A More Cautious Approach to Security Against Mass Surveillance

Jean Paul Degabriele[1], Pooya Farshim[2], and Bertram Poettering[3]

[1] Royal Holloway, University of London, United Kingdom
[2] Queen's University Belfast, United Kingdom
[3] Ruhr University Bochum, Germany

Abstract. At CRYPTO 2014 Bellare, Paterson, and Rogaway (BPR) presented a formal treatment of symmetric encryption in the light of algorithm-substitution attacks (ASAs), which may be employed by 'big brother' entities for the scope of mass surveillance. Roughly speaking, in ASAs big brother may bias ciphertexts to establish a covert channel to leak vital cryptographic information. In this work, we identify a seemingly benign assumption implicit in BPR's treatment and argue that it artificially (and severely) limits big brother's capabilities. We then demonstrate the critical role that this assumption plays by showing that even a slight weakening of it renders the security notion completely unsatisfiable by any, possibly deterministic and/or stateful, symmetric encryption scheme. We propose a refined security model to address this shortcoming, and use it to restore the positive result of BPR, but caution that this defense does not stop most other forms of covert-channel attacks.

Keywords. Mass surveillance, algorithm-substitution attack, symmetric encryption, covert channel.

1 Introduction

In 2013 Edward Snowden shocked the world with revelations of several ongoing surveillance programs targeting citizens worldwide [2,11]. There is now incontestable evidence that national intelligence agencies can go to great lengths to undermine our privacy. The methods employed to attack and infiltrate our communication infrastructure are rather disturbing. Amongst others these include sabotaging Internet routers, wiretapping international undersea cables, installing backdoors in management front ends of telecom providers, injecting malware in real time into network packets carrying executable files, and intercepting postal shipping to replace networking hardware.

Some of the revelations concern the domain of cryptography. Somewhat reassuringly, there was no indication that any of the well-established cryptographic primitives and hardness assumptions could be broken by the national intelligence agencies. Instead these agencies resorted to more devious means in order to compromise the security of cryptographic protocols. In one particular instance the National Security Agency (NSA) maneuvered cryptographic standardization bodies to recommend a cryptographic primitive which contained a backdoor [14]: The specification of the `Dual_EC_DRBG` cryptographic random-number generator [3] contains arbitrary-looking parameters for which there exists trapdoor information, known to its creators, that can be used to predict future results from a sufficiently long stretch of output [17]. A recent study [6] explores the practicality of exploiting this vulnerability in TLS. In particular, it shows that support of the Extended Random TLS extension [15] (an IETF draft co-authored by an NSA employee) makes the vulnerability much easier to exploit. Furthermore the NSA is known to have made secret payments to vendors in order to include the `Dual_EC_DRBG` in their products and increase proliferation [12].

Such tactics clearly fall outside of the threat models that we normally assume in cryptography and call for a reconsideration of our most basic assumptions. It is hence natural to ask what other means could be employed by such powerful entities to subvert cryptographic protocols. Recent

work by Bellare, Paterson and Rogaway [5] explores the possibility of mass surveillance through *algorithm-substitution attacks* (ASA). Consider some type of closed-source software making use of a standard symmetric encryption scheme to achieve its security goals. In an ASA the standard encryption scheme is substituted with an alternative scheme that the attacker has authored; we call this latter scheme a *subversion*. A successful ASA would allow the adversary, henceforth referred to as *big brother*, to undermine the confidentiality of the data and at the same time circumvent *detection* by its users.

BPR's treatment. Bellare, Paterson and Rogaway (BPR) [5] define a formal framework for analyzing resistance to a certain class of ASAs in the context of symmetric encryption. At a very high level, their notion of surveillance resistance requires that big brother be incapable of distinguishing ciphertext produced by the legitimate scheme from those produced by the subverted scheme. They also put forward a notion of undetectability, that can be seen as a dual of the former notion, which guarantees that no efficient detection algorithm is capable of distinguishing legitimate ciphertext from those produced by the subverted scheme. An attack that is undetectable would therefore be a particularly damaging one, and indeed BPR show strong forms of undetectable subversions in [5]. Moreover, this notion is only used to prove *negative* results in that work.

BPR are able to establish a set of positive and negative results within their framework. They build on the work of [10] to demonstrate ASAs on specific schemes such as the CTR$ and CBC$ modes of operation. Their negative results culminate with the *biased-ciphertext attack* which can be mounted against any randomized symmetric encryption scheme that uses a sufficient amount of randomness. This attack establishes a *covert channel* between the subverted encryption algorithm and big brother, through which he is able to retrieve the full user key. Furthermore, the biased-ciphertext attack is shown to be undetectable. This ultimately leads to the conclusion that no probabilistic encryption scheme can resist ASAs. Consequently, BPR turn to (stateful) *deterministic* encryption schemes and identify a combinatorial property that is sufficient to ensure security within their model. Most modern encryption schemes are nonce-based [16] and satisfy this property.

Contributions. In this work we revisit the security model proposed by BPR [5] and re-examine its underlying assumptions. Our main criticism concerns the notion of *perfect decryptability*, and the requirement that *every subversion* must satisfy it. Decryptability is introduced as a minimal requirement that a subversion must meet in order to have some chance of avoiding detection. Accordingly, the assumption is that big brother would only consider subversions that satisfy this condition. We argue, however, that this requirement is stronger than what is substantiated by this rationale, and it results in artificially limiting big brother's set of available strategies. Indeed, we show that with a minimal relaxation of the decryptability condition the BPR security notion becomes totally unsatisfiable. More precisely, for *any* symmetric encryption scheme, deterministic or not, we construct a corresponding undetectable subversion that can be triggered to leak information when run on specific inputs known solely by big brother. We call this an *input-triggered* attack. From a theoretical perspective this shows that the instantiability of the security model crucially depends on the strictness of the decryptability requirement. From a more practical perspective, security in the BPR model simply does not translate to security in practice.

As pointed out in [5], defending against ASAs requires the ability to detect them. Indeed, the ability to detect an ASA is an important measure of security which should be surfaced by any security definition. We observe that in this respect the BPR security definition falls short of providing an adequate formalization: encryption schemes are considered secure against ASAs as

long as subversions can be detected with *non-zero* probability. A scheme guaranteeing a detection probability of say 2^{-128} does not give any assurance of practical value, but in the BPR model it is deemed secure as long as the subversion is perfectly decryptable.

Building on the work of Bellare, Paterson and Rogaway [5] we propose an alternative security definition to address the above shortcomings. Our model disposes of the perfect decryptability requirement and instead quantifies security via a new detectability notion. In particular we require that a scheme come with a corresponding detection algorithm. The detection game starts by running big brother as in the BPR surveillance game and a transcript of his queries is stored. The detection algorithm is then given the user key and the transcript and must determine whether a subversion has taken place.

Although in our security definition the detector runs after big brother, the implemented detection strategy need not necessarily be after the fact. The detector could be run each time a new ciphertext is computed. As long as no anomaly is detected the ciphertext will be transmitted. Such a monitoring detection strategy appears to be necessary since static detection strategies (as considered in [5]) are not effective against the input-triggered subversions discussed above. We define security by requiring that for any subversion the detector's advantage must be quantitatively comparable to big brother's surveillance advantage. Put differently, when a subversion is effective it must be also detectable, and when a subversion is not detectable it must be ineffective. We then re-establish the relative strength of deterministic encryption schemes in comparison to randomized ones, as suggested in [5], in our security model.

Limitations of the security model. In this paper we reconsider BPR's security model and propose a refinement of it. Accordingly, we try to deviate as least as possible from the setting and assumptions considered there. However, this is not to say that the new and old models are devoid of other potential shortcomings. Bellare, Paterson and Rogaway state very clearly the restricted scope of their analysis, and naturally these restrictions are inherited in our analysis as well. In particular, in both their and our models information can only be leaked to big brother through ciphertexts, and other forms of covert channels, for instance based on time or power analysis, are not addressed. Similarly we only look at symmetric encryption, whereas real-world protocols often employ other cryptographic primitives as well. Thus protecting symmetric encryption from subversion (in our sense) does not guarantee that the security protocols in which they are used are protected against subversion (in a broader sense). Nonetheless, we believe that BPR's and our work are important first steps towards a better understanding of the problems relating to mass surveillance and the limitations in protecting against it.

Other related work. The first systematic analysis of how malicious modification of implemented cryptosystems can weaken their expected security dates back to Simmons [18]. He studied how cryptographic algorithms in black-box implementations can be made to leak information about secret keying material via *subliminal channels*. In the setting considered by Simmons, anyone who successfully reverse-engineers the manipulated code would also be able to recover the leaked secrets.

Similar concepts were put forward by Young and Yung in a sequence of works [19,20,21,22,23,24] under the label of *Kleptography*, focusing mainly on primitives in the realm of public-key cryptography (encryption and signature schemes based on RSA and DLP). In their proposals an essential part of any subverted protocol algorithm would be the public key of the subverter; all leakage originating from the subversion would be 'safely encrypted' to this key. The idea is then that if a subverted implementation is successfully reverse-engineered by a user and the existence of a back-

door is revealed, the intended security of the overall system does not collapse, as the attacker's secret key would be held 'responsibly' (by, say, a governmental agency). Thus, the settings of ASAs (as considered in our work) and of Kleptography are different: while the primary aim of the former is to subvert undetectably, the latter is more focused on graceful security degradation. Kleptographic attacks on RSA systems were also reported by Crépeau and Slakmon [7] who optimized the efficiency of subverted key-generation algorithms by using symmetric techniques. Concerning higher-level protocols, algorithm-substitution attacks targeting specifically the SSL/TLS and SSH protocols were reported by Goh et al. [10], and Young and Yung [25].

Following BPR's work, a number of articles analyse the subvertability of other cryptographic primitives, for instance random number generators [9] or signature schemes [1].

2 Preliminaries

NOTATION. Unless otherwise stated, an algorithm may be randomized. An adversary is an algorithm. For any algorithm $\mathscr{A}$, $y \leftarrow \mathscr{A}(x_1, x_2, \dots)$ denotes executing $\mathscr{A}$ with fresh coins on inputs $x_1, x_2, \dots$ and assigning its output to y. For n, a positive integer, we use $\{0,1\}^n$ to denote the set of all binary strings of length n and $\{0,1\}^*$ to denote the set of all finite binary strings. The empty string is represented by ε. For any two strings x and y, $x \parallel y$ denotes their concatenation and $|x|$ denotes the length of x. For any vector $\mathbf{X}$, we denote by $\mathbf{X}[i]$ its ith component. If $\mathcal{S}$ is a finite set then $|\mathcal{S}|$ denotes its size, and $y \leftarrow\!\!\$\ \mathcal{S}$ denotes the process of selecting an element from $\mathcal{S}$ uniformly at random and assigning it to y. $\Pr[\, P : E \,]$ denotes the probability of event E occurring after having executed process P. Security definitions are formulated through the code-based game-playing framework.

SYMMETRIC ENCRYPTION. A *symmetric encryption scheme* is a triple of algorithms $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. Associated to Π are the message space $\mathcal{M} \subseteq \{0,1\}^*$ and the associated data space $\mathcal{AD} \subseteq \{0,1\}^*$. The *key space* $\mathcal{K}$ is a non-empty set of strings of some fixed length. The *encryption algorithm* $\mathcal{E}$ may be randomized, stateful, or both. It takes as input the secret key $K \in \mathcal{K}$, a message $M \in \{0,1\}^*$, an associated data $A \in \{0,1\}^*$, and the current encryption state σ to return a ciphertext C or the special symbol $\perp$, together with an updated state. The symbol $\perp$ may be returned for instance if $M \notin \mathcal{M}$ or $A \notin \mathcal{AD}$. The *decryption algorithm* $\mathcal{D}$ is deterministic but may be stateful. It takes as input the secret key K, a ciphertext string $C \in \{0,1\}^*$, an associated data string $A \in \{0,1\}^*$, and the current decryption state ϱ to return the corresponding message M or the special symbol $\perp$, and an updated state. Pairs of ciphertext and associated data that result in $\mathcal{D}$ outputting $\perp$ are called *invalid*.

The encryption and decryption states are always initialized to ε. We say that $\mathcal{E}$ (resp., $\mathcal{D}$) is a stateless algorithm if for all inputs in $\mathcal{K} \times \{0,1\}^* \times \{0,1\}^* \times \{\varepsilon\}$ the updated state remains ε. The scheme Π is said to be stateless if both $\mathcal{E}$ and $\mathcal{D}$ are stateless. We also require that for any $M \in \mathcal{M}$ and any $A \in \mathcal{AD}$ it holds that $\{0,1\}^{|M|} \subseteq \mathcal{M}$ and $\{0,1\}^{|A|} \subseteq \mathcal{AD}$.

For any symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, any $\ell \in \mathbb{N}$, any vector $\mathbf{M} = [M_1, \dots, M_\ell] \in \mathcal{M}^\ell$ and any vector $\mathbf{A} = [A_1, \dots, A_\ell] \in \mathcal{AD}^\ell$, we write $(\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon)$ as shorthand for

$$(C_1, \sigma_1) \leftarrow \mathcal{E}_K(M_1, A_1, \varepsilon);\ \dots;\ (C_\ell, \sigma_\ell) \leftarrow \mathcal{E}_K(M_\ell, A_\ell, \sigma_{\ell-1})\ ,$$

where $\mathbf{C} = [C_1, \dots, C_\ell]$. Similarly we write $(\mathbf{M}', \varrho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon)$ to denote the analogous process for decryption.

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

Fig. 1: Game defining the IND-CPA security of scheme Π against $\mathscr{A}$.

Definition 1 (Correctness [5]). *A symmetric encryption scheme* Π *is said to be* (q, δ)-correct *if for all* $\ell \leq q$*, all* $\mathbf{M} \in \mathcal{M}^\ell$ *and all* $\mathbf{A} \in \mathcal{AD}^\ell$*,*

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

Schemes that achieve correctness with $\delta = 0$ *for all* $q \in \mathbb{N}$ *are said to be* perfectly correct*.*

We now recall the standard IND-CPA security notion for symmetric encryption [4].

Definition 2 (Privacy). *Let* $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ *be a symmetric encryption scheme and let* $\mathscr{A}$ *be an adversary. Consider the game* $\text{IND-CPA}^{\mathscr{A}}_{\Pi}$ *depicted in Figure 1. The adversary's advantage is defined as*

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

Informally, when the above advantage is sufficiently small for every efficient adversary $\mathscr{A}$ we say that scheme Π is IND-CPA secure.

3 Algorithm-Substitution Attacks

In an algorithm-substitution attack (ASA), big brother is able to covertly replace the code of an encryption algorithm $\mathcal{E}$ with a subverted encryption algorithm $\widetilde{\mathcal{E}}$. Here, $\widetilde{\mathcal{E}}$ additionally to the inputs of $\mathcal{E}$ takes a subversion key $\widetilde{K}$. This key is assumed to be embedded in $\widetilde{\mathcal{E}}$'s code in an obfuscated manner and hence inaccessible to users. Intuitively, the subversion key significantly improves big brother's ability to leak information via the ciphertexts without being detected. For instance, it can use $\widetilde{K}$ to encrypt a user's key and use the result as a random-looking IV in the ciphertext. Big brother can later intercept this ciphertext, recover the user's key from the IV, and use it to decrypt the rest of the ciphertexts. In addition, we allow the operations of $\widetilde{\mathcal{E}}$ to depend on user-specific identification data i.

Note that when considering ASAs the concern is not about whether the real encryption scheme contains a backdoor, possibly due to an obscurely generated set of parameters. In fact an inherent assumption proposed in [5] and here is that the real encryption scheme $\mathcal{E}$ achieves the required level of security and in particular is free from backdoors. Instead, the question being asked is whether an *implementation* of the real scheme, possibly obfuscated, contains a backdoor and under what circumstances this can be detected.

Subversions. For any symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ a subversion of Π is a pair $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}})$ as follows. The *subversion key space* $\widetilde{\mathcal{K}}$ is a finite non-empty set. The *subverted encryption algorithm* $\widetilde{\mathcal{E}}$ may be randomized, stateful, or both. It takes as 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 $A \in \{0,1\}^*$, an encryption

state σ, and a user identifier i, to return a ciphertext C or the special symbol $\perp$, together with an updated state. The user identifier is considered a public value such as an IP or MAC address unique to each key K; the precise value of i does not matter and is merely used to label users.

According to [5], a subversion additionally consists of a *plaintext-recovery algorithm* $\widetilde{\mathcal{D}}$ that allows the subverter to recover information from ciphertexts. However, the security notions in [5] and here do not refer to this algorithm (and in fact the functionality offered by $\widetilde{\mathcal{D}}$ can be assumed to be implicit in big brother). Therefore, as in [5] we omit $\widetilde{\mathcal{D}}$ from the syntax of subversions.

3.1 The BPR model

Bellare, Paterson and Rogaway (BPR) [5] formulate security against ASAs as follows. In addition to the standard security requirements for symmetric encryption they consider two new notions, namely *undetectability* and *surveillance* defined respectively using the DETECT and SURV games shown in Figure 2.

In DETECT an algorithm $\mathscr{U}$, representing a detection test, is required to distinguish $\mathcal{E}$ from $\widetilde{\mathcal{E}}$ when given oracle access to one of these two algorithms. More specifically, a bit b and a subversion key $\widetilde{K}$ are first sampled and $\mathscr{U}$ is then given access to two oracles, KEY and ENC. The game models a multi-user setting and the KEY oracle serves to let user i create a secret key. The ENC oracle takes a message M, associated data A, and a user identifier i and depending on the value of b it returns an encryption under either $\mathcal{E}$ or $\widetilde{\mathcal{E}}$. The game ends when $\mathscr{U}$ halts and outputs a bit b' as its guess of bit b. The corresponding advantage is

$$\mathbf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathscr{U}) := 2 \cdot \Pr\left[\, \mathrm{DETECT}^{\mathscr{U}}_{\Pi,\widetilde{\Pi}} \,\right] - 1\,.$$

In SURV an adversary $\mathscr{B}$, who does not have access to the users' secret keys but knows the subversion key, is required to distinguish $\mathcal{E}$ from $\widetilde{\mathcal{E}}$ when given oracle access to one of these algorithms. The game proceeds by first sampling a bit b and a subversion key $\widetilde{K}$, and then $\mathscr{B}$ is given access to $\widetilde{K}$ and two oracles, KEY and ENC. Oracle KEY only serves to initialize a secret key for specified user i and does not return any value. The ENC oracle takes a message M, associated data A, and a user identifier i, and depending on the value of b it returns an encryption under either $\mathcal{E}$ or $\widetilde{\mathcal{E}}$. The game ends when $\mathscr{B}$ halts and outputs a bit b' as its guess of bit b. The corresponding advantage is defined as:

$$\mathbf{Adv}^{\mathrm{srv}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) := 2 \cdot \Pr\left[\, \mathrm{SURV}^{\mathscr{B}}_{\Pi,\widetilde{\Pi}} \,\right] - 1\,.$$

In addition to the above two notions, BPR specify the following *decryptability* condition.

Definition 3 (Decryptability). *A subversion $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}})$ is said to satisfy (q, δ)-decryptability with respect to the scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ if symmetric encryption scheme $(\widetilde{\mathcal{K}} \times \mathcal{K}, \widetilde{\mathcal{E}}, \mathcal{D}')$ where $\mathcal{D}'_{\widetilde{K},K}(C, A, \varrho) := \mathcal{D}_K(C, A, \varrho)$ is (q, δ)-correct (for all choices of inputs i to $\widetilde{\mathcal{E}}$).*

If $\widetilde{\Pi}$ is $(q, 0)$-decryptable with respect to Π for all $q \in \mathbb{N}$, it is said to be perfectly decryptable. We highlight that BPR requires any subversion to satisfies perfect decryptability. For reasons that will become apparent later we chose to distinguish between (q, δ)-decryptability and perfect decryptability. However BPR do not make this distinction and use the term decryptability to mean perfect decryptability.

Note that the DETECT game is formulated from big brother's point of view who wants his subversion to remain undetected. The notion it yields is that of *undetectability*, and in [5] it is used

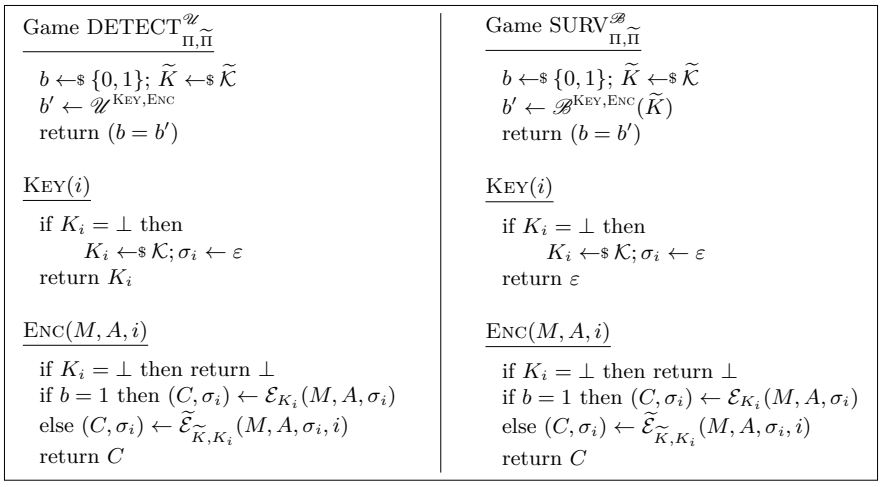

Fig. 2: The DETECT and SURV games defining the BPR security model [5].

only for proving *negative* results. For instance BPR use this to show that any randomized encryption scheme can be subverted in an undetectable manner. Concretely, for any randomized scheme Π that uses sufficient amount of randomness, there exists a perfectly decryptable subversion $\widetilde{\Pi}$ such that for all efficient detection tests $\mathscr{U}$ the advantage $\mathbf{Adv}^{\text{det}}_{\Pi,\widetilde{\Pi}}(\mathscr{U})$ is small. Moreover, the subversion $\widetilde{\Pi}$ allows big brother to completely recover the user's key K with overwhelming probability.

Security against surveillance is defined through the SURV game. The requirement here is that big brother, who knows the subversion key $\widetilde{K}$, is unable to tell whether ciphertexts are being produced by the real encryption algorithm $\mathcal{E}$ or the subverted encryption algorithm $\widetilde{\mathcal{E}}$. This implicitly ensures that if the real scheme is IND-CPA secure then the subverted scheme still does not reveal to big brother anything about the plaintext. (See Appendix A for a formalization and extension.) Clearly, without any further restriction on $\widetilde{\Pi}$ surveillance resilience is not attainable, since for any scheme Π there always exists a trivial subversion $\widetilde{\Pi}$ and an adversary $\mathscr{B}$ which can distinguish the two. (Consider for example the subversion which appends a redundant zero bit to the ciphertexts.) Hence some resistance to detection should hold simultaneously. This is imposed by means of the *decryptability* condition in [5]. More formally, an encryption scheme Π is said to be surveillance secure if for all subversions $\widetilde{\Pi}$ that are perfectly decryptable with respect to Π and all adversaries $\mathscr{B}$ with reasonable resources, the advantage $\mathbf{Adv}^{\text{srv}}_{\Pi,\widetilde{\Pi}}(\mathscr{B})$ is small. We emphasize that the DETECT game does *not* play a role in this definition.

3.2 Critique of the BPR model

Although decryptability is formulated as a correctness requirement in [5], its use is closer to that of *undetectability*. More precisely, it is understood to be the 'weakest notion' of undetectability that big brother should aim for, and whose failure would certainly lead to his subversion being discovered. BPR write [5, page 6]

> This represents the most basic form of resistance to detection, and we will assume any subversion must meet it.

On the other hand the undetectability notion associated to the DETECT game is meant to be a much stronger one. Another excerpt reads [5, page 7]

Algorithm $\widetilde{\mathcal{E}}_{\widetilde{K},K}(M, A, \sigma, i)$

$(C, \sigma) \leftarrow \mathcal{E}_K(M, A, \sigma)$
if $\mathbf{R}(\widetilde{K}, K, M, A, \sigma, i) = \mathsf{true}$ then
  return $(C \parallel K, \sigma)$
else return (C, σ)

Fig. 3: The encryption algorithm of the subversion $\widetilde{\Pi}$ used in Theorem 1.

A subversion $\widetilde{\Pi}$ in which this advantage [that is, $\mathbf{Adv}^{\mathrm{det}}_{\Pi,\widetilde{\Pi}}(\mathscr{U})$] is negligible for all practical tests $\mathscr{U}$ is said to be *undetectable* and would be one that evades detection in a powerful way. If such a subversion permitted plaintext recovery, big brother would consider it a very successful one.

This all seems to imply that for any subversion, decryptability is a necessary requirement to avoid detection, and that undetectability is sufficient to yield a strong guarantee of avoiding detection. It is hence natural to expect that undetectability implies decryptability, but as the authors of [5] admit this is not the case. The two notions are in fact incomparable. This is a source of inconsistency, especially when considering that the negative and positive results in [5] are established using measures of undetectability that are incomparable.

The main reason for this discord between decryptability and undetectability is that undetectability allows detection test $\mathscr{U}$ to succeed with negligible probability, whereas (perfect) decryptability requires the test's success probability to be exactly zero. This is unnecessarily strict, as detection tests which succeed only with negligible probability are insignificant and pose no effective threat to big brother. Accordingly, it is unrealistic to assume that big brother will only produce subversions that satisfy *perfect* decryptability. To summarize, requiring the latter imposes an unnatural restriction on big brother's potential subversion strategies, thereby unjustifiably weakening the security notion.

It would seem that both of the above issues could be easily addressed (at least in part) by letting decryptability admit a small negligible error, that is requiring (q, δ)-decryptability for a small δ instead of decryptability with $\delta = 0$. In particular, one could hope that decryptability would then be implied by undetectability. Unfortunately the situation is not that simple. As we demonstrate next, this minor alteration would render the BPR security notion *unsatisfiable* by any symmetric encryption scheme with a reasonably large message space. More specifically, for any symmetric encryption scheme we can construct a subversion that not only is (q, δ)-decryptable (with negligible δ for any reasonable value q) but is in fact undetectable, and yet there always exists an adversary $\mathscr{B}$ capable of subverting the scheme. This serves to show that the BPR security definition crucially relies on the presupposition that all subversions must satisfy perfect decryptability, and is consequently a rather fragile security definition.

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

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

Moreover, there exist efficient adversaries $\mathscr{A}$ and $\mathscr{B}$ such that

$$\mathbf{Adv}^{\text{srv}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) \geq 1 - (\mathbf{Adv}^{\text{ind-cpa}}_{\Pi}(\mathscr{A}) + \delta + 2^{-\lambda})\,.$$

Proof. The subversion $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}})$ is defined by letting $\widetilde{\mathcal{K}} := \{0,1\}^\lambda$ and $\widetilde{\mathcal{E}}$ be the algorithm depicted in Figure 3. The predicate $\mathbf{R}(\widetilde{K}, K, M, A, \sigma, i)$ that is used in $\widetilde{\mathcal{E}}$ takes the Boolean value true for all tuples where $\widetilde{K} = M$ and the value false otherwise. Note that for all inputs where $\widetilde{K} \neq M$ the subverted encryption algorithm $\widetilde{\mathcal{E}}$ behaves exactly like the real encryption algorithm $\mathcal{E}$. Let E denote the event that for some $1 \leq j \leq \ell$ it holds that $\widetilde{K} = \mathbf{M}[j]$. Then for all $1 \leq \ell \leq q$ and all message vectors $\mathbf{M} \in \mathcal{M}^\ell$ we have

$$\begin{aligned}
&\Pr\left[(\widetilde{K}, K) \leftarrow\!\!\$\ \widetilde{\mathcal{K}} \times \mathcal{K}; (\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon); (\mathbf{M}', \varrho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon) : \mathbf{M} \neq \mathbf{M}'\right] \\
&\quad \leq \Pr\left[(\widetilde{K}, K) \leftarrow\!\!\$\ \widetilde{\mathcal{K}} \times \mathcal{K}; (\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon); (\mathbf{M}', \varrho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon)\ :\ E\right] \\
&\qquad + \Pr\left[(\widetilde{K}, K) \leftarrow\!\!\$\ \widetilde{\mathcal{K}} \times \mathcal{K}; (\mathbf{C}, \sigma_\ell) \leftarrow \mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon); (\mathbf{M}', \varrho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon)\ :\ \mathbf{M} \neq \mathbf{M}' \mid \overline{E}\right] \\
&\quad \leq q \cdot 2^{-\lambda} + \delta\,,
\end{aligned}$$

where the bound on the second term follows from the (q, δ)-correctness of Π. Hence $\widetilde{\Pi}$ satisfies $(q, q \cdot 2^{-\lambda} + \delta)$-decryptability with respect to Π. Since $\widetilde{K}$ remains information theoretically hidden from $\mathscr{U}$ during its run, for any (even computationally unbounded) detection test $\mathscr{U}$ making at most q queries $\mathbf{Adv}^{\text{det}}_{\Pi,\widetilde{\Pi}}(\mathscr{U})$ is bounded above by $q \cdot 2^{-\lambda}$.

The adversary $\mathscr{B}$, which knows the subversion key, simply queries the pair $(\widetilde{K}, A)$ to its $\textsc{Enc}$ oracle for some $A \in \mathcal{AD}$, and gets in return a ciphertext C^*. It then attempts to parse C^* as $C \parallel K$ and checks whether $\widetilde{K} = \mathcal{D}_K(C, A, \varepsilon)$. If this test succeeds it outputs 0 and otherwise it outputs 1. Note that when the encryption oracle is instantiated with the subversion ($b = 0$), the adversary guesses $b' = 0$ with probability $1 - \delta$ by the correctness of Π. When the oracle is instantiated with the real scheme ($b = 1$), we upper-bound $\mathscr{B}$'s success probability by $\mathbf{Adv}^{\text{ind-cpa}}_{\Pi}(\mathscr{A}) + 2^{-\lambda}$ for some adversary $\mathscr{A}$. Letting b' denote $\mathscr{B}$'s output we get that

$$\mathbf{Adv}^{\text{srv}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) \geq (1 - \delta) - (\mathbf{Adv}^{\text{ind-cpa}}_{\Pi}(\mathscr{A}) + 2^{-\lambda})\,,$$

as desired.

To establish the claimed bounded above we construct an IND-CPA adversary $\mathscr{A}$ against Π using $\mathscr{B}$ as follows. Adversary $\mathscr{A}$ starts by picking a subversion key $\widetilde{K}$ uniformly at random and runs $\mathscr{B}(\widetilde{K})$ and answers its encryption query (M_0, A), where $M_0 = \widetilde{K}$, as follows. It samples a random message M_1 of length $|M_0|$, submits (M_0, M_1, A) to its LR oracle, and forwards the ciphertext C^* that it receives to $\mathscr{B}$. At this point $\mathscr{B}$ halts and $\mathscr{A}$ outputs whatever $\mathscr{B}$ outputs, which we denote by b'. Let d denote the bit in the IND-CPA game indicating which message is being encrypted. Then

$$\mathbf{Adv}^{\text{ind-cpa}}_{\Pi}(\mathscr{A}) = \Pr\left[\, b' = 0 \mid d = 0 \,\right] - \Pr\left[\, b' = 0 \mid d = 1 \,\right]\,. \tag{1}$$

Now when $d = 0$, the ciphertext C^* is an encryption of (M_0, A) and $\mathscr{B}$ is run in the SURV environment with bit b set to 1:

$$\Pr\left[\, b' = 0 \mid d = 0 \,\right] = \Pr\left[\, b' = 0 \mid b = 1 \,\right]\,. \tag{2}$$

On the other hand when $d = 1$ the ciphertext C^* encrypts a random message that is independent of $\widetilde{K} = M_0$. Hence $\mathscr{B}$ cannot do better than guessing $\widetilde{K}$:

$$\Pr\left[\, b' = 0 \mid d = 1 \,\right] \leq 2^{-\lambda}\,. \tag{3}$$

Combining Equations (1), (2) and (3) we get the desired bound, i.e.,

$$\Pr\left[\, b' = 0 \mid b = 1 \,\right] \leq \mathbf{Adv}^{\text{ind-cpa}}_{\Pi}(\mathscr{A}) + 2^{-\lambda}\,.$$

□

Input-triggered subversions. We emphasize that the above subversion generically applies to any symmetric encryption scheme, irrespective of it being probabilistic, deterministic, or stateful. Additionally, while we present the subversion of Figure 3 merely as a component of Theorem 1, it actually embodies a powerful subversion strategy for mounting ASAs that are hard to detect.[4] The underlying principle is that a subversion leaks information to big brother only when receiving specific inputs. That is, in order for big brother to exploit the subversion and undermine the privacy of the communication, a trigger needs to be set. On the other hand, without the knowledge of this trigger it is practically impossible to distinguish the subversion from the real scheme. In our case the trigger is the set of inputs for which the predicate $\mathbf{R}$ holds. In practice, $\mathbf{R}$ can depend on any information that the subverted encryption algorithm may have access to, such as an IP address, a username, or some location information. Such information, in particular network addresses and routing information, can be readily available in the associated data. It is not unreasonable, and is in fact in conformance with the usual approach adopted in cryptography, to assume that big brother may be capable of influencing this information when it needs to intercept a communication. We hence see no basis for excluding such attacks from consideration.

Security guarantees. BPR start from the premise that surveillance security is not possible without requiring some form of resistance to detection, and they address this by requiring that all subversions satisfy perfect decryptability. Indeed, it seems that the only way to protect against ASAs is to have a mechanism that detects such attacks. Accordingly, an encryption scheme should be deemed surveillance secure if we have a sufficiently good chance of detecting subversions of that scheme. However, the BPR security notion gives only a very weak guarantee of detecting ASAs. More specifically, for a secure scheme we are only guaranteed to be able to detect a subversion with non-zero probability, regardless of how small that may be. In particular, if for a specific scheme there exist subversions that can all be detected with non-zero but negligible probability, but nevertheless is perfectly correct, in the BPR's model the scheme is considered subversion secure. It should be evident that such a scheme offers no significant resistance to subversion in practice.

Another shortcoming of relying on decryptability as a means of detection is that it does not clearly state what tests one ought to do in order to detect a subversion. Decryption failures may happen for other reasons, and if they occur sporadically they may easily go unnoticed. Moreover, it may not suffice to rely on the decryption algorithm at the receiver's end. For instance, if ciphertexts contain additional information that big brother can exploit but which would result in a decryption failure, big brother could rectify this at the point of interception after having recovered the information he needs. Alternatively, big brother may have replaced the decryption algorithm

[4] This is akin to a trapdoor. It is a classic technique in computer security to introduce trapdoors in various objects and we certainly do not claim to be the first to do so.

with one that can handle ciphertexts from the subverted encryption algorithm without raising any exceptions. While for an open system like TLS [8] it may be reasonable to assume that big brother is unable to mount an ASA on all of its implementations, for a closed system there is no reason to assume that big brother is not able to substitute both the encryption and decryption algorithms.[5]

4 The Proposed Security Model

The analysis of Section 3.2 leaves us with an unsatisfactory state of affairs. On the one hand we wish for a more realistic security model, devoid of the perfect decryptability condition. On the other hand we saw that this would allow input-triggered subversions which are generically applicable to any symmetric encryption scheme. This in turn raises the question of whether we have any hope at all of protecting against ASAs. We address this question by proposing an alternative security model, which builds on the work of Bellare, Paterson and Rogaway [5].

Our premise is that input-triggered subversions cannot be detected with significant probability through a one-time test, as in the DETECT game. Instead, it seems that the best we can hope for is to detect whether the encryption algorithm is leaking information during a communication session. That is, we don't aim at determining whether or not the encryption algorithm has been substituted, since without knowledge of the trigger we have very little chance of detecting this. However we may be able to detect whether big brother is exploiting an existing subversion and is able to gather information from it, which is what we really care about.

In formulating security, we consider all possible subversions that big brother may come up with, without imposing any additional restrictions that a subversion must satisfy. Specifically, we identify a scheme to be subversion resistant if for all of its possible subversions, either the subversion leaks no information to big brother, or if it does leak information then this is detectable with high probability. We formalize this by means of a new pair of games, $\overline{\text{DETECT}}$ and $\overline{\text{SURV}}$. The game $\overline{\text{SURV}}$ is a single-user version of the SURV game from [5], and can be shown to be equivalent to it, through a standard hybrid argument, up to a factor equal to the number of users. The new game serves to specify formally what we intuitively referred to as 'leaking information to big brother'. The $\overline{\text{DETECT}}$ game, on the other hand, differs substantially form the DETECT game of the BPR security model. Most importantly, it is intended for specifying a notion of *detectability* rather than *un*detectability. In $\overline{\text{DETECT}}$, the detection test $\mathscr{U}$ does not get access to an encryption oracle; instead it only gets a transcript of $\mathscr{B}$'s queries to its own oracle. We will then quantify the effectiveness of the detection test $\mathscr{U}$ by comparing its success in the $\overline{\text{DETECT}}$ game to that of $\mathscr{B}$ in the $\overline{\text{SURV}}$ game.

The surveillance game starts by picking a bit b uniformly at random, and then generates the keys K and $\widetilde{K}$. Big brother is then given access to the subversion key and an encryption oracle, but not to the key K. Depending on the value of b, the encryption oracle will either return encryptions under scheme Π and the user's key K or encryptions under the subverted scheme (which has access to both keys). The adversary outputs a bit b' as its guess of the challenge bit b. See Figure 4 (right) for the details. The detection game augments the surveillance game as follows. First $\mathscr{B}$ is run in the same manner as in the surveillance game and a transcript T of its encryption queries is formed. The detection algorithm $\mathscr{U}$ is then given access to this transcript and the user's key. Its goal is to output a bit b'' as its guess of the challenge bit b. See Figure 4 for a formal description of both

[5] For example, this could be for some proprietary application/protocol that uses a standard (non-proprietary) encryption scheme, but for which there exists only one implementation.

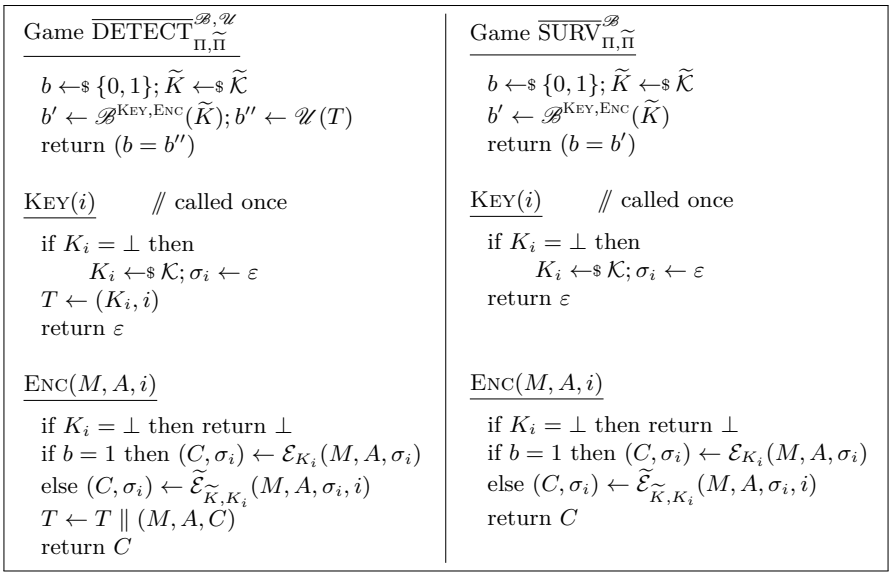

Fig. 4: Games defining the refined single-user security models. Big brother $\mathscr{B}$ can only call the KEY oracle once.

security games. Note that in the $\overline{\text{DETECT}}$ game $\mathscr{B}$'s output is discarded as it's role in this game is only to generate the transcript of queries.

We now move on to define security in terms of the above games. For each of these games we define the corresponding advantages in the usual manner. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and let $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}})$ be a subversion of it. For an adversary $\mathscr{B}$ its surveillance advantage is given by

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

Similarly, the detection advantage of algorithm $\mathscr{U}$ with respect to $\mathscr{B}$ is given by

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

We will require that a secure scheme Π come with a detection test $\mathscr{U}$ which will allow the detection of information leakage from a subverted encryption algorithm. Intuitively our security definition should guarantee that for any adversary $\mathscr{B}$ that wins the $\overline{\text{SURV}}$ game with a significant advantage ϵ, the detection test $\mathscr{U}$ will win the $\overline{\text{DETECT}}$ game with a correspondingly significant advantage δ. Thus, as our first attempt at defining subversion resilience we may require that for all adversaries $\mathscr{B}$ and all subversions $\widetilde{\Pi}$,

$$\mathbf{Adv}^{\overline{\text{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) \le \delta \implies \mathbf{Adv}^{\overline{\text{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) \le \epsilon\ .$$

For $\delta, \epsilon \in [0, 1]$ let us say $(\Pi, \mathscr{U})$ is (δ, ϵ)-subversion-resistant if it satisfies the above condition for all $\mathscr{B}$ and $\widetilde{\Pi}$. The following statements hold.

(i) For any $\delta' \le \delta$ and $\epsilon' \ge \epsilon$, a (δ, ϵ)-subversion-resistant pair $(\Pi, \mathscr{U})$ is also (δ', ϵ')-subversion-resistant.
(ii) No pair $(\Pi, \mathscr{U})$ is (δ, ϵ)-subversion-resistant if $\delta > \epsilon$.

The first property follows trivially from the definition. For the second, assume that $(\Pi, \mathscr{U})$ is (δ, ϵ)-subversion-resistant with $\delta > \epsilon$. We show that there exists a subversion $\widetilde{\Pi}$ and an adversary $\mathscr{B}$ such that

$$\mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) \leq \delta \quad \text{but} \quad \mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) > \epsilon\,,$$

thereby contradicting (δ, ϵ)-subversion-resistance. To this end, let $\widetilde{\mathcal{K}}$ be such that $|\widetilde{\mathcal{K}}| \geq (\delta - \epsilon)^{-1}$, where the latter quantity is, by assumption, positive. Define the subverted encryption algorithm $\widetilde{\mathcal{E}}$ as follows. Partition the combined key space $\mathcal{K} \times \widetilde{\mathcal{K}}$ into two sets where $\lfloor \delta \cdot |\mathcal{K} \times \widetilde{\mathcal{K}}| \rfloor$ of the key pairs belong to the first set and the rest to the second. For key pairs in the first set, algorithm $\widetilde{\mathcal{E}}$ is defined to return the error symbol $\perp$. For those in the second set it returns unmodified encryptions under $\mathcal{E}$. Adversary $\mathscr{B}$ asks for an encryption of a fixed message and outputs 0 if it receives $\perp$; otherwise it outputs 1. Clearly the detection advantage is at most δ for this subversion. The surveillance advantage, on the other hand, is

$$\frac{\lfloor \delta \cdot |\mathcal{K}| \cdot |\widetilde{\mathcal{K}}| \rfloor}{|\mathcal{K}| \cdot |\widetilde{\mathcal{K}}|} > \delta - \frac{1}{|\widetilde{\mathcal{K}}|} \geq \epsilon\,,$$

where the last inequality follows from the lower bound imposed on $|\widetilde{\mathcal{K}}|$.

While the notion of (δ, ϵ)-subversion-resistance seems reasonable and in line with the style of concrete security, by itself it does not yield a satisfactory security definition. This is due to the fact that when the surveillance advantage is less than ϵ we cannot conclude anything about the success of the detection test. Furthermore if the surveillance advantage is substantially greater than ϵ for some other subversion,the detection guarantees still remain at δ, which could be much smaller than big brother's advantage. Finally, note that ϵ can be increased or decreased to any desired value using a subversion akin to that discussed above.

In essence (δ, ϵ)-subversion-resistance conveys information about a single point over the range of all possible values in $[0, 1]^2$. A more uniform treatment would be to define security by *relating* the detection and surveillance advantages via a function ρ. Formally, we say that the pair $(\Pi, \mathscr{U})$ is ρ-subversion-resistant if for all adversaries $\mathscr{B}$ and all subversions $\widetilde{\Pi}$,

$$\mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) \geq \rho\left(\mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B})\right)\,.$$

It can be shown using property (i) that any such function ρ must be monotonically increasing. Moreover, property (ii) says that any function ρ is dominated by the identity function. Hence *id*-subversion-resistant is the best security we can hope for. As we shall see in the next section, this level form of subversion resistance is achievable and we will adopt it as our proposed security definition.

Definition 4 (Subversion resistance). *A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to be subversion resistant with respect to a (universal) detection algorithm $\mathscr{U}$ if all efficient adversaries $\mathscr{B}$ and all efficient subversions $\widetilde{\Pi}$,*

$$\mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) \leq \mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U})\,.$$

DEFINITIONAL CHOICES. Observe that our surveillance game is identical to the single-user version of BPR's original surveillance game in Figure 2.[6] In particular, it allows big brother to launch $\widetilde{K}$-dependent chosen-plaintext attacks. Our detection game is also single-user and this reflects the fact that users do not need to run a *coordinated* detection procedure. Detection requires the existence of a strong *universal* detector that depends neither on the subverted algorithm nor on big brother. This is in contrast to BPR's formulation where detection was used for negative results and non-universal multi-user detectors were allowed to demonstrate the strength of BPR's attacks. For detection, as in BPR, we assume explicit knowledge of user keys but do not allow access to the (possibly subverted) encryption procedure or the internal state/randomness of the scheme. Restricting the resources of the detector only strengthens our positive results. On the other hand, the communicated ciphertexts/messages should be made available to the detector via the transcript. As we have seen, without this strengthening, resistance against input-triggered subversions is impossible, even for multi-user, oracle-assisted detectors. We note that our actual detection procedure in Section 5 processes ciphertexts one at a time and hence storing only the last computed ciphertext would also be sufficient.

5 Subversion Resistance from Unique Ciphertexts

We have not yet determined whether there exist symmetric encryption schemes which satisfy our security definition. In [5] the authors describe a powerful generic attack, termed the *biased-ciphertext attack*, that can be applied to any probabilistic symmetric encryption scheme. Hence any scheme that resists subversion must be deterministic. Bellare, Paterson, and Rogaway identified the *unique ciphertexts* property for symmetric encryption schemes as sufficient to satisfy their notion of surveillance security. We now show that this property is strong enough to also guarantee subversion security in sense of Definition 4. Let us first recall the definition of unique ciphertexts from [5].

Definition 5 (Unique ciphertexts). *A symmetric encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is said to have unique ciphertexts if the following conditions hold.*

(i) Π *satisfies perfect correctness.*

(ii) 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

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

It follows from Definition 5 that any symmetric encryption scheme that has unique ciphertexts must be deterministic. Note on the other hand that a deterministic encryption scheme does not necessarily have unique ciphertexts. In [5] it is shown how stateful encryption schemes having unique ciphertexts are easily obtained from most nonce-based encryption schemes [16] which are known to satisfy the tidiness property of [13]. The following theorem says that for schemes with unique ciphertexts we are guaranteed to always detect a subversion with the tightest possible success rate.

Theorem 2. *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme with unique ciphertexts and let $\mathscr{U}$ be the detection test in Figure 5. Then for all subversions $\widetilde{\Pi}$ and all adversaries $\mathscr{B}$ we have that*

$$\mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) \leq \mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) \,.$$

[6] The single-user and multi-user games can be shown equivalent via a standard hybrid argument [5]. Since our detection procedure is also in the single-user setting, we have adopted a single-user surveillance game as well. This choice also translates to a more faithful comparison of concrete advantage terms.

Algorithm $\mathscr{U}(T)$

Parse T as $(K, i) \parallel T'$
$j \leftarrow 1; \mathbf{M} \leftarrow [\,]; \mathbf{A} \leftarrow [\,]; \mathbf{C} \leftarrow [\,]$
for each (M, A, C) in T' do
  $\mathbf{M}[j] \leftarrow M, \mathbf{A}[j] \leftarrow A; \mathbf{C}[j] \leftarrow C$
  $j \leftarrow j + 1$
$(\mathbf{M}', \varrho_\ell) \leftarrow \mathcal{D}_K(\mathbf{C}, \mathbf{A}, \varepsilon)$
return $(\mathbf{M}' = \mathbf{M})$

Fig. 5: The detection test $\mathscr{U}$ used in Theorem 2.

Proof. Fix a subversion $\widetilde{\Pi} = (\widetilde{\mathcal{K}}, \widetilde{\mathcal{E}}, \widetilde{\mathcal{D}})$ and an adversary $\mathscr{B}$. Define

Event E: algorithm $\mathscr{B}$ makes a sequence of queries $(\mathbf{M}, \mathbf{A})$ such that the real and subverted encryption algorithms output a different ciphertext sequence, i.e., $\mathcal{E}_K(\mathbf{M}, \mathbf{A}, \varepsilon) \neq \widetilde{\mathcal{E}}_{\widetilde{K},K}(\mathbf{M}, \mathbf{A}, \varepsilon, i)$.

Then for any key K, any subversion key $\widetilde{K}$, any subversion $\widetilde{\Pi}$ and any adversary $\mathscr{B}$ the corresponding surveillance advantage can be expressed as

$$\mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) = 2 \cdot \Pr\left[\, \overline{\mathrm{SURV}}^{\mathscr{B}}_{\Pi,\widetilde{\Pi}} \mid E \,\right] \Pr[\, E \,] + 2 \cdot \Pr\left[\, \overline{\mathrm{SURV}}^{\mathscr{B}}_{\Pi,\widetilde{\Pi}} \mid \overline{E} \,\right] \Pr\left[\, \overline{E} \,\right] - 1\,,$$

where the probabilities are calculated over the coins of $\mathscr{B}$, the coins of $\widetilde{\mathcal{E}}$, the sampling of the two keys, and bit b. Now if E does *not* occur, $\mathscr{B}$ has no information about the bit b in the $\overline{\mathrm{SURV}}$ game and hence $\Pr\left[\, \overline{\mathrm{SURV}}^{\mathscr{B}}_{\Pi,\widetilde{\Pi}} \mid \overline{E} \,\right] = 1/2$. We many continue as follows.

$$\begin{aligned}\mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) &= 2 \cdot \Pr\left[\, \overline{\mathrm{SURV}}^{\mathscr{B}}_{\Pi,\widetilde{\Pi}} \mid E \,\right] \Pr[\, E \,] + \Pr\left[\, \overline{E} \,\right] - 1 \\ &\leq 2 \cdot \Pr[\, E \,] - \Pr[\, E \,] = \Pr[\, E \,]\ .\end{aligned}$$

We can expand the detection advantage of $\mathscr{U}$ with respect to $\mathscr{B}$ in a similar manner:

$$\mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) = 2 \cdot \Pr\left[\, \overline{\mathrm{DETECT}}^{\mathscr{B},\mathscr{U}}_{\Pi,\widetilde{\Pi}} \mid E \,\right] \cdot \Pr[\, E \,] + 2 \cdot \Pr\left[\, \overline{\mathrm{DETECT}}^{\mathscr{B},\mathscr{U}}_{\Pi,\widetilde{\Pi}} \mid \overline{E} \,\right] \cdot \Pr\left[\, \overline{E} \,\right] - 1\,.$$

As before, if E does not occur $\mathscr{U}$ has no information about the bit b in the $\overline{\mathrm{DETECT}}$ game and cannot do better than guessing. Moreover, when E occurs, it follows from the construction of $\mathscr{U}$ (see Figure 5) and the fact that Π has unique ciphertexts that $\mathscr{U}$ can always distinguish the real scheme from a subversion. Thus $\Pr\left[\, \overline{\mathrm{DETECT}}^{\mathscr{B},\mathscr{U}}_{\Pi,\widetilde{\Pi}} \mid \overline{E} \,\right] = 1/2$ and $\Pr\left[\, \overline{\mathrm{DETECT}}^{\mathscr{B},\mathscr{U}}_{\Pi,\widetilde{\Pi}} \mid E \,\right] = 1$. This gives

$$\mathbf{Adv}^{\overline{\mathrm{det}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}, \mathscr{U}) = \Pr[\, E \,] \geq \mathbf{Adv}^{\overline{\mathrm{srv}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B})\,,$$

as desired. □

We remark that an alternative detection procedure that uses $\mathcal{E}$ to check the consistency of ciphertexts in the transcript can be defined. The analogous theorem statement would then state that any *deterministic* scheme is subversion resistant with respect to this detection procedure.

6 Concluding Remarks

Through this work we unravelled definitional challenges in modeling resistance against algorithm-substitution attacks, and in the process we proposed a refinement to address some of the shortcomings of the recent model by Bellare, Paterson, and Rogaway. Within the new model we are able to re-establish that deploying ciphertext-unique encryption schemes can provide a provable (but limited) degree of resistance against adversarial entries who carry out ASAs. In practice, there are many more avenues for big brother to undermine the security of real-world cryptosystems than those considered by BPR and here. Characterizing when it is possible to resist against mass surveillance using cryptographic techniques (even in principle) and when this lies beyond their reach is, in our opinion, an important area of research that needs further investigation.

Acknowledgments. The authors would like to thank Daniel J. Bernstein for his comments on the earlier versions of this paper. Degabriele and Poettering were supported by EPSRC Leadership Fellowship EP/H005455/1. Poettering was also supported by a Sofja Kovalevskaja Award of the Alexander von Humboldt Foundation, and the German Federal Ministry for Education and Research.

References

1. Giuseppe Ateniese, Bernardo Magri, and Daniele Venturi. Subversion-resilient signature schemes. Cryptology ePrint Archive, Report 2015/517, 2015. `http://eprint.iacr.org/2015/517` (to appear on the ACM Conference on Computer and Communications Security — CCS 2015).
2. James Ball, Julian Borger, and Glenn Greenwald. Revealed: how US and UK spy agencies defeat internet privacy and security. *The Guardian*, Sep 2013. `http://www.theguardian.com/world/2013/sep/05/nsa-gchq-encryption-codes-security`.
3. Elaine Barker and John Kelsey. Recommendation for random number generation using deterministic random bit generators, Jan 2012. `http://csrc.nist.gov/publications/nistpubs/800-90A/SP800-90A.pdf`.
4. Mihir Bellare, Anand Desai, Eric Jokipii, and Phillip Rogaway. A concrete security treatment of symmetric encryption. In *38th FOCS*, pages 394–403, Miami Beach, Florida, October 19–22, 1997. IEEE Computer Society Press.
5. 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, Germany.
6. Stephen Checkoway, Ruben Niederhagen, Adam Everspaugh, Matthew Green, Tanja Lange, Thomas Ristenpart, Daniel J. Bernstein, Jake Maskiewicz, Hovav Shacham, and Matthew Fredrikson. On the practical exploitability of dual EC in TLS implementations. In Kevin Fu and Jaeyeon Jung, editors, *Proceedings of the 23rd USENIX Security Symposium, San Diego, CA, USA, August 20-22, 2014.*, pages 319–335. USENIX Association, 2014.
7. Claude Crépeau and Alain Slakmon. Simple backdoors for RSA key generation. In Marc Joye, editor, *CT-RSA 2003*, volume 2612 of *LNCS*, pages 403–416, San Francisco, CA, USA, April 13–17, 2003. Springer, Berlin, Germany.
8. Tim Dierks and Eric Rescorla. The Transport Layer Security (TLS) Protocol version 1.2. RFC 5246, August 2008. `https://www.ietf.org/rfc/rfc5246.txt`.
9. Yevgeniy Dodis, Chaya Ganesh, Alexander Golovnev, Ari Juels, and Thomas Ristenpart. A formal treatment of backdoored pseudorandom generators. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 101–126, Sofia, Bulgaria, April 26–30, 2015. Springer, Berlin, Germany.
10. Eu-Jin Goh, Dan Boneh, Benny Pinkas, and Philippe Golle. The design and implementation of protocol-based hidden key recovery. In Colin Boyd and Wenbo Mao, editors, *ISC 2003*, volume 2851 of *LNCS*, pages 165–179, Bristol, UK, October 1–3, 2003. Springer, Berlin, Germany.
11. Glenn Greenwald. *No Place to Hide: Edward Snowden, the NSA and the Surveillance State*. Penguin Books Limited, 2014.

12. Joseph Menn. Exclusive: Secret contract tied NSA and security industry pioneer. *Reuters*, Dec 2013. http://www.reuters.com/article/2013/12/20/us-usa-security-rsa-idUSBRE9BJ1C220131220.
13. Chanathip Namprempre, Phillip Rogaway, and Thomas Shrimpton. Reconsidering generic composition. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 257–274, Copenhagen, Denmark, May 11–15, 2014. Springer, Berlin, Germany.
14. Nicole Perlroth. Government announces steps to restore confidence on encryption standards. *The New York Times*, Sep 2013. http://bits.blogs.nytimes.com/2013/09/10/government-announces-steps-to-restore-confidence-on-encryption-standards/.
15. Eric Rescorla and Margaret Salter. Extended random values for TLS. Internet Draft, March 2009. https://tools.ietf.org/html/draft-rescorla-tls-extended-random-02.
16. 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, Germany.
17. Daniel Shurmow and Niels Ferguson. On the possibility of a back door in the NIST SP800-90 dual EC PRNG. CRYPTO Rump Session, 2007. http://rump2007.cr.yp.to/15-shumow.pdf.
18. 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.
19. 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, Germany.
20. 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, Germany.
21. 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, Germany.
22. 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, Germany.
23. Adam Young and Moti Yung. Malicious cryptography: Kleptographic aspects (invited talk). In Alfred Menezes, editor, *CT-RSA 2005*, volume 3376 of *LNCS*, pages 7–18, San Francisco, CA, USA, February 14–18, 2005. Springer, Berlin, Germany.
24. 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, Germany.
25. Adam L. Young and Moti Yung. Space-efficient kleptography without random oracles. In Teddy Furon, François Cayre, Gwenaël J. Doërr, and Patrick Bas, editors, *Information Hiding, 9th International Workshop, IH 2007, Saint Malo, France, June 11-13, 2007*, volume 4567 of *LNCS*, pages 112–129. Springer, 2007.

A Generic Security Preservation

Undetectability does not necessarily mean that big brother is unable to subvert the encryption routine, but allows the big brother to do this as long as the subversion is undetectable. This raises the question if standard security requirements from the scheme such as IND-CPA security remain intact when a system is operated with respect to the (undetectable) subverted algorithm, even with respect to adversaries that know the subversion key. A positive answer would ensure, for example, that the confidentially of plaintexts is not affected under undetectable subversion. In this section we confirm this and show that a broad class of security properties are preserved in any undetectable subversion.

Generic IND games. The class of games that we consider here correspond to those that have black-box access to the encryption procedure with respect to a random *unknown* key. (In particular, the code of this game, including its return statement, does not depend on this key.) For instance, the

standard IND-CPA and IND$-CPA games satisfy these requirements.[7] Formally, we call an efficient oracle machine $\mathrm{Game}^{\mathcal{E}}$ a *generic IND game* if $\mathrm{Game}^{\mathcal{E}}$ on input the description of an efficient machine $\mathscr{A}$ returns a Boolean value and has advantage definition

$$\mathbf{Adv}^{\mathrm{Game}}_{\Pi}(\mathscr{A}) := 2 \cdot \Pr\left[\mathrm{Game}^{\mathcal{E}_K(\cdot)}(\mathscr{A})\right] - 1\,,$$

where the probability is taken over the choice of $K \leftarrow\!\$\, \mathcal{K}$, the coins of the game and those of $\mathscr{A}$. We define security with respect to a subversion via

$$\mathbf{Adv}^{\mathrm{Game}}_{\widetilde{\Pi}}(\mathscr{A}) := 2 \cdot \Pr\left[\mathrm{Game}^{\widetilde{\mathcal{E}}_{K,\widetilde{K}}(\cdot)}(\mathscr{A}(\widetilde{K}))\right] - 1\,,$$

where the probability is taken over $(K, \widetilde{K}) \leftarrow\!\$\, \mathcal{K} \times \widetilde{\mathcal{K}}$ the coins of the game and those of $\mathscr{A}$.

Our generic security-preservation theorem is as follows.

Theorem 3 (Security preservation). *Let* Game *be a generic IND game. Then for any efficient adversary $\mathscr{A}$ there are efficient adversaries $\mathscr{A}'$ and $\mathscr{B}$ such that*

$$\mathbf{Adv}^{\mathrm{Game}}_{\widetilde{\Pi}}(\mathscr{A}) = \mathbf{Adv}^{\mathrm{Game}}_{\Pi}(\mathscr{A}') + 2 \cdot \mathbf{Adv}^{\overline{\mathrm{SURV}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B})\ .$$

In particular, if Π is surveillance secure with respect to $\mathscr{U}$ there are efficient adversaries $\mathscr{A}'$ and $\mathscr{B}$ such that

$$\mathbf{Adv}^{\mathrm{Game}}_{\widetilde{\Pi}}(\mathscr{A}) \le \mathbf{Adv}^{\mathrm{Game}}_{\Pi}(\mathscr{A}') + 2 \cdot \mathbf{Adv}^{\overline{\mathrm{DETECT}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B},\mathscr{U})\ .$$

The proof views the composition of the game and the adversary as a surveillance adversary. Then by surveillance security $\mathrm{Game}^{\mathcal{E}}(\mathscr{A}(\widetilde{K})) \approx \mathrm{Game}^{\widetilde{\mathcal{E}}}(\mathscr{A}(\widetilde{K}))$, which means the subverted scheme is secure if the original scheme is. Note that for this argument we need that Game is efficient (aka. falsifiable) and does not have access to the key K. We give the details next.

Proof. Consider big brother $\mathscr{B}(\widetilde{K})$ that runs $\mathrm{Game}(\mathscr{A}(\widetilde{K}))$ and answers the game's oracle queries using its own encryption oracle. When this game terminates and returns a bit, big brother flips it and returns it as b'. We have

$$\begin{aligned} \Pr\left[b' = 0 | b = 0\right] &= \Pr\left[\mathrm{Game}^{\widetilde{\mathcal{E}}}(\mathscr{A}(\widetilde{K}))\right] \\ \text{and} \quad \Pr\left[b' = 0 | b = 1\right] &= \Pr\left[\mathrm{Game}^{\mathcal{E}}(\mathscr{A}(\widetilde{K}))\right]\ . \end{aligned}$$

Since $\widetilde{K}$ is not used in $\mathrm{Game}^{\mathcal{E}}$, it can be perfectly simulated for $\mathscr{A}$. Hence for an adversary $\mathscr{A}'$ we have that

$$\Pr\left[\mathrm{Game}^{\mathcal{E}}(\mathscr{A}(\widetilde{K}))\right] = \Pr\left[\mathrm{Game}^{\mathcal{E}}(\mathscr{A}')\right]\ .$$

[7] We note that stronger security notions such as security under chosen-ciphertext, related-key, or key-dependent message attacks do not fall under this class of games. Strengthening the surveillance game can allow for a more general preservation theorem to be established.

The first part of the theorem now follows since we have that

$$\begin{aligned}
\mathbf{Adv}^{\overline{\mathrm{SURV}}}_{\Pi,\widetilde{\Pi}}(\mathscr{B}) &= \Pr\left[b' = 0 | b = 0\right] - \Pr\left[b' = 0 | b = 1\right] \\
&= \Pr\left[\mathrm{Game}^{\widetilde{\mathcal{E}}}(\mathscr{A}(\widetilde{K}))\right] - \Pr\left[\mathrm{Game}^{\mathcal{E}}(\mathscr{A}(\widetilde{K}))\right] \\
&= \Pr\left[\mathrm{Game}^{\widetilde{\mathcal{E}}}(\mathscr{A}(\widetilde{K}))\right] - \Pr\left[\mathrm{Game}^{\mathcal{E}}(\mathscr{A}')\right] \\
&= \frac{1}{2} \cdot \left(\mathbf{Adv}^{\mathrm{Game}}_{\widetilde{\Pi}}(\mathscr{A}) - \mathbf{Adv}^{\mathrm{Game}}_{\Pi}(\mathscr{A}')\right).
\end{aligned}$$

The second part of the theorem follows from the definition of surveillance security. □

The above theorem shows that surveillance security constitutes a sufficient condition for the generic preservation of security across a broad class of security games. It is, however, unclear if this condition is also necessary. To prove this one would need to model the surveillance game itself as a generic IND game. A natural way would be to define a game that flips a bit and give the adversary access to $\mathcal{E}_K(\cdot)$ of $\widetilde{\mathcal{E}}_{\widetilde{K},K}(\cdot)$ according to the value of the bit. This argument is valid as long as given $\widetilde{K}$ algorithm $\widetilde{\mathcal{E}}_{\widetilde{K},K}(\cdot)$ can be simulated with oracle access to $\mathcal{E}_K(\cdot)$. Such *black-box* subversions, although powerful, do not necessarily encompass the set of all (possibly white-box) subversions. Nevertheless, surveillance security is both necessary and sufficient for security preservation with respect to generic IND games for black-box subversions.