mathrm{kr}}_{\Pi,\mathcal{M}}(\mathcal{B}) \geq 1 - \mathsf{Adv}^{\mathrm{prf}}_{F}(\mathcal{A}) - \varepsilon(q,s,k)$$

where

$$\varepsilon(q,s,k) \leq ke^{-q/k} + q2^{-s} + q^2 s^2 \cdot 2^{-\mathbf{H}_\infty(\Pi)-1}$$

The running time of $\mathcal{A}$ is about the sum of the running times of $\widetilde{\mathcal{E}}$ and $\mathcal{B}.\textsc{Ext}$, and it makes at most qs oracle queries.

Theorem 5 ([2])**.** Let Π be a randomized, stateless scheme for symmetric encryption and let k the user key length. Let $F : \{0,1\}^k \times \{0,1\}^* \to \{0,1\} \times [1,\ldots,k]$ be a PRF. Let $q,s \geq 1$ and let $\widetilde{\Pi}$ the subversion in Figure 9. Let $\mathcal{U}$ be an adversary against the strong undetectability of $\widetilde{\Pi}$ that makes at most n queries to its Enc oracle. Then we can build a PRF adversary $\mathcal{A}$ such that

$$\mathsf{Adv}^{\mathrm{sdet}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq 2 \cdot \mathsf{Adv}^{\mathrm{prf}}_{F}(\mathcal{A}) + n^2 s^2 \cdot 2^{-\mathbf{H}_\infty(\Pi)}.$$

The running time of $\mathcal{A}$ is about that of $\mathcal{U}$ and it makes at most ns oracle queries.

4.3. The Input-Triggered Attack [10]

This is the generalized version of the input-triggered attack anticipated in Section 3.1.3, and it can be applied to any randomized, deterministic, and/or stateful scheme. In Figure 10, the subverted encryption method $\widetilde{\mathcal{E}}$ always returns the output of the conventional encryption algorithm $\mathcal{E}$, except when a predicate R holds for a specific set of inputs. This triggering predicate indicates when the subversion must leak information to Big Brother using its output. It is plausible that if the detector does not know the predicate R (the triggering condition), it will only be able to distinguish the subversion from the real scheme with a negligible probability. The inputs of the predicate R are $(\widetilde{K}, K, M, A, \sigma, i)$, where i is a numeric identifier of the user and A represents the associated data. The latter field can be utilized to characterize several public secondary information (e.g., IP address, timestamps, network ports,...). The triggering condition could be determined by a number of factors, some of which the adversary may influence. A chosen-message attack like this may appear to be highly powerful and difficult to carry out; however, the scientific literature contains numerous examples of real-world attacks in which the adversary was able to gain such a powerful possibility. Returning to our security models, this capability is present in the surveillance games of the BPR and DFP models, but not in the BJK's KR game. On the other side, as described in Section 3.1.3, the perfect decryptability required by BPR and BJK models does not allow to catch the undetectability of this subversion. The only available characterization is provided by the following theorem, which proves that the subversion is undetectable with respect to the DFP model (Definition 20).

Theorem 6. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a $(1,\delta)$-correct scheme for symmetric encryption. Suppose the message space $\mathcal{M}$ contains $\{0,1\}^\lambda$ for a suitable large λ. Then subversion $\widetilde{\Pi}$ in Figure 10 is $(q, q \cdot 2^{-\lambda} + \delta)$-decryptable, and for all detection tests $\mathcal{U}$ that make at most q encryption queries

$$\mathsf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathcal{U}) \leq q \cdot 2^{-\lambda}$$

```
$\widetilde{\mathcal{E}}_{\widetilde{K},K}(M,A,\sigma,i)$
$(C,\sigma) \leftarrow \mathcal{E}_K(M,A,\sigma)$
if $R(\widetilde{K},K,M,A,\sigma,i)$ then
    return $(C||K,\sigma)$
return $(C,\sigma)$
```

Figure 10: The input-triggered attack.

It is noteworthy that, despite being applicable to any encryption scheme, this attack is always detectable in the DFP model. Indeed, the detector has access to the transcript of the Big Brother, which, in order to win the surveillance game, has to exploit the trigger-based mechanism introducing in the transcript an anomalous ciphertext that permits the detector to distinguish with a non-negligible probability.

5. Defenses

The results of the previous sections do not give us much hope. Theorems 4 and 5 demonstrate that any stateless randomized symmetric encryption scheme can be subverted without the user's knowledge. This suggests that anyone could successfully subvert many of the encryption schemes we use on a daily basis. The question of whether there are classes of encryption algorithms that are resistant to these kinds of attacks is intriguing from a theoretical perspective. We must undoubtedly search among deterministic and stateful algorithms in light of the prior findings. The first class of immune algorithms, the so-called *Unique-Ciphertext* encryption schemes, was identified by BPR.

5.1. Unique-Ciphertext Encryption Schemes

Intuitively, if a symmetric encryption scheme satisfies the unique-ciphertext property, it means that for any given key, message, associated data, and state, the decryptor will accept and decrypt only one ciphertext to the specific message. This requirement implies that a unique-ciphertext scheme is also deterministic. The reverse is not true: not all deterministic encryption systems support the unique-ciphertext property. Consider a deterministic regular encryption $\mathcal{E}$, like plain AES, and an artificial extension $\mathcal{E}'$, which systematically extends the ciphertext with a fixed-length null tail of bits. The corresponding decryption utility $\mathcal{D}'$ will remove the extra tail without checking its content, and the task will be completed with the regular decryption $\mathcal{D}$. This scheme is clearly deterministic, but it fails to provide unique-ciphertext property because there exist multiple ciphertexts that can be opened on the same message. The formal definition of the unique-ciphertext encryption scheme, as well as the statement of the subversion impossibility theorem, are provided below.

Definition 23 (Unique-Ciphertext Encryption Scheme)**.** A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to have unique ciphertexts (or it is said a unique-ciphertext encryption scheme) if the following conditions hold:

1. Π satisfies perfect correctness
2. For all $\ell \in \mathbb{N}$, all $K \in \mathcal{K}$, all $\mathbf{M} \in \mathcal{M}^\ell$ and all $\mathbf{A} \in \mathcal{AD}^\ell$, there exists exactly one ciphertext vector $\mathbf{C}$ such that

$$(\mathbf{M}, \sigma_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon) \text{ for some } \sigma_\ell.$$

A concrete example of a symmetric encryption scheme based on the encode-then-encipher technique [4], satisfying the unique-ciphertext property, is provided by BPR in [3]. In the same paper, the authors provided a second nonce-based encryption scheme [18, 19] that satisfies the same property but in a more efficient way. For more details on these constructions, we refer to the original paper.

Theorem 7 ([3])**.** Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a unique-ciphertext encryption scheme. Let $\widetilde{\Pi}$ be a subversion of Π that obeys the perfect decryptability condition relative to Π. Let $\mathcal{B}$ an adversary. Then $\mathsf{Adv}^{\mathrm{srv}}_{\Pi,\widetilde{\Pi}}\mathcal{B} = 0$.

This result holds in the DFP security model, as demonstrated in [10]. We omit the theorem for the sake of brevity.

5.2. Other Defensive Techniques

In recent years, additional security models have been developed to provide unsubvertible IND-CPA secure encryption schemes. Unfortunately, unlike the works discussed here, they do not fall under the complete subversion model (i.e., a scenario in which all parts or algorithms could be subverted), so their results are weaker than those reported here.

For instance, [17] proposed a new model called *cryptographic reverse firewalls* in which a third party, called *firewall*, stays between the communicating parties involved in a cryptographic protocol. These firewalls [1, 7–9, 17] keep the communication between the parts safe by re-randomizing the messages and stopping any leaks that could be caused by subversion. The main limitation of these firewalls is that they must be considered totally trustworthy, so not subvertible.

Fischlin and Mazaheri [11] proposed the notion of *self-guarding scheme*. In this model, they provide an alternative defensive mechanism for reverse firewalls that doesn't rely on trusted third parties and proactively thwarts ASAs. The focus here is on a scenario where the party has a genuine version of the algorithm as before it is replaced by malicious software or a time bomb, triggering the algorithm's malicious behavior. To prevent leakage, the proposed schemes rely on data gathered during the initial phase, when the scheme is deemed unsubvertible.

Finally, Russel *et al.* [21] introduced an IND-CPA secure encryption scheme that cannot be subverted by decomposing the involved algorithms into a few functional components which are tested by an offline watchdog. The main difference with the models presented in our survey is the fact that the detector has access to a series of oracles, each of which represents a functional component of the encryption algorithm.

6. Conclusions

We should consider a wide range of potential threats to our digital security in light of the subversion attacks shown here. Certainties, such as the reliability of the software we use, are not always assured and may raise serious concerns about what is truly safe. We are still a long way from figuring out the best security paradigm and, consequently, what specific countermeasures might be. However, we think that this kind of research can advance the scientific community's comprehension of the problem's numerous facets. It can also help cryptographic system users realize that the security offered has limitations.

Acknowledgments. The University of Catania funded this work as part of the "PIACERI 2024-2026" project.

REFERENCES

[1] Giuseppe Ateniese, Bernardo Magri, and Daniele Venturi. Subversion-resilient signature schemes. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 364–375, Denver, CO, USA, October 12–16, 2015. ACM Press.

[2] Mihir Bellare, Joseph Jaeger, and Daniel Kane. Mass-surveillance without the state: Strongly undetectable algorithm-substitution attacks. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 1431–1440, Denver, CO, USA, October 12–16, 2015. ACM Press.

[3] Mihir Bellare, Kenneth G. Paterson, and Phillip Rogaway. Security of symmetric encryption against mass surveillance. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 1–19, Santa Barbara, CA, USA, August 17–21, 2014. Springer, Berlin, Heidelberg, Germany.

[4] Mihir Bellare and Phillip Rogaway. Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography. In Tatsuaki Okamoto, editor, *ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 317–330, Kyoto, Japan, December 3–7, 2000. Springer, Berlin, Heidelberg, Germany.

[5] Sebastian Berndt and Maciej Liskiewicz. Algorithm substitution attacks from a steganographic perspective. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017*, pages 1649–1660, Dallas, TX, USA, October 31 – November 2, 2017. ACM Press.

[6] Sebastian Berndt, Jan Wichelmann, Claudius Pott, Tim-Henrik Traving, and Thomas Eisenbarth. ASAP: Algorithm substitution attacks on cryptographic protocols. In Yuji Suga, Kouichi Sakurai, Xuhua Ding, and Kazue Sako, editors, *ASIACCS 22*, pages 712–726, Nagasaki, Japan, May 30 – June 3, 2022. ACM Press.

[7] Suvradip Chakraborty, Stefan Dziembowski, and Jesper Buus Nielsen. Reverse firewalls for actively secure MPCs. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part II*, volume 12171 of *LNCS*, pages 732–762, Santa Barbara, CA, USA, August 17–21, 2020. Springer, Cham, Switzerland.

[8] Suvradip Chakraborty, Bernardo Magri, Jesper Buus Nielsen, and Daniele Venturi. Universally composable subversion-resilient cryptography. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part I*, volume 13275 of *LNCS*, pages 272–302, Trondheim, Norway, May 30 – June 3, 2022. Springer, Cham, Switzerland.

[9] Rongmao Chen, Yi Mu, Guomin Yang, Willy Susilo, Fuchun Guo, and Mingwu Zhang. Cryptographic reverse firewall via malleable smooth projective hash functions. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 844–876, Hanoi, Vietnam, December 4–8, 2016. Springer, Berlin, Heidelberg, Germany.

[10] Jean Paul Degabriele, Pooya Farshim, and Bertram Poettering. A more cautious approach to security against mass surveillance. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 579–598, Istanbul, Turkey, March 8–11, 2015. Springer, Berlin, Heidelberg, Germany.

[11] Marc Fischlin and Sogol Mazaheri. Self-guarding cryptographic protocols against algorithm substitution attacks. In Steve Chong and Stephanie Delaune, editors, *CSF 2018 Computer Security Foundations Symposium*, pages 76–90, Oxford, UK, July 9–12, 2018. IEEE Computer Society Press.

[12] Virgil D. Gligor and Pompiliu Donescu. Fast encryption and authentication: XCBC encryption and XECB authentication modes. In Mitsuru Matsui, editor, *FSE 2001*, volume 2355 of *LNCS*, pages 92–108, Yokohama, Japan, April 2–4, 2002. Springer, Berlin, Heidelberg, Germany.

[13] Eu-Jin Goh, Dan Boneh, Benny Pinkas, and Philippe Golle. The design and implementation of protocol-based hidden key recovery. In *ISC*, volume 2851 of *Lecture Notes in Computer Science*, pages 165–179. Springer, 2003.

[14] Glenn Greenwald. *No Place to Hide: Edward Snowden, the NSA and the Surveillance State*. Penguin Books Limited, 2014.

[15] Julian Borger James Ball and Glenn Greenwald. Revealed: how US and UK spy agencies defeat internet privacy and security. *The Guardian*, 2013.

[16] Charanjit S. Jutla. Encryption modes with almost free message integrity. *Journal of Cryptology*, 21(4):547–578, October 2008.

[17] Ilya Mironov and Noah Stephens-Davidowitz. Cryptographic reverse firewalls. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 657–686, Sofia, Bulgaria, April 26–30, 2015. Springer, Berlin, Heidelberg, Germany.

[18] Phillip Rogaway. Authenticated-encryption with associated-data. In Vijayalakshmi Atluri, editor, *ACM CCS 2002*, pages 98–107, Washington, DC, USA, November 18–22, 2002. ACM Press.

[19] Phillip Rogaway. Nonce-based symmetric encryption. In Bimal K. Roy and Willi Meier, editors, *FSE 2004*, volume 3017 of *LNCS*, pages 348–359, New Delhi, India, February 5–7, 2004. Springer, Berlin, Heidelberg, Germany.

[20] Alexander Russell, Qiang Tang, Moti Yung, and Hong-Sheng Zhou. Cliptography: Clipping the power of kleptographic attacks. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 34–64, Hanoi, Vietnam, December 4–8, 2016. Springer, Berlin, Heidelberg, Germany.

[21] Alexander Russell, Qiang Tang, Moti Yung, and Hong-Sheng Zhou. Generic semantic security against a kleptographic adversary. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017*, pages 907–922, Dallas, TX, USA, October 31 – November 2, 2017. ACM Press.

[22] Daniel Shurmow and Niels Ferguson. On the possibility of a back door in the NIST SP800-90 dual EC PRNG. CRYPTO Rump Session, 2007.

[23] Gustavus J. Simmons. The prisoners' problem and the subliminal channel. In David Chaum, editor, *CRYPTO'83*, pages 51–67, Santa Barbara, CA, USA, 1983. Plenum Press, New York, USA.

[24] Gustavus J. Simmons. Authentication theory/coding theory. In G. R. Blakley and David Chaum, editors, *CRYPTO'84*, volume 196 of *LNCS*, pages 411–431, Santa Barbara, CA, USA, August 19–23, 1984. Springer, Berlin, Heidelberg, Germany.

[25] Gustavus J. Simmons. A secure subliminal channel (?). In Hugh C. Williams, editor, *CRYPTO'85*, volume 218 of *LNCS*, pages 33–41, Santa Barbara, CA, USA, August 18–22, 1986. Springer, Berlin, Heidelberg, Germany.

[26] Yi Wang, Rongmao Chen, Xinyi Huang, and Moti Yung. Sender-anamorphic encryption reformulated: Achieving robust and generic constructions. In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part VI*, volume 14443 of *LNCS*, pages 135–167, Guangzhou, China, December 4–8, 2023. Springer, Singapore, Singapore.

[27] Adam Young and Moti Yung. The dark side of "black-box" cryptography, or: Should we trust capstone? In Neal Koblitz, editor, *CRYPTO'96*, volume 1109 of *LNCS*, pages 89–103, Santa Barbara, CA, USA, August 18–22, 1996. Springer, Berlin, Heidelberg, Germany.

[28] Adam Young and Moti Yung. Kleptography: Using cryptography against cryptography. In Walter Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 62–74, Konstanz, Germany, May 11–15, 1997. Springer, Berlin, Heidelberg, Germany.

[29] Adam Young and Moti Yung. The prevalence of kleptographic attacks on discrete-log based cryptosystems. In Burton S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 264–276, Santa Barbara, CA, USA, August 17–21, 1997. Springer, Berlin, Heidelberg, Germany.

[30] Adam Young and Moti Yung. Bandwidth-optimal kleptographic attacks. In Çetin Kaya Koç, David Naccache, and Christof Paar, editors, *CHES 2001*, volume 2162 of *LNCS*, pages 235–250, Paris, France, May 14–16, 2001. Springer, Berlin, Heidelberg, Germany.

[31] Adam Young and Moti Yung. A space efficient backdoor in RSA and its applications. In Bart Preneel and Stafford Tavares, editors, *SAC 2005*, volume 3897 of *LNCS*, pages 128–143, Kingston, Ontario, Canada, August 11–12, 2006. Springer, Berlin, Heidelberg, Germany.

[32] Adam L. Young and Moti Yung. Space-efficient kleptography without random oracles. In *Information Hiding*, volume 4567 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2007.

D. CARNEMOLLA
Department of Mathematics and Computer Science
University of Catania
Viale Andrea Doria 6, Catania, Italy
e-mail: `davide.carnemolla@phd.unict.it`

M. DI RAIMONDO
Department of Mathematics and Computer Science
University of Catania
Viale Andrea Doria 6, Catania, Italy
e-mail: `mario.diraimondo@unict.it`