Anamorphic Encryption, Revisited[*]

Fabio Banfi[†]
Zühlke Engineering AG
fabio.banfi@zuhlke.com

Konstantin Gegier
ETH Zurich
konstantin.gegier@inf.ethz.ch

Martin Hirt
ETH Zurich
hirt@inf.ethz.ch

Ueli Maurer
ETH Zurich
maurer@inf.ethz.ch

Guilherme Rito[†]
Ruhr-Universität Bochum
guilherme.teixeirarito@rub.de

Abstract

An anamorphic encryption scheme allows two parties who share a so-called *double key* to embed covert messages in ciphertexts of an established PKE scheme. This protects against a dictator that can force the receiver to reveal the secret keys for the PKE scheme, but who is oblivious about the existence of the double key. We identify two limitations of the original model by Persiano, Phan, and Yung (EUROCRYPT 2022). First, in their definition a double key can only be generated once, *together* with a key-pair. This has the drawback that a receiver who wants to use the anamorphic mode *after* a dictator comes to power, needs to deploy a new key-pair, a potentially suspicious act. Second, a receiver cannot distinguish whether or not a ciphertext contains a covert message.

In this work we propose a new model that overcomes these limitations. First, we allow to associate *multiple* double keys to a key-pair, *after* its deployment. This also enables *deniability* in case the double key only depends on the public key. Second, we propose a natural robustness notion, which guarantees that anamorphically decrypting a regularly encrypted message results in a special symbol indicating that no covert message is contained, which also eliminates certain attacks.

Finally, to instantiate our new, stronger definition of anamorphic encryption, we provide generic and concrete constructions. Concretely, we show that ElGamal and Cramer-Shoup satisfy a new condition, *selective randomness recoverability*, which enables robust anamorphic extensions, and we also provide a robust anamorphic extension for RSA-OAEP.

---

[*] This is the full version of an article appearing in the proceedings of EUROCRYPT 2024 which is available at https://doi.org/10.1007/978-3-031-58723-8_1.

[†] Work done while the author was at ETH Zurich.

Contents

1 Introduction

1.1 Background and Motivation

Cryptography has a huge impact on society, particularly with regards to the right to privacy. The increased use of electronic communication has heightened concerns over privacy, leading to debates between researchers and politicians about the need to limit encryption as a safeguard to privacy.

In [PPY22], Persiano, Phan, and Yung, point out that the security guarantees offered by cryptography for private communication rely on two implicit and fundamental assumptions: the sender-freedom assumption and the receiver-privacy assumption. The former assumes that, when using a regular public-key encryption (PKE) scheme, the sender is free to pick the message to be sent, while the latter assumes that the message is considered private based on the assumption that the receiver's private key is not compromised. The authors argue that both assumptions can be challenged by those parties whose power cryptography threatens to limit. For instance, in a dictator-led country, individuals can be forced by authorities to encrypt and send adversarially-selected messages, thereby undermining the sender-freedom assumption. Additionally, law enforcement agencies can request the private keys of citizens, thereby undermining the receiver-privacy assumption. This presents a significant challenge for cryptography, particularly as governments around the world seek to limit the power of encryption in order to maintain control.

In order to overcome the second challenge, Persiano et al. introduce a new cryptographic paradigm, *receiver-anamorphic encryption*. As introduced in [PPY22], a receiver-anamorphic PKE scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ can be deployed in two different ways: *normal* or *anamorphic*. In the normal mode, a receiver initially generates a regular key-pair (sk, pk) using $\mathsf{PKE.Gen}$ and authentically broadcasts the public key pk to everyone; then a sender can encrypt a message $\mathtt{m}$ with pk using $\mathsf{PKE.Enc}$, which can be recovered by the receiver from the resulting ciphertext $\mathtt{c}$ with the secret key sk using $\mathsf{PKE.Dec}$. Additionally, PKE is associated with a so-called *anamorphic triplet* $\mathsf{AnamPKE} = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$. In the anamorphic mode, the receiver initially generates an *anamorphic* key-pair (ask, apk) together with a so-called *double key* dk using $\mathsf{AnamPKE.aGen}$, and only broadcasts the anamorphic public key apk, which ideally is indistinguishable from a regular public key pk. Now, any sender can use apk to normally encrypt a message $\mathtt{m}$ using $\mathsf{PKE.Enc}$, but selected senders, with whom the receiver additionally shared the double key dk, using $\mathsf{AnamPKE.aEnc}$ will be able to embed a covert message $\hat{\mathtt{m}}$ in the encryption of a normal message $\mathtt{m}$. Hence, a ciphertext $\mathtt{c}$ produced using the double key dk (which necessarily also contains the anamorphic public key apk) carries two messages: the normal message $\mathtt{m}$ that can be decrypted with the anamorphic secret key ask using $\mathsf{PKE.Dec}$, and the covert message $\hat{\mathtt{m}}$ that can be decrypted with the double key dk using $\mathsf{AnamPKE.aDec}$. Such scheme allows to protect sensitive information from an adversary who might force the receiver to surrender the secret key. When requested to do so, the owner of an anamorphic key-pair (ask, apk) will reveal the associated anamorphic secret key ask, but *not* the double key dk. The adversary will then gain access only to the normal message $\mathtt{m}$, while the covert message $\hat{\mathtt{m}}$ containing sensitive information remains private. The security requirement is that the anamorphic key-pair (ask, apk) must be indistinguishable from its normal counterparts (sk, pk), and the ciphertexts produced using the double key dk must be indistinguishable from those produced using a normal public key pk.

For the receiver-anamorphic setting, on which this paper solely focuses, Persiano et al. put forth two constructions. The first is based on *rejection sampling*, a technique inspired by the biased-ciphertext attack of [BPR14], that allows to send one bit as covert message. As part of their shared double key, sender and receiver agree upon a secret key K for a PRF F mapping ciphertexts to bits. Then, to embed a covert bit b in the encryption of a normal message m, the sender generates fresh ciphertexts $c \leftarrow \mathsf{Enc}(pk, m)$ until $\mathsf{F}(\mathsf{K}, c) = b$. Note that this approach can be naturally extended to bitstrings, but to keep anamorphic encryption efficient, the sender can only transmit logarithmically many bits in the security parameter κ.

The second construction is based on the celebrated Naor-Yung transform (NYT), that given an IND-CPA PKE scheme $\mathsf{CPA} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and a simulation-sound NIZK $\mathsf{NIZK} = (\mathsf{Prove}, \mathsf{Verify})$ for a polynomial-time relation capturing equality of plaintext for CPA, with simulator $(\mathsf{S}_0, \mathsf{S}_1)$, yields an IND-CCA PKE scheme $\mathsf{NYT} = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$. We first briefly recall how the traditional NYT is defined, and then give an overview of its anamorphic mode given by Persiano et al. The key generation algorithm $\mathsf{NYT.Gen}'$ defines the public key $pk' := (pk_0, pk_1, \sigma)$, where pk_0 and pk_1 two independently chosen public keys of CPA, and σ is a CRS for NIZK. Crucially, the secret key is defined only as $sk' := sk_0$, the secret key associated with pk_0, that is, sk_1 (the secret key associated with pk_1) is *forgotten* by $\mathsf{NYT.Gen}'$. To encrypt message m, the encryption algorithm $\mathsf{NYT.Enc}'$ first computes ciphertexts $c_0 := \mathsf{CPA.Enc}(pk_0, m; r_0)$ and $c_1 := \mathsf{CPA.Enc}(pk_1, m; r_1)$, using random and independent coin tosses r_0 and r_1. Then, it runs NIZK.Prove on input instance $((pk_0, c_0), (pk_1 c_1))$, witness (r_0, r_1, m), and σ from pk', producing a proof π that c_0 and c_1 encrypt the same message, and finally outputs ciphertext $c := (c_0, c_1, \pi)$. The decryption algorithm $\mathsf{NYT.Dec}'$ simply runs NIZK.Verify to check π and, if successful, outputs $m := \mathsf{Dec}(sk_0, c_0)$.

The anamorphic triplet $\mathsf{AnamNYT} = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ for NYT is defined as follows. The anamorphic key generation algorithm AnamNYT.aGen defines the anamorphic public key as $apk := (pk_0, pk_1, \sigma)$, where pk_0 and pk_1 two independently chosen public keys of CPA, but the CRS σ is instead obtained by running simulator S_0, which returns the CRS-trapdoor pair (σ, τ). The anamorphic secret key is defined as $ask := sk_0$, whereas the double key is defined as $dk := (pk_0, pk_1, sk_1, \tau)$. The anamorphic encryption algorithm AnamNYT.aEnc takes as input the double key dk, a normal messages m, and a covert message $\hat{m}$ from the same message space, and first computes $c_0 := \mathsf{CPA.Enc}(pk_0, m; r_0)$ and $c_1 := \mathsf{CPA.Enc}(pk_1, \hat{m}; r_1)$. The (fake) proof π is then constructed by running the simulator S_1 on input the trapdoor τ and the instance $((pk_0, c_0), (pk_1, c_1))$. The anamorphic decryption algorithm AnamNYT.aDec works exactly as $\mathsf{NYT.Dec}'$. Maybe somewhat surprisingly, the anamorphic mode for the Naor-Yung scheme allows to embed covert messages from the same domain of normal messages. Persiano et al. argue that therefore, this effectively allows for bandwidth rate of 1, as opposed to $(\log \kappa)/\kappa$ for the rejection sampling technique.

Limitation of Persiano's et al. Work. In this paper we identify two limitations of the original work. The first is that in Persiano et al.'s model of (receiver-)anamorphic encryption, anamorphic key-pairs and double keys are coupled, that is, once the public key is deployed, it is not possible anymore to associate with it a new double key. This is indeed the case for their Naor-Yung anamorphic encryption scheme. To see this, imagine a receiver broadcasting a regular NYT public key (pk_0, pk_1, σ); now, since the CRS σ is generated normally rather than with the simulator S_0, it is impossible to associate a fresh double key $dk = (pk_0, pk_1, sk_1, \tau)$, because the receiver neither has sk_1 (recall that this is *forgotten* by regular key-generation), nor the trapdoor τ.

The second limitation is that in the original set of notions, an important property is missing: When decrypting anamorphically a ciphertext that was generated using normal encryption, it should be natural to except an error signaling the receiver that the ciphertext is void of any covert message. By default this is not achieved by both the rejection sampling technique and the Naor-Yung anamorphic scheme, since anamorphic decryption will always output a message.

We will therefore modify the model by allowing double keys to be created independently of key-pairs, and also introduce a new notion for anamorphic encryption which we call *robustness*, addressing the above mentioned issue. We will first develop simple solutions relying on sender and receiver being synchronized by keeping matching counters, but the main challenge will be to get rid of this last assumption. Therefore, the natural question which we fully solve in this paper can be summarized as:

> *Can we construct (receiver-)anamorphic PKE schemes that are robust and do not require the sender and the receiver to be synchronized?*

We will affirmatively answer this question by proposing both an improvement of the model, as well as novel constructions within it, starting from ones that assume synchrony between sender and receiver, and culminating in one which is *unsynchronized*. We see a parallel between our work and that of Abdalla et al. [ABN10], which introduced the robustness notion for PKE only a decade after the notion of key-privacy was originally introduced by Bellare et al. [BBDP01], and which are by now understood to be two essential properties which go hand in hand in the context of anonymity (cf. [KMO+13]).

1.2 Contributions

Stronger Model. In [PPY22], the anamorphic key generation algorithm outputs a key-pair and a double key. The requirement is then that the output “anamorphic” key-pair essentially is indistinguishable from a regular key-pair. We strengthen the model by requiring that the anamorphic key generation only outputs a double key, on input a key-pair. Therefore, in our model there is no “anamorphic” key-pair, and for this reason in this work we use quotation marks (when referring to concepts from the original work of [PPY22]). For the same reason, in our work we update the original term “anamorphic triplet” used by Persiano et al. to “anamorphic *extension*”. We identify several advantages of our new model.

Multiple double keys. Our new model allows the receiver to set up several double keys for its public key, not just one. This enables the possibility to have multiple covert channels, something that is for example impossible to achieve with the Naor-Yung anamorphic encryption scheme.

On-the-fly double keys. Maybe even more crucially, being able to open a covert channel *after* having deployed a public key seems to be a crucial requirement in the dictator model envisioned by Persiano et al. Indeed, we think that the Naor-Yung anamorphic encryption scheme partially contradicts the original paradigm, since a dictator that comes to power *after* the receiver has deployed a public key, will be suspicious of such public key being updated (to accommodate for a secret double key).

Covert channel towards a different receiver. Another advantage of decoupling double keys from key pairs is that it potentially allows to embed covert messages addressed to a party *different* than the one in possession of the secret key associated with the public key used. We remark that this is only the case if anamorphic decryption does *not* depend on the secret key of the normal receiver, which will be the case for all our constructions but the first.

Deniability. Finally, we note that our model naturally enables an important property for the receiver: *deniability*. In case the double key *only depends on the public key* (which is indeed the case for all of our constructions), a malicious sender holding the receiver's double key cannot convince the dictator that the receiver holds the same double key, and therefore would be misbehaving (in the eyes of the dictator). More precisely, because the double key can be generated either by the sender or the receiver, the sender could simulate a double key and a couple of messages, without the help of the receiver.

This is in contrast with Persiano et al.'s anamorphic Naor-Yung transform AnamNYT. There, note that a malicious sender could frame a receiver who set up a double key in conjunction with an "anamorphic" key-pair (and therefore implicitly depending on the "anamorphic" secret key) simply by conveying the double key $dk = (pk_0, pk_1, sk_1, \tau)$ to the dictator. Then, the latter can do the following, to be convinced that indeed the receiver set up the key-pair $pk = (pk_0, pk_1, \sigma)$ anamorphically: First, pick two different messages m_0, m_1 and obtain the ciphertexts $c_0 := \mathsf{CPA.Enc}(pk_0, m_0; r_0)$ and $c_1 := \mathsf{CPA.Enc}(pk_1, m_1; r_1)$, for some fresh and independent randomnesses r_0, r_1. Then, produce a fake proof π using the simulator S_1 on input the trapdoor τ (from the double key dk) and the (false) instance $((pk_0, c_0), (pk_1, c_1))$. Finally, run NIZK.Verify on input the CRS σ, the instance $((pk_0, c_0), (pk_1, c_1))$, and the proof π. If this yields 1, then the dictator can tell that indeed it was able to generate a covert message that is bound to the receiver via its public key pk. The key insight is that this proves that the receiver must have shared the trapdoor of the simulated CRS with the snitching sender, as otherwise the latter could not have come up with a valid trapdoor τ for the CRS σ that would have made the dictator successfully verify.

Robustness Notion. In [PPY22], no notion of robustness was considered. In particular, for a given PKE scheme (Gen, Enc, Dec) with an anamorphic extension (aGen, aEnc, aDec) and honestly generated ("anamorphic") key-pair (sk, pk) and double key dk, the authors only contemplated the following three cases (additionally to the case consisting of the regular use of the base scheme):

1. A message and covert message pair $(m, \hat{m})$ is encrypted using the *anamorphic* encryption algorithm aEnc and is decrypted using the *anamorphic* decryption algorithm aDec (*fully anamorphic encryption mode* or fAME in the original work).

2. A message and covert message pair $(m, \hat{m})$ is encrypted using the *anamorphic* encryption algorithm aEnc and is decrypted using the *regular* decryption algorithm Dec (*anamorphic with normal decryption* or andAME in the original work).

3. A message m is encrypted using the *regular* encryption algorithm Enc and is decrypted using the *regular* decryption algorithm Dec (*normal mode of operation* or nAME in the original work).

Clearly, an important case is missing:[1]

4. A message m is encrypted using the *regular* encryption algorithm Enc and is decrypted using the **anamorphic** decryption algorithm aDec.

In this latter case, it is intuitively desirable that a special symbol $\perp$ is output indicating that the ciphertext (intentionally) contains no covert message. This is important because in the dictator model introduced in [PPY22], a crucial paradigm is that of anamorphically enhancing schemes that are well-established, and therefore potentially already being actively used for regular communication (and only occasionally required to transmit covert messages, from some point in time onward). We put forth a notion of *robustness* for anamorphic encryption that aims exactly at capturing this. We require that messages encrypted with the regular encryption algorithm, if decrypted anamorphically, reveal no covert message whatsoever (since there was none meant in the first place), that is, the special symbol $\perp$ is output instead.

Maybe even more critically, we also observe that robustness might be not solely about functionality, but about security as well: The dictator could trick receivers into reveling that they are indeed in possession of a double key by sending them normally encrypted messages and observing whether they show any reaction. For a non-robust anamorphic scheme this might indeed be the case, while for a robust scheme this attack yields no information to the dictator.

Assuming we have an anamorphic extension of a PKE scheme that is not robust in the sense above, a naive approach that achieves the notion is for the sender and receiver to agree on a subspace of the covert messages that are deemed invalid. The larger such subspace, the higher the chances that a ciphertext not intentionally carrying a covert message, is not falsely interpreted by the receiver as instead carrying one. For example, considering the rejection sampling technique outlined before for transmitting one bit covertly, one could pick a PRF mapping ciphertexts to $\mathsf{t}+1$ bits instead of just one bit, and then require that only ciphertext mapped by the PRF to a bitstring $0^{\mathsf{t}}\mathsf{b}$, for some $\mathsf{b} \in \{0,1\}$, are to be understood as intentionally carrying the covert message b.

Indeed, in one of our constructions, we will follow this approach. Still, a natural question is whether it is possible to construct anamorphic extensions that already achieve robustness, without the need of sacrificing a subset of the possible covert messages as in the approach outlined above. We will show that indeed such anamorphic extensions are possible.

Constructions. We begin by providing a generic approach to obtain anamorphic extensions achieving robustness that assumes sender and receiver to be synchronized, that is, by assuming they use *matching* counters to anamorphically encrypt and decrypt the same covert message. We then identify a new class of PKE schemes by putting forth a new property, which we call *selective randomness recoverability* (SRR), and which allows for the parties to be *unsynchronized.* More

---

[1] The original work considers a further case, the *anamorphic with normal encryption* or aneAME, but in our model, since the anamorphic key generation algorithm does *not* output a key-pair, this case is equivalent to our third case, and hence irrelevant.

precisely, the sender will keep state (or even be stateless), but crucially the receiver will be able to decrypt without the need of knowing the sender's state. We will show that the well-established schemes of ElGamal [ElG85] and Cramer-Shoup [CS98] satisfy our SRR notion, and can therefore be used in a robustly anamorphic mode. Finally, we also provide a generic transformation yielding a robustly anamorphic encryption scheme from one that is anamorphic but not robust. We apply this transformation to the OAEP scheme from [BR95], thus showing that the well-established RSA-OAEP scheme can be used in a robustly anamorphic mode.

On Covert Message Space Size. We remark that our constructions support covert messages smaller than normal messages, unlike Persiano et al.'s anamorphic Naor-Yung transform AnamNYT which allows covert messages of the same size as normal ones. Indeed, as the authors of [PPY22] claim, their scheme achieves bandwidth rate of 1, since it is possible to embed covert messages from the same space as that of normal messages. However, we argue that their notion of bandwidth rate is perhaps not practically justifiable. Rather than comparing the size of covert messages with the size of normal messages, we propose to compare it with *the expansion caused by encryption*. Intuitively, this makes sense, because, in practical terms, the number of extra bits introduced by encrypting define an upper bound on how many bits can actually be securely embedded as covert message in the ciphertext. With respect to this new notion, the scheme AnamNYT would not achieve bandwidth rate of 1. See Section 3.3 for a more in-depth discussion.

1.3 Related Work

Anamorphic encryption shares similar goals with key-escrow [Mic93, Bla94, FY95, Dak96, AAB$^+$97, YY98, AAB$^+$15, GKL21], deniable encryption [CDNO97], kleptography [YY96, YY97, YY98, YY10, CNE$^+$14, BPR14, RTYZ16, RTYZ17], (public-key) steganography [Sim83, R$^+$98, vH04], and subvertable backdoored encryption [HPRV19], but also significantly differs from those in various aspects. For a comprehensive comparison to the cited papers, we refer the reader to the original work by Persiano et al. [PPY22].

Concurrently to our work, Kutyłowski, Persiano, Phan, Yung, and Zawada have extended the anamorphic setting to capture digital signatures [KPP$^+$23b] and also investigated more PKE schemes satisfying their notions of receiver-anamorphic encryption [KPP$^+$23a]. We point out that in the latter work, even though the authors considered anamorphic modes for schemes also studied in our paper, such as ElGamal, Cramer-Shoup, and RSA-OAEP, their model does *not* include the notion of robustness. Moreover, maybe even more importantly, we note that one of their solutions, used for example in their anamorphic mode for ElGamal and Cramer-Shoup, consists of including parts of the receiver's secret key in the double key[2], which is *not* the case for any of our constructions. We consider this approach dissatisfactory, since sharing the secret key directly undermines the security of the regular use of the underlying PKE scheme.

After publication of our preprint [BGH$^+$23], more articles pertaining robustness in anamorphic encryption appeared. Wang, Chen, Huang, Yung [WCHY23] introduced an analogous robustness notion for *sender*-anamorphic encryption. Their notion also captures the guarantee that decryption

---

[2] In case of ElGamal the *whole* secret key is included.

of anamorphic ciphertexts with the wrong duplicate secret key should result in an explicit abort signal. In [CGM24], Catalano, Giunta, and Migliaro extend the notion of anamorphism to homomorphic encryption, and in their work they also provide new anamorphic constructions of regular PKE schemes that also satisfy robustness. In particular they show how both the classic hybrid encryption paradigm and the IBE-to-CCA construction enable a robust anamorphic extension.

2 Preliminaries

2.1 Notation

Let $\mathbb{N} = \{1, 2, \ldots\}$. For any $n \in \mathbb{N}$, we use the convention $[n] \doteq \{1, \ldots, n\}$. For any sets $\mathcal{K}, \mathcal{T}$, we model a look-up table $\mathbf{T}$ mapping a key $k \in \mathcal{K}$ to a value $v \in \mathcal{V}$ as a function $\mathcal{K} \to \mathcal{V} \cup \{\bot\}$, and we define the following operations: Initializing a look-up table $\mathbf{T}$ to an empty one is denoted $\mathbf{T} := []$; Assigning value v to key k in $\mathbf{T}$ is denoted $\mathbf{T}[k] := v$, and we assume that any value previously assigned to k will be overwritten by v; Reading the value assigned to key k in $\mathbf{T}$ and assigning it to v is denoted $v := \mathbf{T}[k]$, and if $\mathbf{T}$ does not hold any value for k (that is, no value has been assigned to k in $\mathbf{T}$ before), then v will be assigned the special symbol $\bot$. Finally, if X is a finite set, we let $x \overset{\$}{\leftarrow} X$ denote picking an element of X uniformly at random and assigning it to x, and for a probabilistic algorithm A we let $y \leftarrow A^{O_1, O_2, \ldots}$ denote running A with oracle access to $O_1, O_2, \ldots$, modeled as functions, and assigning the output to y.

2.2 Games, Adversaries, and Reductions

We work in the concrete security setting pioneered by Bellare et al. [BKR94, BDJR97], and use the code-based game-playing framework of Bellare and Rogaway [BR06]. All of our security notions are defined following the *real-or-ideal* paradigm, where a *distinguisher*[3] needs to tell two games apart. A game G specifies a number of procedures $O_1, O_2, \ldots$ that model oracles for the distinguisher D. G also optionally defines a procedure $\textsc{Init}$, and (if not specified otherwise), D will output a bit b. Execution of distinguisher D with game G then consists of running D with oracle access to $\textsc{Init}$ and $O_1, O_2, \ldots$, with the restrictions that D's single call to $\textsc{Init}$ must be its first overall call. Any input given to $\textsc{Init}$ as well as any variable specified therein, will be subsequently accessible (both readable and writable) to any of the other oracles. Moreover, a distinguisher D' internally running another distinguisher D with (simulated) oracles $\textsc{Init}^*, O_1, O_2, \ldots$ will also learn all the input and output values resulting from the interaction of D with its oracles, and these will be implicitly referred in the code. The output of the execution is the bit output by D, and we use the notation $\Pr[G(D)] \doteq \Pr[b = 0 \mid b \leftarrow D^{\textsc{Init}, O_1, O_2, \ldots}]$. We abuse notation and let $\Pr[\mathsf{bad}]$ denote the probability that a flag bad (initially set to false) is set to true in some game. Finally, in order not to overload the notation, we associate public and anamorphic parameters of schemes *implicitly* in games and adversaries.

---

[3] We use the term distinguisher rather than adversary because the latter is more general, but our notions are all real-or-ideal.

```
Game $\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}$
  INIT():
    $\mathsf{K} \xleftarrow{\$} \mathcal{K}$
  EVAL(X):
    return $\mathsf{F}(\mathsf{K}, \mathsf{X})$
```

```
Game $\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}$
  INIT():
    $\mathsf{f} \xleftarrow{\$} \mathcal{Y}^{\mathcal{X}}$
  EVAL(X):
    return $\mathsf{f}(\mathsf{X})$
```

Figure 1: Games defining prf security for a function $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$.

2.3 Public-Key Encryption (PKE)

We begin by recalling the conventional syntax of public-key encryption and its associated notion of security. In this work we only consider indistinguishability under chosen-plaintext attacks, rather than chosen-ciphertext attacks.

Definition 2.1. *A* public-key encryption (PKE) *scheme is a tuple* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *depending on some implicit public parameter* pp*, where:*

- Gen *is a probabilistic algorithm that outputs a key-pair* $(sk, pk) \leftarrow \mathsf{Gen}()$*.*
- Enc *is a probabilistic algorithm that on input a public key* pk *and a message* $m \in \mathcal{M}$*, outputs a ciphertext* $c \leftarrow \mathsf{Enc}(pk, m)$*. When necessary, we make the randomness* $r \in \mathcal{R}$ *explicit by writing* $c := \mathsf{Enc}(pk, m; r)$*.*
- Dec *is a deterministic algorithm that on input a secret key* sk *and a ciphertext* c*, outputs a message* $m := \mathsf{Dec}(sk, c) \in \mathcal{M}$*, or potentially a special symbol* $\perp \notin \mathcal{M}$ *indicating an error.*

We call a PKE scheme perfectly correct *if for every message* $m \in \mathcal{M}$*,*

$$\Pr[\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m \,|\, (sk, pk) \leftarrow \mathsf{Gen}()] = 1.$$

In this paper we tacitly consider only PKE schemes that are perfectly correct and which sample the randomness always *uniformly* at random, which means that $c \leftarrow \mathsf{Enc}(pk, m)$ is always the same as $r \xleftarrow{\$} \mathcal{R}$ followed by $c := \mathsf{Enc}(pk, m; r)$. Moreover, since we only consider PKE schemes that achieve IND-CPA security, we also assume that for any pk, m, r, and r', $\mathsf{Enc}(pk, m; r) \neq \mathsf{Enc}(pk, m; r')$.

2.4 Pseudorandom Functions (PRF)

Let $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be an efficiently computable function. We say that F is a *(secure) pseudorandom function* (prf) if for any $\mathsf{K} \in \mathcal{K}$, $\mathsf{F}(\mathsf{K}, \cdot)$ is indistinguishable from a uniformly selected $\mathcal{X} \to \mathcal{Y}$ function f.

Definition 2.2. *For* $\mathsf{F} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$*, we define the advantage of a* prf *distinguisher* D *as*

$$\mathbf{Adv}_{\mathsf{F}}^{\mathsf{prf}}(\mathrm{D}) \doteq \Pr[\mathrm{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}(\mathrm{D})] - \Pr[\mathrm{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}(\mathrm{D})],$$

with games $\mathrm{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}$ *and* $\mathrm{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}$ *as defined in Figure 1. We let* $q(\mathrm{D})$ *denote the total number of queries to* EVAL *made by* D*.*

3 Rethinking the Anamorphic Model

In this section we present our first contribution regarding the model, which we see as putting anamorphic encryption on solid grounds.[4] Recall that in this paper we are only focusing on *receiver*-anamorphic encryption, and therefore we will drop the prefix most of the times.

3.1 Enhancing the Model: Decoupling Double Keys from Key-Pairs

As previously mentioned, our first contribution consists in changing the model for (receiver-)anamorphic encryption so that the process of generating a double key is not coupled with the process of generating a key-pair. Again, this has several advantages, such as allowing to set up double keys *on the fly* for an already deployed public key, the possibility to set up more than just one double key, and therefore have *different* covert channels, and finally also the possibility to set up covert channels towards parties *other* than the holder of the used public key.

Syntax of (Receiver-)Anamorphic PKE. We begin by defining the syntax of an anamorphic extension Σ for a given PKE scheme Π. Note that Π implicitly defines some public parameters pp, upon which Σ's implicit *anamorphic* parameters ap depend.

Definition 3.1. *For a PKE scheme* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with implicit public parameter* pp*, an* anamorphic extension *for* Π *is a tuple* $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *depending on some implicit anamorphic parameter* ap *(depending on* pp*), where:*

- aGen *is a probabilistic algorithm that on input a key-pair* (sk, pk) *for* Π*, outputs a double key* $dk \leftarrow \mathsf{aGen}(sk, pk)$*.*
- aEnc *is a probabilistic algorithm that on input a double key* dk*, a (normal) message* $\mathtt{m} \in \mathcal{M}$ *for* Π*, and a covert message* $\hat{\mathtt{m}} \in \hat{\mathcal{M}}$*, outputs a ciphertext* $\mathtt{c} \leftarrow \mathsf{aEnc}(dk, \mathtt{m}, \hat{\mathtt{m}})$ *for* Π*. When necessary, we make* aEnc *stateful by including a state* $\mathtt{st}$ *as input and a new state* $\mathtt{st}'$ *as output, and writing* $(\mathtt{c}; \mathtt{st}') := \mathsf{aEnc}(dk, \mathtt{m}, \hat{\mathtt{m}}; \mathtt{st})$*. We denote by* ε *the initial empty state.*
- aDec *is a deterministic algorithm that on input a double key* dk *and a ciphertext* $\mathtt{c}$ *for* Π*, outputs a covert message* $\hat{\mathtt{m}} := \mathsf{aDec}(dk, \mathtt{c}) \in \hat{\mathcal{M}}$ *or the special symbol* $\perp \notin \hat{\mathcal{M}}$ *indicating the absence of a covert message. When necessary, we make* aDec *stateful including a state* $\mathtt{st}$ *as input and a new state* $\mathtt{st}'$ *as output, and writing* $(\hat{\mathtt{m}}; \mathtt{st}') := \mathsf{aDec}(dk, \mathtt{c}; \mathtt{st})$*.*

Note that unlike how we defined PKE in Definition 2.1, for anamorphic extensions we do not hard-code correctness in their syntax, but we will rather model it as a separate property. The reason is that for one of our constructions, correctness will not be perfect, but only computational.

---

[4] We identify a parallel between our re-formulation and enhancement of the anamorphic model to the work of Young and Yung [YY18], who claimed to have done the same for universal re-encryption of Golle et al. [GJJS04].

```
Game $\mathsf{G}^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma,\mathfrak{m}}$
  INIT():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $dk \leftarrow \mathsf{aGen}(sk, pk)$
    $\boxed{\mathtt{st} := \varepsilon}$
  AENCADEC($\hat{\mathfrak{m}}$):
    $\mathfrak{c} \leftarrow \mathsf{aEnc}(dk, \mathfrak{m}, \hat{\mathfrak{m}})$
    $\boxed{(\mathfrak{c}; \mathtt{st}') \leftarrow \mathsf{aEnc}(dk, \mathfrak{m}, \hat{\mathfrak{m}}; \mathtt{st})}$
    $\boxed{\mathtt{st} := \mathtt{st}'}$
    $\hat{\mathfrak{m}}' := \mathsf{aDec}(dk, \mathfrak{c})$
    return $\hat{\mathfrak{m}}'$

Game $\mathsf{G}^{\mathsf{cor}\text{-}1}_{\Pi,\Sigma,\mathfrak{m}}$
  INIT():
    // Do nothing
  AENCADEC($\hat{\mathfrak{m}}$):
    return $\hat{\mathfrak{m}}$
```

Figure 2: Games defining correctness of an anamorphic encryption scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with anamorphic extension $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$. The boxed code is for anamorphic extensions with stateful aEnc and stateless aDec.

Correctness of (Receiver-)Anamorphic PKE. For a PKE scheme Π with anamorphic extension Σ, we define *correctness* (cor) by capturing that for any message $\mathfrak{m}$, it must be hard to find a covert message $\hat{\mathfrak{m}}$ which if encrypted anamorphically with $\mathfrak{m}$ into $\mathfrak{c} \leftarrow \mathsf{aEnc}(dk, \mathfrak{m}, \hat{\mathfrak{m}})$ and subsequently anamorphically decrypted into $\hat{\mathfrak{m}}' := \mathsf{aDec}(dk, \mathfrak{c})$, is such that $\hat{\mathfrak{m}}' \neq \hat{\mathfrak{m}}$. Formally, instead of defining a game with a winning condition, we formulate this property as the equivalent distinguishing problem.

Definition 3.2. *For a PKE scheme Π with anamorphic extension Σ and arbitrary message $\mathfrak{m} \in \mathcal{M}$, we define the advantage of a* cor *distinguisher* D *as*

$$\mathbf{Adv}^{\mathsf{cor}}_{\Pi,\Sigma,\mathfrak{m}}(\mathrm{D}) \doteq \Pr[\mathsf{G}^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma,\mathfrak{m}}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{cor}\text{-}1}_{\Pi,\Sigma,\mathfrak{m}}(\mathrm{D})],$$

with games $\mathsf{G}^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma,\mathfrak{m}}$ and $\mathsf{G}^{\mathsf{cor}\text{-}1}_{\Pi,\Sigma,\mathfrak{m}}$ as defined in Figure 2. We let $q(\mathrm{D})$ denote the total number of messages queried to AENCADEC *by* D.

Security of (Receiver-)Anamorphic PKE. Following [PPY22], for a PKE scheme Π with anamorphic extension Σ, we define security in terms of *indistinguishability of anamorphic mode from normal mode* (sec). More specifically, we require for ciphertexts generated by the anamorphic encryption algorithm to be indistinguishable from ciphertexts generated by the normal encryption algorithm.

Definition 3.3. *For a PKE scheme Π with anamorphic extension Σ, we define the advantage of an* sec *distinguisher* D *as*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma}(\mathrm{D}) \doteq \Pr[\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma}(\mathrm{D})],$$

with games $\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma}$ and $\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma}$ as defined in Figure 3. We let $q(\mathrm{A})$ denote the total number of messages queried to AENC *by* A.

```
Game $G^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $dk \leftarrow \mathsf{aGen}(sk, pk)$
    $\boxed{\mathtt{st} := \varepsilon}$
    return $(sk, pk)$
  $\textsc{AEnc}(m, \hat{m})$:
    $c \leftarrow \mathsf{aEnc}(dk, m, \hat{m})$
    $\boxed{(c; \mathtt{st}') \leftarrow \mathsf{aEnc}(dk, m, \hat{m}; \mathtt{st})}$
    $\boxed{\mathtt{st} := \mathtt{st}'}$
    return $c$
```

```
Game $G^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
  $\textsc{AEnc}(m, \hat{m})$:
    $c \leftarrow \mathsf{Enc}(pk, m)$
    return $c$
```

Figure 3: Games defining the sec notion of an anamorphic extension $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ for PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$. The boxed code is for anamorphic extensions with stateful aEnc.

```
Game $G^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $dk \leftarrow \mathsf{aGen}(sk, pk)$
  $\textsc{EncADec}(m\boxed{, \mathtt{st}})$:
    $c \leftarrow \mathsf{Enc}(pk, m)$
    $\hat{m} := \mathsf{aDec}(dk, c\boxed{; \mathtt{st}})$
    return $\hat{m}$
```

```
Game $G^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma}$
  $\textsc{Init}()$:
    // Do nothing
  $\textsc{EncADec}(m\boxed{, \mathtt{st}})$:
    return $\perp$
```

Figure 4: Games defining robustness of an anamorphic encryption scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with anamorphic extension $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$. The boxed code is for anamorphic extensions with stateful aDec.

For a PKE scheme Π with anamorphic extension Σ, [PPY22] additionally defines security in terms of *indistinguishability of anamorphic ciphertexts under a chosen-plaintext attack* (ind-anam-cpa). More specifically, they require that for a fixed (normal) message m, anamorphic encryptions of covert messages $\hat{m}_0$ and $\hat{m}_1$ with m be indistinguishable. They also show that the sec notion for anamorphic extensions implies ind-anam-cpa security, which roughly speaking means that in order to show that anamorphic ciphertexts are indistinguishable from one another, it suffices to show that anamorphic ciphertexts are indistinguishable from regular ones. In Appendix B we reformulate ind-anam-cpa security and reprove the implication in our formalization.

3.2 Enhancing the Model: Robustness

For a PKE scheme Π with anamorphic extension Σ, we define *robustness* (rob) by capturing that it must be hard to find a message m which if encrypted normally into $c \leftarrow \mathsf{Enc}(pk, m)$ and subsequently *anamorphically* decrypted into $\hat{m} := \mathsf{aDec}(dk, c)$, is such that $\hat{m} \neq \perp$. Formally, instead of defining a game with a winning condition, we formulate this property as the equivalent distinguishing problem.

Definition 3.4. *For a PKE scheme* Π *with anamorphic extension* Σ*, we define the advantage of a* rob *distinguisher* D *as*

$$\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma}(D) \doteq \Pr[G^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma}(D)] - \Pr[G^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma}(D)],$$

with games $G^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma}$ *and* $G^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma}$ *as defined in Figure 4. We let* $q(D)$ *denote the total number of messages queried to* $\textsc{EncADec}$ *by* D*.*

3.3 Anamorphic Length Efficiency

A PKE scheme generally expands a messages of length μ to a longer ciphertext of length $\mu + \lambda$, where λ is usually referred to as the *ciphertext expansion.* For an anamorphic encryption scheme, due to basic information-theoretic reasons, the length ℓ of a covert message is bounded by λ, and if robustness is required, it necessarily needs to be shorter than λ. One can hence define the ratio

$$\phi \doteq \frac{\ell}{\lambda}$$

as the *anamorphic length efficiency* of such a scheme, that aims at quantifying how well the anamorphic encryption makes use of the extra λ bits in the ciphertext. Note that this quantity can potentially achieve 1 only if robustness is *not* considered.

Looking ahead, for our ElGamal-based construction from Section 5.2 we have $\lambda \approx \rho$, where ρ is the length of the randomness. Therefore,

$$\phi_{\mathsf{EG}} \approx \frac{\ell}{\rho},$$

where $\ell \ll \rho$ due to our requirement of robustness. More concretely, ℓ needs to be polynomial in the security parameter, while ρ needs to be exponential (this can be concretely obtained by the bounds on the adversarial advantages for correctness and robustness from Lemmas 4.6 and 4.8). Even if this indeed implies that the anamorphic length efficiency of our construction is less than 1, we conjecture that this is an intrinsic requirement for any possible anamorphic extension of ElGamal requiring robustness, as otherwise it seems that the discrete log assumption would be contradicted. We leave it is an interesting open question to prove that indeed our scheme has optimal anamorphic length efficiency, in our updated model of (receiver-) anamorphic encryption that considers robustness.

Let us now compute the newly defined anamorphic length efficiency ϕ_{NYT} for Persiano et al.'s anamorphic Naor-Yung transform $\mathsf{AnamNYT}$ instead. For this, let λ' be the ciphertext expansion of the underlying IND-CPA scheme and ω the proof size of the underlying NIZK scheme. Then, we have that $\mu + \lambda = 2(\mu + \lambda') + \omega$, that is,

$$\lambda = \mu + 2\lambda' + \omega.$$

Now, consider an underlying IND-CPA scheme with randomness of (fixed) size ρ' which is optimal in the sense that $\lambda' \approx \rho'$ (this is indeed the case for many practical schemes, such as for ElGamal). Then, we can rewrite the above equation as

$$\lambda \approx \mu + 2\rho' + \omega.$$

Therefore, since $\ell = \mu$ for $\mathsf{AnamNYT}$, according to our new notion of anamorphic length efficiency, we have

$$\phi_{\mathsf{NYT}} \approx \frac{\mu}{\mu + 2\rho' + \omega} < 1.$$

We point out that Persiano et al. used a different ratio to estimate the efficiency an anamorphic encryption scheme, namely ℓ/μ, which they call *bandwidth rate*. Indeed, according to this notion, $\mathsf{AnamNYT}$ achieves bandwidth rate of 1.

4 Generic Robustly Anamorphic Extensions

In this section we present four ways to achieve robustly anamorphic public-key encryption. We begin with an overview of the results by giving, for each construction, an informal interpretation and practicality considerations. All proofs of the main results in this section are deferred to Appendix A.

4.1 Overview of the Results

We begin by proposing a simple approach that allows to enhance any PKE scheme into one which has a robust anamorphic extension by embedding covert messages in the randomness upon encryption. This first solution is synchronized, that is, requires sender and receiver to keep matching counters for each new covert message.

Theorem (informal). Σ_1 *(Construction 1) provides a secure and robust (Definitions 3.3 and 3.4) synchronized anamorphic mode for any PKE scheme for transmitting at most* τ *covert messages (Lemmas 4.1 and 4.2), for any* τ *with* $\log \tau$ *polynomial in the underlying security parameter.*

We then optimize this anamorphic extension for a special class of PKE schemes, which encompasses the classic ElGamal [ElG85] and Cramer-Shoup [CS98] schemes.

Theorem (informal). Σ_2 *(Construction 2) provides a secure and robust (Definitions 3.3 and 3.4) synchronized anamorphic mode for any SRR PKE scheme (Definition 4.1) for transmitting at most* τ *covert messages (Lemmas 4.1 and 4.2), for any* τ *with* $\log \tau$ *polynomial in the underlying security parameter.*

We then proceed by optimizing the anamorphic extension even further for such special PKE schemes, resulting in a robustly anamorphic PKE scheme that does not require the sender and the receiver to be synchronized, at the cost of reducing the number of covert messages that can be transmitted overall.

Theorem (informal). Σ_3 *(Construction 3) provides a correct, secure, and robust (Definitions 3.2, 3.3 and 3.4) unsynchronized anamorphic mode for any SRR PKE scheme (Definition 4.1) for transmitting $\sigma\tau$ covert messages (Lemmas 4.6, 4.7 and 4.8), for any σ and τ polynomial in the underlying security parameter.*

Finally, we provide a means to transform an anamorphic PKE scheme that is not robust into one which is. We apply this approach to the OAEP scheme from [BR95], hence showing that also RSA-OAEP can be used in a robustly anamorphic mode.

Theorem (informal). Σ_4 *(Construction 3) provides a secure and robust (Definitions 3.3 and 3.4) synchronized anamorphic mode for any (non-robust) anamorphic PKE scheme (Lemma 4.9).*

	Σ_1 (sync.)	Σ_2 (sync.)		Σ_3 (unsync.)		Σ_4 (sync.)
		w/o P.C.	w/ P.C.	w/o P.C.	w/ P.C.	
aGen	1	1	$\leq \ell$	1	$\leq \ell$	1
aEnc	1	1	1	$\approx \tau$	$\approx \tau$	1
(sk)aDec	$\leq \ell$	$\leq \ell$	1	$\leq \sigma\ell$	$\leq \sigma$	1

Figure 5: Comparison of the runtime complexities for the introduced constructions, in terms of required iterations. P.C. stands for pre-computation. Note that for Σ_3, the approximated runtime complexity of aEnc is taken in expectation, since the algorithm is randomized.

In Figure 5, we summarize the runtime complexities of each construction. Note that for each of them, the size ℓ of the covert message space must be polynomial in the underlying security parameter, since (sk)aDec (or aGen, in case of pre-computation) have runtime complexity $\leq \ell$.[5] Moreover, note that the further limitation on the number of transmissible covert messages for Σ_3 is due to the runtime complexity of aEnc.

4.2 Σ_1: A Synchronized Solution for Any PKE Scheme

Our first solution allows to embed covert messages in ciphertext of any PKE which is at a minimum randomized and IND-CPA secure. The idea is quite simple: Assuming sender and receiver can be synchronized by keeping a matching counter $\mathtt{ctr}$, whenever the sender wants to embed a covert message $\hat{m}$ from some small space $\hat{\mathcal{M}}$ into an encryption of a normal message m, it will first compute $r := \mathsf{F}(\mathsf{K}, (\mathtt{ctr}, \hat{m}))$, where F is a PRF and K is a key that the two parties agreed upon in advance as part of their double key dk, and then it will generate the ciphertext $c := \mathsf{Enc}(pk, m; r)$. Now, since we are assuming that the receiver knows the exact value $\mathtt{ctr}$ that was used by the sender, it will be able to retrieve $\hat{m}$ simply by first normally decrypting c into $m := \mathsf{Dec}(sk, c)$, and then trial-re-encrypt m as $c' := \mathsf{Enc}(pk, m; \mathsf{F}(\mathsf{K}, (\mathtt{ctr}, \hat{m}')))$ for every $\hat{m}' \in \hat{\mathcal{M}}$, until $c' = c$. At that point, the receiver will know that the successful covert message $\hat{m}'$ for which equality holds was indeed the one meant by the sender, or at least with good enough probability. Note that for this to work, the receiver also needs to additionally provide its secret key sk upon anamorphic decryption,

---

[5] In case of pre-computation, this is true also for the *space* complexity of aDec.

```
$\mathsf{aGen}(sk, pk)$:
    $\mathsf{K} \xleftarrow{\$} \mathcal{K}$
    $dk := (\mathsf{K}, pk)$
    return $dk$

$\mathsf{aEnc}(dk, \mathrm{m}, \hat{\mathrm{m}}; \mathtt{ctr})$:
    $\boxed{\mathrm{r} := \mathsf{F}(\mathsf{K}, (\mathtt{ctr}, \hat{\mathrm{m}}))}$
    $\dbox{\mathrm{r} := \hat{\mathrm{m}} \oplus \mathsf{F}(\mathsf{K}, \mathtt{ctr})}$
    $\mathrm{c} := \mathsf{Enc}(pk, \mathrm{m}; \mathrm{r})$
    return $(\mathrm{c}; \mathtt{ctr} + 1)$

$\mathsf{skaDec}(sk, dk, \mathrm{c}; \mathtt{ctr})$:
    $\mathrm{m} := \mathsf{Dec}(sk, \mathrm{c})$
    foreach $\hat{\mathrm{m}} \in \hat{\mathcal{M}}$ do
        $\boxed{\mathrm{r}' := \mathsf{F}(\mathsf{K}, (\mathtt{ctr}, \hat{\mathrm{m}}))}$
        $\dbox{\mathrm{r}' := \hat{\mathrm{m}} \oplus \mathsf{F}(\mathsf{K}, \mathtt{ctr})}$
        $\mathrm{c}' := \mathsf{Enc}(pk, \mathrm{m}; \mathrm{r}')$
        if $\mathrm{c}' = \mathrm{c}$ then
            return $(\hat{\mathrm{m}}; \mathtt{ctr} + 1)$
    return $\perp$
```

Figure 6: Synchronized robustly anamorphic extensions $\boxed{\Sigma_1 = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{skaDec})}$ from Construction 1 for any PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and $\dbox{\Sigma_1' = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{skaDec})}$ for PKE schemes $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ where the randomness space is a group $\langle \mathcal{R}, \oplus \rangle$.

and for this reason we denote this algorithm slightly differently as skaDec. We next formalize this construction, and then formally prove these two properties.

Construction 1. *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be an arbitrary PKE scheme with randomness space* $\mathcal{R}$ *and* $\rho \doteq |\mathcal{R}|$*. For covert message space* $\hat{\mathcal{M}}$*, with* $\ell \doteq |\hat{\mathcal{M}}|$*, and* F *a function* $\mathcal{K} \times ([\tau] \times \hat{\mathcal{M}}) \to \mathcal{R}$*, let the anamorphic extension* $\Sigma_1 \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{skaDec})$ *with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, \tau, \mathsf{F})$ *be defined as in Figure 6. Note that* Σ_1 *trivially satisfies perfect correctness.*

Security of Σ_1. To see that the scheme is indeed secure, note that we can replace $\mathsf{F}(\mathsf{K}, \cdot)$ by a truly random function f. Therefore, since the counters are assumed not to repeat, r will always be uniformly distributed, hence c will be indistinguishable from a regular ciphertext output by Enc. The following result is restated and proven formally in Appendix A.1.

Lemma 4.1. *Let* Σ_1 *be the anamorphic extension from Construction 1 for an arbitrary PKE scheme* Π*. There exists an efficient transformation of any* sec *distinguisher* D *into a* prf *distinguisher* D' *with* $\mathrm{q}(\mathrm{D}') = \mathrm{q}(\mathrm{D}) \leq \tau$ *such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi, \Sigma_1}(\mathrm{D}) = \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathrm{D}').$$

Robustness of Σ_1. To see that the scheme is indeed robust, further observe that when using regular encryption and sampling a uniformly random r, the chance that for a fixed counter $\mathtt{ctr}$ there exists a covert message $\hat{\mathrm{m}}$ such that $\mathrm{r} = \mathrm{f}((\mathtt{ctr}, \hat{\mathrm{m}}))$, for a uniformly random function f, is $1/\rho$. Note that this probability is negligible if $\mathcal{R}$ has exponential size, and such a collision can happen for each of $\mathcal{A}$'s queries, and for each $\hat{\mathrm{m}}$. The following result is restated and proven formally in Appendix A.1.

Lemma 4.2. *Let* Σ_1 *be the anamorphic extension from Construction 1 for an arbitrary PKE scheme* Π*. There exists an efficient transformation of any* rob *distinguisher* D *into a* prf *distinguisher* D' *with* $q \doteq q(D') = q(D) \leq \tau$ *such that*

$$\mathsf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_1}(D) \leq \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(D') + \frac{q\ell}{\rho}.$$

Σ_1': Assuming the Randomness Space is a Group. If the PKE scheme Π is such that the randomness space $\mathcal{R}$ used by Enc forms a group under some operation $\oplus$, then it is possible to slightly modify the construction Σ_1 into Σ_1' as outlined in Figure 6. This would require F to be a function $\mathcal{K} \times [\tau] \to \mathcal{R}$, and $\hat{\mathcal{M}} \subseteq \mathcal{R}$. The advantage would be that computing F would be faster since the input is smaller, but the constraint would be that covert messages must now fit into the randomness space. In our next construction, we will indeed make this assumptions, hence the proof of security of Σ_1' follows directly from Lemma 4.3.

4.3 Σ_2: A Better Synchronized Solution for Special PKE Schemes

We next present a construction that unlike the previous one does not require the receiver of an anamorphic ciphertext to know the secret key of the original receiver. This has a major advantage: *It is possible for a sender to embed a covert message addressed to a party different than the original receiver of the ciphertext!* To achieve this property, we require a special type of PKE. Ideally, as we will later see in Section 4.5, a scheme that allows to recover the randomness used to generate the ciphertext, naturally lends itself to an anamorphic mode (even though, in this case we would again require the receiver to know the original secret key). Still, as we will next show, it is possible for some scheme to *selectively* recover the randomness used to generate a ciphertext. More precisely, we will use the fact that if a part of the ciphertext depends only on the randomness (and neither on the public key, nor on the message), then if we only use a subset of the randomness space, we can test whether a certain value r was used as randomness. We next formalize the required property on such a PKE scheme, and then outline the whole idea in more detail.

Definition 4.1. *A PKE scheme* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* selectively-randomness-recoverable (SRR) *if the following three conditions hold:*

1. *The randomness space* $\mathcal{R}$ *of* Enc *forms a group under some operation* $\oplus$.
2. *For any public key* pk*, message* m*, and randomness* $r \in \mathcal{R}$*, there exists efficiently computable injective functions* α *and* β *such that for the ciphertext* $c := \mathsf{Enc}(pk, m; r)$*,*

$$c = (\alpha(pk, m, r), \beta(r)).^{6}$$

3. *There exists an efficiently computable function* γ *such that, for any* $a, b \in \mathcal{R}$*,*

$$\gamma(\beta(a \oplus b), b) = \beta(a).$$

---

[6] In practice, the ciphertext might be a bit string, in which case we would instead have $c = \alpha(pk, m, r) \| \beta(r)$. Moreover, note that order does not matter, so we could also have $c = (\beta(r), \alpha(pk, m, r))$.

```
$\mathsf{aGen}(sk, pk)$:
    $K \xleftarrow{\$} \mathcal{K}$
    $dk := (K, pk)$
    return $dk$

$\mathsf{aEnc}(dk, m, \hat{m}; \mathtt{ctr})$:
    $t := F(K, \mathtt{ctr})$
    $r := \hat{m} \oplus t$
    $c := \mathsf{Enc}(pk, m; r)$
    return $(c; \mathtt{ctr} + 1)$

$\mathsf{aDec}(dk, (c_1, c_2); \mathtt{ctr})$:
    $t := F(K, \mathtt{ctr})$
    $s := \gamma(c_2, t)$
    foreach $\hat{m} \in \widehat{\mathcal{M}}$ do
        if $\beta(\hat{m}) = s$ then
            return $(\hat{m}; \mathtt{ctr} + 1)$
    return $\perp$
```

Figure 7: Synchronized robustly anamorphic extension $\Sigma_2 = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ for SRR PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

Consider now a PKE scheme Π that is SRR. Then, in order to embed a covert message $\hat{m}$ into a ciphertext for a normal message m, the sender simply xors $\hat{m}$ with a one-time pad $t := F(K, \mathtt{ctr})$, and uses $r := \hat{m} \oplus t$ as randomness to generate $c := \mathsf{Enc}(pk, m; r)$. By virtue of Π being SRR, it then holds that $c = (\alpha(pk, m, r), \beta(r)))$, where $\beta(r) = \beta(\hat{m} \oplus F(K, \mathtt{ctr}))$, and therefore the receiver can recover $\hat{m}$ knowing K and $\mathtt{ctr}$, since $\gamma(\beta(\hat{m} \oplus F(K, \mathtt{ctr})), F(K, \mathtt{ctr})) = \beta(\hat{m})$. More precisely, on input a ciphertext (c_1, c_2), it first computes $s := \gamma(c_2, F(K, \mathtt{ctr}))$, which equals $\beta(\hat{m})$, and then tries all values $\hat{m}' \in \widehat{\mathcal{M}}$ until $\beta(\hat{m}') = s$. At that point, the receiver will know that the successful covert message $\hat{m}'$ for which equality holds was indeed the one meant by the sender, or at least with good enough probability. We next formalize this construction, and then formally prove these two properties.

Construction 2. *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be an SRR PKE scheme with randomness space* $\mathcal{R}$ *and* $\rho \doteq |\mathcal{R}|$*. For covert message space* $\widehat{\mathcal{M}} \subseteq \mathcal{R}$*, with* $\ell \doteq |\widehat{\mathcal{M}}|$*, and* F *a function* $\mathcal{K} \times [\tau] \to \mathcal{R}$*, let the anamorphic extension* $\Sigma_2 \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, \tau, F)$ *be defined as in Figure 7. Note that* Σ_2 *trivially satisfies perfect correctness.*

Security of Σ_2. To see that the scheme is indeed secure, note that we can replace $F(K, \cdot)$ by a truly random function f. Therefore, since the counters are assumed not to repeat, $t = f(\mathtt{ctr})$ will always be uniformly distributed. This will be true for $r = \hat{m} \oplus t$ as well, since $\mathcal{R}$ is a group, hence c will be indistinguishable from a regular ciphertext output by Enc. The following result is restated and proven formally in Appendix A.2.

Lemma 4.3. *Let* Σ_2 *be the anamorphic extension from Construction 2 for an SRR PKE scheme* Π *satisfying Definition 4.1. There exists an efficient transformation of any* sec *distinguisher* D *into a* prf *distinguisher* D' *with* $q(D') = q(D) \leq \tau$ *such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_2}(D) = \mathbf{Adv}^{\mathsf{prf}}_{F}(D').$$

Robustness of Σ_2. To see that the scheme is indeed robust, further observe that when using regular encryption and sampling a uniformly random r, the chance that for a fixed counter $\mathtt{ctr}$ there exists a covert message $\hat{m}$ such that $\beta(\hat{m}) = \gamma(\beta(r), f(\mathtt{ctr}))$, for a uniformly random function f, is the same as the chance that $r = \hat{m} \oplus f(\mathtt{ctr})$, which is $1/\rho$. Note that this probability is negligible if $\mathcal{R}$

```
$\mathsf{aGen}(sk, pk)$:
    $K \xleftarrow{\$} \mathcal{K}$
    $\mathbf{T} := []$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        $\mathbf{T}[\beta(\hat{m})] := \hat{m}$
    $dk := (K, \mathbf{T}, pk)$
    return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m}; \mathtt{ctr})$:
    $t := F(K, \mathtt{ctr})$
    $r := \hat{m} \oplus t$
    $c := \mathsf{Enc}(pk, m; r)$
    return $(c; \mathtt{ctr} + 1)$
```

```
$\mathsf{aDec}(dk, (c_1, c_2); \mathtt{ctr})$:
    $t := F(K, \mathtt{ctr})$
    $s := \gamma(c_2, t)$
    $\hat{m} := \mathbf{T}[s]$
    return $(\hat{m}; \mathtt{ctr} + 1)$
```

Figure 8: Synchronized robustly anamorphic extension $\Sigma_2' = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ with pre-computation for SRR PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

has exponential size, and such a collision can happen for each of A's queries, and for each $\hat{m}$. The following result is restated and proven formally in Appendix A.2.

Lemma 4.4. *Let Σ_2 be the anamorphic extension from Construction 2 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* rob *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \tau$ such that*

$$\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_2}(D) \leq \mathbf{Adv}^{\mathsf{prf}}_{F}(D') + \frac{q\ell}{\rho}.$$

Σ_2': Optimizing Σ_2 with Pre-Computation. Note that the time complexity of aDec from Σ_2 is still comparable to that of skaDec from Σ_1. Still, for Σ_2 it is possible to perform a significant optimization that cannot be applied to Σ_1. Since the only check done inside the for loop is $\beta(\hat{m}) = s$, it is possible to pre-compute the inverse mapping β^{-1} in form of a look-up table $\mathbf{T}$. More precisely, aGen will insert value $\hat{m}$ under key $\beta(\hat{m})$ in $\mathbf{T}$, and include $\mathbf{T}$ in the double key. Then, upon anamorphic decryption, the for loop can be substituted by a simple look-up operation in $\mathbf{T}$. The resulting scheme Σ_2' is formalized in Figure 8, and it inherits both security and robustness of Σ_2 from Lemmas 4.3 and 4.4.

4.4 Σ_3: An Unsynchronized Solution for Special PKE Schemes

We now show how to take advantage of the SRR property from Definition 4.1 even further, and develop a simple technique yielding an anamorphic extension for any SRR PKE that does away with the requirement of sender and receiver to keep synchronized (on the counters). The gist of it is to create anamorphic ciphertexts so that the receiver can (partially) extract the counters from it. Recall the scheme from Figure 8: Upon anamorphic encryption, we generate the (one-time) pad $F(K, \mathtt{ctr})$, set the randomness as $r := \hat{m} \oplus F(K, \mathtt{ctr})$, where $\hat{m}$ is the covert message, and then (deterministically) obtain the ciphertext c as $\mathsf{Enc}(pk, m; r)$. Since the PKE scheme Π is SRR, recall that $c = (\alpha(pk, m, r), \beta(r))$. The main idea is now to carefully select a counter $\mathtt{ctr}$, such that it is possible to actually recover $\mathtt{ctr}$ itself from $\beta(r) = \beta(\hat{m} \oplus F(K, \mathtt{ctr}))$. To achieve this, one could use an efficiently computable function δ, and repeatedly try fresh values for r, until $\delta(\beta(r)) = \mathtt{ctr}$.

But this approach has a severe limitation: Roughly speaking, on average one value r will correspond to one value $\mathtt{ctr}$, so in the worst case it might be possible *not* to find a pair $(r, \mathtt{ctr})$ such that $\delta(\beta(r)) = \mathtt{ctr}$, which would imply that $\hat{m}$ cannot be anamorphically encrypted! Therefore, we need to "split" the state into $\mathtt{st} = (x, y)$, for $x \in [\sigma]$ and $y \in [\tau]$, for some σ, τ defined as part of the anamorphic parameters ap. We can now concretely require δ to be a $\mathrm{Im}(\beta) \to [\tau]$ function, and look for a pair (x, y) such that $\delta(\beta(\hat{m} \oplus \mathsf{F}(\mathsf{K}, (x, y)))) = y$. In order to ensure that finding such a pair does not take too long, we need that δ partitions $\mathcal{R}$, with $\rho \doteq |\mathcal{R}|$, as uniformly as possible, that is,

$$\forall y \in [\tau] : \ \left|(\delta \circ \beta)^{-1}(y)\right| \geq \left\lfloor \frac{\rho}{\tau} \right\rfloor. \tag{1}$$

Note that (1) implies that for any $y \in [y]$, $\lfloor \rho/\tau \rfloor \leq \left|(\delta \circ \beta)^{-1}(y)\right| \leq \lceil \rho/\tau \rceil$, and therefore also $\rho/\tau - 1 \leq \left|(\delta \circ \beta)^{-1}(y)\right| \leq \rho/\tau + 1$.

To anamorphically decrypt, we first get $y := \delta(c_2) = \delta(\beta(r))$, and then we look for the first x such that there exists an $\hat{m}$ for which $\gamma(c_2, \mathsf{F}(\mathsf{K}, (x, y))) = \beta(\hat{m})$. Since as for Σ_2 it is possible to pre-compute the inverse of β, we directly define this construction by employing a look-up table $\mathbf{T}$ mapping $\beta(\hat{m})$ to $\hat{m}$. Recall that, for $t := \mathsf{F}(\mathsf{K}, (x, y))$, correctness then follows by:

$$\begin{aligned} \hat{m} = \mathbf{T}[s] = \beta^{-1}(s) = \beta^{-1}(\gamma(c_2, t)) = \beta^{-1}(\gamma(\beta(r), t)) \\ = \beta^{-1}(\gamma(\beta(\hat{m} \oplus t), t)) = \beta^{-1}(\beta(\hat{m})) = \hat{m}. \end{aligned}$$

Note that the main advantage is that now the receiver does not need to know the counter to decrypt anamorphically, therefore sender and receiver *need not be synchronized.* Still, a drawback of this approach is that now it is possible for anamorphic decryption to return the wrong covert message. This means, that this construction cannot achieve perfect correctness. Nevertheless, we will show that it achieves computational correctness, by providing a bound that makes explicit how parameters should be set. Looking ahead, Lemma 4.6 essentially says that one should choose the size σ of the domain of the counter part x not to be too large.

We first present a stateful version of this construction, that is, one where the sender keeps updating the state (x, y) by increasing it *lexicographically* for each try. More precisely, given the a state (x, y), we update it to $(x, y + 1)$ if $y < \tau$, to $(x + 1, 1)$ if $y = \tau$ and $x < \sigma$, and $(1, 1)$ otherwise. We denote this operation by $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$. This stateful approach allows for an easier analysis; we then slightly modify it into a stateless construction.

Construction 3. *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be an SRR PKE scheme with randomness space* $\mathcal{R}$ *and* $\rho \doteq |\mathcal{R}|$*, and function* β *as for Definition 4.1. For covert message space* $\hat{\mathcal{M}} \subseteq \mathcal{R}$*, with* $\ell \doteq |\hat{\mathcal{M}}|$*,* F *a function* $\mathcal{K} \times ([\sigma] \times [\tau]) \to \mathcal{R}$*, and* δ *a function* $\mathrm{Im}(\beta) \to [\tau]$ *satisfying (1), let the anamorphic extension* $\Sigma_3 \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, \delta, \sigma, \tau, \mathsf{F})$ *be defined as in Figure 9.*

Efficiency of Σ_3. Since anamorphic encryption aEnc of Σ_3 needs to iterate an undefined number of times, we need to know an estimate of its running time in order to deem it practical, or just to be sure the algorithm indeed terminates. To do so, we make the simplifying assumption that the PRF F is replaced by a truly random function f. Then, since each pair (x, y) input to f is never repeated, we can assume r to be freshly and uniformly distributed in each iteration.

```
aGen($sk$, $pk$):
$K \xleftarrow{\$} \mathcal{K}$
$\mathbf{T} := []$
foreach $\hat{m} \in \hat{\mathcal{M}}$ do
    $\mathbf{T}[\beta(\hat{m})] := \hat{m}$
$dk := (K, \mathbf{T}, pk)$
return $dk$
```

```
aEnc($dk$, $m$, $\hat{m}$; $(x, y)$):
repeat
    $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
    $t := F(K, (x, y))$
    $r := \hat{m} \oplus t$
until $\delta(\beta(r)) = y$
$c := \mathsf{Enc}(pk, m; r)$
return $(c; (x, y))$
```

```
aDec($dk$, $(c_1, c_2)$):
$y := \delta(c_2)$
foreach $x \in [\sigma]$ do
    $t := F(K, (x, y))$
    $s := \gamma(c_2, t)$
    $\hat{m} := \mathbf{T}[s]$
    if $\hat{m} \neq \perp$ then
        return $\hat{m}$
return $\perp$
```

Figure 9: Stateful unsynchronized robustly anamorphic extension $\Sigma_3 = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ with pre-computation for SRR PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

Lemma 4.5. *Let* T *be the random variable denoting the number of iterations performed by* aEnc, *and assume* r *is uniformly distributed over* $\mathcal{R}$ *in each iteration. Then,* $\mathbb{E}[T] \leq \frac{(\rho+\tau)\rho\tau}{(\rho-\tau)^2}$.

Proof. Let $\omega \in \mathbb{N}$ and $r_1, \ldots, r_\omega$ be uniformly distributed over $\mathcal{R}$ and $y_1, \ldots, y_\omega \in [\tau]$ arbitrary (representing the different values taken by y in each iteration). Then, using (1), we have

$$\begin{aligned}
\Pr[T = \omega] &= \Pr\left[\left(\bigcap_{i=1}^{\omega-1}\{(\delta \circ \beta)(r_i) \neq y_i\}\right) \cap \{(\delta \circ \beta)(r_\omega) = y_\omega\}\right] \\
&= \prod_{j=1}^{\omega-1} \Pr[(\delta \circ \beta)(r_i) \neq y_i] \cdot \Pr[(\delta \circ \beta)(r_\omega) = y_\omega] \\
&= \prod_{j=1}^{\omega-1} \Pr[r_i \notin (\delta \circ \beta)^{-1}(y_i)] \cdot \Pr[r_\omega \in (\delta \circ \beta)^{-1}(y_\omega)] \\
&= \left(1 - \frac{\left|(\delta \circ \beta)^{-1}(y_i)\right|}{\rho}\right)^{\omega-1} \cdot \frac{\left|(\delta \circ \beta)^{-1}(y_\omega)\right|}{\rho} \\
&\leq \left(1 - \frac{\rho/\tau - 1}{\rho}\right)^{\omega-1} \cdot \frac{\rho/\tau + 1}{\rho} \\
&= \left(1 - \frac{\rho - \tau}{\rho\tau}\right)^{\omega-1} \cdot \frac{\rho + \tau}{\rho\tau}.
\end{aligned}$$

Therefore, since $\tau < \rho$,

$$\begin{aligned}
\mathbb{E}[T] = \sum_{\omega=1}^{\infty} \omega \cdot \Pr[T = \omega] &\leq \sum_{\omega=1}^{\infty} \omega \cdot \left(1 - \frac{\rho - \tau}{\rho\tau}\right)^{\omega-1} \cdot \frac{\rho + \tau}{\rho\tau} \\
&= \left(1 - \frac{\rho - \tau}{\rho\tau} - 1\right)^{-2} \cdot \frac{\rho + \tau}{\rho\tau} = \frac{(\rho + \tau)\rho\tau}{(\rho - \tau)^2}.
\end{aligned}$$

□

Note that for $\tau \ll \rho$, we have $\mathbb{E}[T] \approx \tau$. Moreover, in case τ divides ρ, then condition (1) can be replaced by $\left|(\delta \circ \beta)^{-1}(y)\right| = \tau/\rho$, resulting in $\mathbb{E}[T] = \tau$.

Correctness of Σ_3. To see that the scheme indeed satisfies computational correctness, suppose a covert message $\hat{m}$ is anamorphically encrypted with a normal message m resulting in ciphertext $c = (c_1, c_2)$ with $c_2 = \beta(r) = \beta(\hat{m} \oplus \mathsf{F}(K, (x, y)))$, for some $x \in [\sigma]$ and $y \in [\tau]$. Then anamorphic decryption of c might give the wrong output in case a $\hat{m}' \neq \hat{m}$ and an $x' \neq x$ exist, such that $\hat{m}' = \mathbf{T}[s]$, that is, $\beta(\hat{m}') = s = \gamma(\beta(\hat{m} \oplus \mathsf{F}(K, (x, y))), \mathsf{F}(K, (x', y)))$. Now, by the definition of γ, we have that this is the case if and only if $\hat{m} \oplus \mathsf{F}(K, (x, y)) = \hat{m}' \oplus \mathsf{F}(K, (x', y))$, which has probability $\sigma\ell/\rho$ of happening. The following result is restated and proven formally in Appendix A.3.

Lemma 4.6. *Let Σ_3 be the anamorphic extension from Construction 3 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* cor *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \sigma\tau$ such that*

$$\mathbf{Adv}^{\mathsf{cor}}_{\Pi,\Sigma_3}(D) \leq \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(D') + \frac{q\sigma\ell}{\rho}.$$

Security of Σ_3. To see that the scheme is indeed secure, note that we can replace $\mathsf{F}(K, \cdot)$ by a truly random function f. Therefore, since the state pairs $(x, , y)$ are assumed not to repeat, $t = f((x, y))$ will always be uniformly distributed. This will be true for $r = \hat{m} \oplus t$ as well, since $\mathcal{R}$ is a group. Once $\delta(\beta(r)) = y$ is satisfied, r will still be freshly and uniformly distributed, hence c will be indistinguishable from a regular ciphertext output by Enc. The following result is restated and proven formally in Appendix A.3.

Lemma 4.7. *Let Σ_3 be the anamorphic extension from Construction 3 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* sec *distinguisher D into a* prf *distinguisher D' with $q(D') = q(D) \leq \sigma\tau$ such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_3}(D) = \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(D').$$

Robustness of Σ_3. To see that the scheme is indeed robust, further observe that when using regular encryption and sampling a uniformly random r, the chance that for a fixed counter $\mathtt{ctr}$ there exists a covert message $\hat{m}$ such that $\beta(\hat{m}) = \gamma(\beta(r), f(\mathtt{ctr}))$, for a uniformly random function f, is the same as the chance that $r = \hat{m} \oplus f(\mathtt{ctr})$, which is $1/\rho$. Note that this probability is negligible if $\mathcal{R}$ has exponential size, and such a collision can happen for each of A's queries, for each $x \in [\sigma]$, and for each $\hat{m}$. The following result is restated and proven formally in Appendix A.3.

Lemma 4.8. *Let Σ_3 be the anamorphic extension from Construction 3 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* rob *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \sigma\tau$ such that*

$$\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_3}(D) \leq \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(D') + \frac{q\sigma\ell}{\rho}.$$

Σ_3': Optimizing Σ_3 with Stateful Anamorphic Encryption. As mentioned above, we can easily modify the stateful anamorphic extension Σ_3 into a stateless anamorphic extension Σ_3', as defined in Figure 10. The idea is simply to pick uniformly random values $x \in [\sigma]$ and $y \in [\tau]$ in each iteration, rather than lexicographically increasing the state pair (x, y). Then, by the birthday problem we have that, the correctness, security, and robustness bounds degrade by approximately an additive term $q^2/\sigma\tau$.

```
$\mathsf{aGen}(sk, pk)$:
$K \xleftarrow{\$} \mathcal{K}$
$\mathbf{T} := []$
foreach $\hat{m} \in \hat{\mathcal{M}}$ do
    $\mathbf{T}[\beta(\hat{m})] := \hat{m}$
$dk := (K, \mathbf{T}, pk)$
return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m})$:
repeat
    $x \xleftarrow{\$} [\sigma]$
    $y \xleftarrow{\$} [\tau]$
    $t := \mathsf{F}(K, (x, y))$
    $r := \hat{m} \oplus t$
until $\delta(\beta(r)) = y$
$c := \mathsf{Enc}(pk, m; r)$
return $c$
```

```
$\mathsf{aDec}(dk, (c_1, c_2))$:
$y := \delta(c_2)$
foreach $x \in [\sigma]$ do
    $t := \mathsf{F}(K, (x, y))$
    $s := \gamma(c_2, t)$
    $\hat{m} := \mathbf{T}[s]$
    if $\hat{m} \neq \perp$ then
        return $\hat{m}$
return $\perp$
```

Figure 10: Unsynchronized robustly anamorphic extension $\Sigma_3' = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ with precomputation for SRR PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

4.5 Σ_4: Making Robust any (Non-Robust) Anamorphic Extension

In this section we present a very simple generic transformation that given a non-robust anamorphic encryption scheme, yields one that is additionally robust. We then show in Section 5.3 how to concretely apply this transformation to the Optimal Asymmetric Encryption Padding (OAEP) technique transforming any trapdoor permutation into a secure PKE scheme from [BR95]. Keeping in mind the original goal of [PPY22], that is to find anamorphic modes of *well-established* schemes, the latter implies that the widely employed RSA-OAEP indeed admits an robustly anamorphic mode, as we will concretely show in Section 5.3.

Our construction will be for stateful anamorphic extensions, and in order to achieve reasonable guarantees it requires that the covert message space of the base anamorphic extension be the randomness space of the underlying PKE scheme. For this reason, the rejection sampling technique from [PPY22] seems not to be suitable, since it is efficient only when transmitting at most logarithmically many covert bits in the security parameter.

Construction 4. *Let Π be a PKE scheme with randomness space $\mathcal{R}$, with $\rho \doteq |\mathcal{R}|$, let $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ be a (non-robust) stateful anamorphic extension for Π with covert message space $\mathcal{R}$. For covert message space $\hat{\mathcal{M}} \subseteq \mathcal{R}$, with $\ell \doteq |\hat{\mathcal{M}}| < \rho$, define the stateful anamorphic extension $\Sigma_4 \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec}')$, where on input a double key dk, a ciphertext c, and a state $\mathtt{st}$, aDec' first computes $(\hat{m}, \mathtt{st}') := \mathsf{aDec}(dk, c; \mathtt{st})$, and then outputs $(\hat{m}, \mathtt{st}')$ if $\hat{m} \in \hat{\mathcal{M}}'$, and $\perp$ otherwise.*

Note that Σ_4 trivially satisfies perfect correctness. Moreover, note that the security of Σ_4 is trivially inherited by the security of the underlying anamorphic extension Σ. Regarding robustness, it is also easy to see that there is an acceptable degradation if ℓ is small (that is, $\ell \ll \rho$). We state the following lemma without proof (the intuition is that once we substitute the ciphertext c by a random string in $\mathsf{G}^{\text{rob-0}}_{\Pi,\Sigma_4}$, due to ind\$-cpa, we then have a chance $\frac{\ell}{\rho}$ for each query to provoke the bad event that $\hat{m} \neq \perp$, which makes distinguishing $\mathsf{G}^{\text{rob-1}}_{\Pi,\Sigma_4}$ trivial).

Lemma 4.9. *Let Σ_4 be the anamorphic extension from Construction 4 for a PKE scheme Π with (non-robust) anamorphic extension Σ. There exists an efficient transformation of any* rob *distinguisher* D *into a* prf *distinguisher* D' *with* $q \doteq q(\mathrm{D}') = q(\mathrm{D})$ *such that*

$$\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_4}(\mathrm{D}) \leq \mathbf{Adv}^{\mathsf{ind\$\text{-}cpa}}_{\Pi,\Sigma}(\mathrm{D}') + \frac{q\ell}{\rho}.$$

5 Concrete Instantiations of the Generic Constructions

In this section we show concrete instantiations of our generic constructions Σ_2 (and the related Σ_2'), Σ_3 (and the related Σ_3'), and Σ_4 from Section 4 for *well-established* PKE schemes, thus showing *practical* anamorphic modes are indeed possible. All proofs are deferred to ??.

5.1 Instantiations of Σ_2: ElGamal and Cramer-Shoup

Synchronized Robustly Anamorphic ElGamal. We now show that the classic ElGamal PKE scheme admits an anamorphic extension since it is SRR. First, let recall the conventional specification of the ElGamal PKE scheme [ElG85].

Construction 5. *Let* $\mathbb{G}$ *be a cyclic group of prime order* q *with generator* g*, and let the public parameter be* $\mathsf{pp} = (\mathbb{G}, q, g)$*. Then the* ElGamal PKE scheme *is defined as the tuple* $\mathsf{ElGamal} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$*, where:*

- Gen*: sample* $sk \xleftarrow{\$} \mathbb{Z}_q$*, set* $pk := g^{sk}$*, and output the key-pair* (sk, pk)*.*[7]
- Enc*: on input a public key* pk *and a message* $m \in \mathbb{G}$*, sample* $r \xleftarrow{\$} \mathbb{Z}_q$*, set* $c_1 := m \cdot pk^r$*,* $c_2 := g^r$*, and output the ciphertext* (c_1, c_2)*.*
- Dec*: on input a secret key* sk *and a ciphertext* (c_1, c_2)*, output* $c_1 \cdot c_2^{-sk}$*.*

Lemma 5.1. *The* ElGamal *PKE scheme is SRR.*

Proof. We prove each item from Definition 4.1:

1. $\langle \mathbb{Z}_q; \oplus \rangle$, where $\oplus$ denotes addition modulo q, is clearly a group.
2. With $\alpha(a, b, c) := b \cdot a^c$ and $\beta(a) := g^a$, we have that for public key pk, message m, and randomness r, $\mathsf{Enc}(pk, m; r) = (\alpha(pk, m, r), \beta(r))$. Moreover, both α and β are clearly injective.
3. With $\gamma(a, b) := a \cdot g^{-b}$, we have that for any $a, b \in \mathbb{Z}_q$,

$$\gamma(\beta(a \oplus b), b) = \gamma(g^{a \oplus b}, b) = g^{a \oplus b} \cdot g^{-b} = g^a = \beta(a). \qquad \square$$

Putting things together, for completeness we finally describe the resulting synchronized anamorphic extension SyncAnamElGamal for ElGamal with pre-computation.

Construction 6. *For covert message space* $\hat{\mathcal{M}} \subseteq \mathbb{Z}_q$*, with* $\ell \doteq |\hat{\mathcal{M}}|$*, and* F *a function* $\mathcal{K} \times [\tau] \rightarrow \mathbb{Z}_q$*, let the anamorphic extension* $\mathsf{SyncAnamElGamal} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *for the* ElGamal *PKE scheme from Construction 5 with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, \tau, \mathsf{F})$ *be defined as in Figure 11.*

---

[7] Recall that, even if we did not explicitate it here, we assume that pp can be obtained from both sk and pk.

```
$\mathsf{aGen}(sk, pk)$:
$K \xleftarrow{\$} \mathcal{K}$
$\mathbf{T} := []$
foreach $\hat{m} \in \hat{\mathcal{M}}$ do
    $\mathbf{T}[g^{\hat{m}}] := \hat{m}$
$dk := (K, \mathbf{T}, pk)$
return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m}; \mathtt{ctr})$:
$t := F(K, \mathtt{ctr})$
$r := \hat{m} \oplus t$
$c_1 := m \cdot pk^r$
$c_2 := g^r$
$c := (c_1, c_2)$
return $(c; \mathtt{ctr} + 1)$
```

```
$\mathsf{aDec}(dk, (c_1, c_2); \mathtt{ctr})$:
$t := F(K, \mathtt{ctr})$
$s := c_2 \cdot g^{-t}$
$\hat{m} := \mathbf{T}[s]$
return $(\hat{m}; \mathtt{ctr} + 1)$
```

Figure 11: Synchronized robustly anamorphic extension SyncAnamElGamal $\doteq$ (aGen, aEnc, aDec) with pre-computation for the SRR PKE scheme ElGamal.

Synchronized Robustly Anamorphic Cramer-Shoup. We now show that also the classic Cramer-Shoup PKE scheme admits an anamorphic extension since it is SRR as well. First, let recall the conventional specification of the Cramer-Shoup PKE scheme [CS98]. Note that in the following definition, we tailor the syntactic description to exactly match our notion of SRR PKE.

Construction 7. *Let* $\mathbb{G}$ *be a cyclic group of prime order* q *with generators* g_1 *and* g_2*, let* $H : \mathbb{G}^3 \to \mathbb{Z}_q$ *be a hash function, and let the public parameter be* $\mathsf{pp} = (\mathbb{G}, q, g_1, g_2, H)$*. Then the* Cramer-Shoup PKE scheme *is defined as the tuple* CramerShoup = (Gen, Enc, Dec)*, where:*

- Gen*: sample* $x_1, x_2, y_1, y_2, z \xleftarrow{\$} \mathbb{Z}_q$*, set* $c := g_1^{x_1} g_2^{x_2}$*,* $d := g_1^{y_1} g_2^{y_2}$*,* $e := g_1^z$*,* $sk := (x_1, x_2, y_1, y_2, z)$*,* $pk := (c, d, e)$ *and output the key-pair* (sk, pk)*.*
- Enc*: on input a public key* $pk = (c, d, e)$ *and a message* $m \in \mathbb{G}$*, sample* $r \xleftarrow{\$} \mathbb{Z}_q$*, set* $u_1 := g_1^r$*,* $u_2 := g_2^r$*,* $v := m \cdot e^r$*,* $h := H(u_1, u_2, v)$*,* $w := c^r d^{rh}$*, and output the ciphertext* $c := ((v, w), (u_1, u_2))$*.*
- Dec*: on input a secret key* sk *and a ciphertext* $((v, w), (u_1, u_2))$*, compute* $h := H(u_1, u_2, v)$*, and if* $u_1^{x_1 + y_1 h} u_2^{x_2 + y_2 h} = w$*, then output* $v \cdot u_1^{-z}$*; otherwise, output the special symbol* $\perp$*.*

Lemma 5.2. *The* CramerShoup *PKE scheme is SRR.*

Proof. We prove each item from Definition 4.1:

1. $\langle \mathbb{Z}_q; \oplus \rangle$, where $\oplus$ denotes addition modulo q, is clearly a group.
2. With $\alpha((a_1, a_2, a_3), b, c) := (b \cdot a_3^c, a_1^c a_2^{c \cdot H(g_1^c, g_2^c, b \cdot a_3^c)})$ and $\beta(a) := (g_1^a, g_2^a)$, we have that for public key pk, message m, and randomness r, $\mathsf{Enc}(pk, m; r) = (\alpha(pk, m, r), \beta(r))$.
3. With $\gamma((a_1, a_2), b) := (a_1 \cdot g_1^{-b}, a_2 \cdot g_2^{-b})$, we have that for any $a, b \in \mathbb{Z}_q$,

$$\begin{aligned} \gamma(\beta(a \oplus b), b) &= \gamma((g_1^{a \oplus b}, g_2^{a \oplus b}), b) \\ &= (g_1^{a \oplus b} \cdot g_1^{-b}, g_2^{a \oplus b} \cdot g_2^{-b}) \\ &= (g_1^a, g_2^a) \\ &= \beta(a). \end{aligned}$$

□

```
$\mathsf{aGen}((c, d, e))$:
    $K \xleftarrow{\$} \mathcal{K}$
    $\mathbf{T} := []$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        $\mathbf{T}[(g_1^{\hat{m}}, g_2^{\hat{m}})] := \hat{m}$
    $dk := (K, \mathbf{T}, (c, d, e))$
    return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m}; \mathsf{ctr})$:
    $t := F(K, \mathsf{ctr})$
    $r := \hat{m} \oplus t$
    $u_1 := g_1^r$
    $u_2 := g_2^r$
    $v := m \cdot e^r$
    $h := H(u_1, u_2, v)$
    $w := c^r d^{rh}$
    $c := ((v, w), (u_1, u_2))$
    return $(c; \mathsf{ctr} + 1)$
```

```
$\mathsf{aDec}(dk, (c_1, (c_{2,1}, c_{2,2})); \mathsf{ctr})$:
    $t := F(K, \mathsf{ctr})$
    $s := (c_{2,1} \cdot g_1^{-t}, c_{2,2} \cdot g_2^{-t})$
    $\hat{m} := \mathbf{T}[s]$
    return $(\hat{m}; \mathsf{ctr} + 1)$
```

Figure 12: Synchronized robustly anamorphic extension $\mathsf{SyncAnamCramerShoup} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ with pre-computation for the SRR PKE scheme $\mathsf{CramerShoup}$.

Putting things together, for completeness we finally describe the resulting synchronized anamorphic extension $\mathsf{SyncAnamCramerShoup}$ for $\mathsf{CramerShoup}$.

Construction 8. *For covert message space $\hat{\mathcal{M}} \subseteq \mathbb{Z}_q$, with $\ell \doteq |\hat{\mathcal{M}}|$, and F a function $\mathcal{K} \times [\tau] \to \mathbb{Z}_q$, let the anamorphic extension $\mathsf{SyncAnamCramerShoup} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ for the $\mathsf{CramerShoup}$ PKE scheme from Construction 5 with anamorphic parameters $\mathsf{ap} = (\mathcal{K}, \tau, F)$ be defined as in Figure 11.*

5.2 Instantiations of Σ_3: ElGamal and Cramer-Shoup

Unsynchronized Robustly Anamorphic ElGamal. We now show how to obtain an even better anamorphic extension for ElGamal by defining a simple function δ satisfying (1). For this, we need to instantiate the underlying group $\mathbb{G}$. Concretely, consider the case of $\mathbb{G} = \mathbb{Z}_p^*$, for p prime, with order $q = p - 1$, and let $\delta(x) \doteq R_\tau(x) + 1 \in [\tau]$, where $R_\tau(\cdot)$ denotes the remainder modulo τ.

Lemma 5.3. *For p prime, g a generator of $\mathbb{Z}_p^*$, $q \doteq p - 1$, $\tau \leq q$, and $y \in [\tau]$:*

$$|\{r \in \mathbb{Z}_q \mid R_\tau(g^r) + 1 = y\}| \geq \left\lfloor \frac{q}{\tau} \right\rfloor.$$

Proof. Note that $R_\tau(g^r) + 1 = y$ is true if and only if $g^r - y + 1 = k\tau$ for some integer k. More precisely, since $k\tau + y - 1 = g^r \in \mathbb{Z}_p^*$, which implies $1 \leq k\tau + y - 1 \leq q$, we have that $k \in \mathfrak{K} \doteq \{(2 - y)/\tau, \dots, (q - y + 1)/\tau\}$. Therefore,

$$\begin{aligned} |\{r \in \mathbb{Z}_q \mid R_\tau(g^r) + 1 = y\}| &= \left|\{c \in \mathbb{Z}_p^* \mid R_\tau(c) + 1 = y\}\right| \\ &= \left|\{c \in \mathbb{Z}_p^* \,\middle|\, \exists k \in \mathfrak{K} : c = k\tau + y - 1\}\right| \\ &= |\mathfrak{K}| = \frac{q - y + 1}{\tau} - \frac{2 - y}{\tau} + 1 \\ &= \frac{q + \tau - 1}{\tau} \geq \frac{q}{\tau} \geq \left\lfloor \frac{q}{\tau} \right\rfloor, \end{aligned}$$

since the mapping $k \mapsto k\tau + y - 1$ is injective, and $\tau \geq 1$. □

```
$\mathsf{aGen}(sk, pk)$:
$K \xleftarrow{\$} \mathcal{K}$
$\mathbf{T} := []$
foreach $\hat{m} \in \hat{\mathcal{M}}$ do
    $\mathbf{T}[g^{\hat{m}}] := \hat{m}$
$dk := (K, \mathbf{T}, pk)$
return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m})$:
repeat
    $x \xleftarrow{\$} [\sigma]$
    $y \xleftarrow{\$} [\tau]$
    $t := F(K, (x, y))$
    $r := \hat{m} \oplus t$
until $R_\tau(g^r) = y$
$c_1 := m \cdot pk^r$
$c_2 := g^r$
$c := (c_1, c_2)$
return $c$
```

```
$\mathsf{aDec}(dk, (c_1, c_2))$:
$y := R_\tau(c_2)$
foreach $x \in [\sigma]$ do
    $t := F(K, (x, y))$
    $s := c_2 \cdot g^{-t}$
    $\hat{m} := \mathbf{T}[s]$
    if $\hat{m} \neq \perp$ then
        return $\hat{m}$
return $\perp$
```

Figure 13: Synchronized robustly anamorphic extension $\mathsf{AnamElGamal} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ with pre-computation for the SRR PKE scheme ElGamal.

Putting things together, for completeness we finally describe the resulting stateless unsynchronized anamorphic extension AnamElGamal for ElGamal with pre-computation.

Construction 9. *For covert messages set* $\hat{\mathcal{M}} \subseteq \mathbb{Z}_q$*, with* $\ell \doteq |\hat{\mathcal{M}}|$*, and* F *a function* $\mathcal{K} \times ([\sigma] \times [\tau]) \to \mathbb{Z}_q$*, let the anamorphic extension* $\mathsf{AnamElGamal} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *for the* ElGamal *PKE scheme from Construction 5 with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, R_\tau, \sigma, \tau, F)$ *be defined as in Figure 13.*

In Appendix C we provide a test implementation of AnamElGamal. Note that it is possible to similarly show that ElGamal instantiated over an elliptic curve, such as Curve25519, also admits a robust anamorphic extension. More concretely, with $\mathbb{G}$ a subgroup of $E/\mathbb{F}_p$ of prime order $q \approx p/8$, for $p = 2^{255} - 19$, we can define δ as in Lemma 5.3, and apply it to the x coordinate of the points.

Unsynchronized Robustly Anamorphic Cramer-Shoup. We now show how to obtain an even better anamorphic extension also for Cramer-Shoup by defining a simple function δ satisfying (1). For this, we need to again instantiate the underlying group $\mathbb{G}$ as $\mathbb{G} = \mathbb{Z}_p^*$. With $\delta((x, y)) \doteq R_\tau(x) + 1$, we can then reuse Lemma 5.3. Putting things together, for completeness we finally describe the resulting stateless unsynchronized anamorphic extension AnamCramerShoup for CramerShoup with pre-computation. Note that we also slightly deviate from the specification of Figure 10 by only using the first of the two group elements in $\beta(\hat{m})$ as index to the look-up table $\mathbf{T}$.

Construction 10. *For covert messages set* $\hat{\mathcal{M}} \subseteq \mathbb{Z}_q$*, with* $\ell \doteq |\hat{\mathcal{M}}|$*, and* F *a function* $\mathcal{K} \times ([\sigma] \times [\tau]) \to \mathbb{Z}_q$*, let the anamorphic extension* $\mathsf{AnamCramerShoup} \doteq (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ *for the* CramerShoup *PKE scheme from Construction 7 with anamorphic parameters* $\mathsf{ap} = (\mathcal{K}, R_\tau, \sigma, \tau, F)$ *be defined as in Figure 14.*

```
$\mathsf{aGen}(sk, pk)$:
$K \xleftarrow{\$} \mathcal{K}$
$\mathbf{T} := []$
foreach $\hat{m} \in \hat{\mathcal{M}}$ do
    $\mathbf{T}[g_1^{\hat{m}}] := \hat{m}$
$dk := (K, \mathbf{T}, pk)$
return $dk$
```

```
$\mathsf{aEnc}(dk, m, \hat{m})$:
repeat
    $x \xleftarrow{\$} [\sigma]$
    $y \xleftarrow{\$} [\tau]$
    $t := F(K, (x, y))$
    $r := \hat{m} \oplus t$
until $R_\tau(g_1^r) = y$
$u_1 := g_1^r$
$u_2 := g_2^r$
$v := m \cdot e^r$
$h := H(u_1, u_2, v)$
$w := c^r d^{rh}$
$c := ((v, w), (u_1, u_2))$
return $c$
```

```
$\mathsf{aDec}(dk, (c_1, (c_{2,1}, c_{2,2})))$:
$y := R_\tau(c_{2,1})$
foreach $x \in [\sigma]$ do
    $t := F(K, (x, y))$
    $s := c_{2,1} \cdot g_1^{-t}$
    $\hat{m} := \mathbf{T}[s]$
    if $\hat{m} \neq \bot$ then
        return $\hat{m}$
return $\bot$
```

Figure 14: Synchronized robustly anamorphic extension AnamCramerShoup $\doteq$ (aGen, aEnc, aDec) with pre-computation for the SRR PKE scheme CramerShoup.

5.3 Instantiation of Σ_4: RSA-OAEP

We begin by recalling the OAEP technique from [BR95]. Let $f : \{0,1\}^n \rightarrow \{0,1\}^n$ be a trapdoor permutation and f^{-1} its inverse. With $l < n$, let $G : \{0,1\}^t \rightarrow \{0,1\}^{n-l}$ and $H : \{0,1\}^{n-l} \rightarrow \{0,1\}^t$ be *random oracles*. We can now construct an IND-CPA PKE scheme by first padding each messages $m \in \{0,1\}^{n-l}$ and then applying the trapdoor permutation as follows: Choose a uniformly random value $r \xleftarrow{\$} \{0,1\}^l$, and then output the ciphertext $c := f(m \oplus G(r) \| r \oplus H(m \oplus G(r)))$. Because of the Feistel-network-like structure of the OAEP, the ciphertext c can then be easily decrypted as follows: First obtain $a \| b := f^{-1}(c)$, and then recompute the randomness $r = b \oplus H(a)$, and finally output the original plaintext $m = a \oplus G(r)$.

An interesting property of OAEP, is that it is (fully) *randomness recoverable* [LW10], meaning that given an encryption c of a message m generated using randomness r, from c and the the secret key *sk* alone, it is possible to fully recover r. This naturally lends itself to a *synchronized* anamorphic scheme as follows: Just like in our first construction Σ_1, use a counter $\mathtt{ctr}$ and a PRF F to generate a one-time pad $t := F(K, \mathtt{ctr}) \in \{0,1\}^l$, and set the randomness to $r := \hat{m} \oplus t$. Being randomness recoverable, OAEP then allows to efficiently retrieve $\hat{m}$ simply by first recovering the randomness r, and then computing $\hat{m} := r \oplus F(K, \mathtt{ctr})$.

Now, by the discussion in Section 4.5, we have that this anamorphic extension for OAEP can be trivially made robust by choosing a small enough $l' < l$ and instantiating Σ_4 with $\hat{\mathcal{M}} \doteq \{0,1\}^{l'}$. Therefore, when f denotes the RSA trapdoor permutation $f(x) := R_N(x^e)$, where $N = pq$ for two primes p, q, and $e \in \mathbb{Z}_{\phi(n)}$, we have that RSA-OAEP indeed admits a robustly anamorphic extension.

References

[AAB+97] Hal Abelson, Ross Anderson, Steven M Bellovin, Josh Benaloh, Matt Blaze, Whitfeld Diffie, John Gilmore, Peter G Neumann, Ronald L Rivest, Jeffrey I Schiller, et al. The risks of key recovery, key escrow, and trusted third-party encryption. 1997.

[AAB+15] Harold Abelson, R Anderson, M Bellovin, J Benaloh, M Blaze, W Diffie, J Gilmore, M Green, PG Neumann, S Landau, et al. Keys under doormats: Mandating insecurity by requiring government access to all data and communications. july 6, 2015. *Google Scholar Google Scholar Digital Library Digital Library*, 2015.

[ABN10] Michel Abdalla, Mihir Bellare, and Gregory Neven. Robust encryption. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 480–497. Springer, Heidelberg, February 2010. doi:10.1007/978-3-642-11799-2_28.

[BBDP01] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 566–582. Springer, Heidelberg, December 2001. doi:10.1007/3-540-45682-1_33.

[BDJR97] Mihir Bellare, Anand Desai, Eric Jokipii, and Phillip Rogaway. A concrete security treatment of symmetric encryption. In *38th FOCS*, pages 394–403. IEEE Computer Society Press, October 1997. doi:10.1109/SFCS.1997.646128.

[BGH+23] Fabio Banfi, Konstantin Gegier, Martin Hirt, Ueli Maurer, and Guilherme Rito. Anamorphic encryption, revisited. Cryptology ePrint Archive, Report 2023/249, 2023. https://eprint.iacr.org/2023/249.

[BKR94] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The security of cipher block chaining. In Yvo Desmedt, editor, *CRYPTO'94*, volume 839 of *LNCS*, pages 341–358. Springer, Heidelberg, August 1994. doi:10.1007/3-540-48658-5_32.

[Bla94] Matt Blaze. Protocol failure in the escrowed encryption standard. In Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, and Ravi S. Sandhu, editors, *ACM CCS 94*, pages 59–67. ACM Press, November 1994. doi:10.1145/191177.191193.

[BPR14] 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. Springer, Heidelberg, August 2014. doi:10.1007/978-3-662-44371-2_1.

[BR95] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In Alfredo De Santis, editor, *EUROCRYPT'94*, volume 950 of *LNCS*, pages 92–111. Springer, Heidelberg, May 1995. doi:10.1007/BFb0053428.

[BR06] Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426. Springer, Heidelberg, May / June 2006. `doi:10.1007/11761679_25`.

[CDNO97] Ran Canetti, Cynthia Dwork, Moni Naor, and Rafail Ostrovsky. Deniable encryption. In Burton S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 90–104. Springer, Heidelberg, August 1997. `doi:10.1007/BFb0052229`.

[CGM24] Dario Catalano, Emanuele Giunta, and Francesco Migliaro. Anamorphic encryption: New constructions and homomorphic realizations. In Thyla van der Merwe and Julia Hesse, editors, *EUROCRYPT 2024, Part ???*, volume ??? of *LNCS*, pages ???–??? Springer Nature, May 2024. `doi:10.1007/???`

[CNE$^+$14] 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, *USENIX Security 2014*, pages 319–335. USENIX Association, August 2014.

[CS98] Ronald Cramer and Victor Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In Hugo Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 13–25. Springer, Heidelberg, August 1998. `doi:10.1007/BFb0055717`.

[Dak96] Howard S Dakoff. The clipper chip proposal: Deciphering the unfounded fears that are wrongfully derailing its implementation, 29 j. marshall l. rev. 475 (1996). *UIC Law Review*, 29(2):8, 1996.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472, 1985. `doi:10.1109/TIT.1985.1057074`.

[FY95] Yair Frankel and Moti Yung. Escrow encryption systems visited: Attacks, analysis and designs. In Don Coppersmith, editor, *CRYPTO'95*, volume 963 of *LNCS*, pages 222–235. Springer, Heidelberg, August 1995. `doi:10.1007/3-540-44750-4_18`.

[GJJS04] Philippe Golle, Markus Jakobsson, Ari Juels, and Paul F. Syverson. Universal re-encryption for mixnets. In Tatsuaki Okamoto, editor, *CT-RSA 2004*, volume 2964 of *LNCS*, pages 163–178. Springer, Heidelberg, February 2004. `doi:10.1007/978-3-540-24660-2_14`.

[GKL21] Matthew Green, Gabriel Kaptchuk, and Gijs Van Laer. Abuse resistant law enforcement access systems. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part III*, volume 12698 of *LNCS*, pages 553–583. Springer, Heidelberg, October 2021. `doi:10.1007/978-3-030-77883-5_19`.

[HPRV19] Thibaut Horel, Sunoo Park, Silas Richelson, and Vinod Vaikuntanathan. How to subvert backdoored encryption: Security against adversaries that decrypt all ciphertexts. In

Avrim Blum, editor, *ITCS 2019*, volume 124, pages 42:1–42:20. LIPIcs, January 2019. doi:10.4230/LIPIcs.ITCS.2019.42.

[KMO+13] Markulf Kohlweiss, Ueli Maurer, Cristina Onete, Björn Tackmann, and Daniele Venturi. Anonymity-preserving public-key encryption: A constructive approach. In Emiliano De Cristofaro and Matthew K. Wright, editors, *PETS 2013*, volume 7981 of *LNCS*, pages 19–39. Springer, Heidelberg, July 2013. doi:10.1007/978-3-642-39077-7_2.

[KPP+23a] Mirek Kutylowski, Giuseppe Persiano, Duong Hieu Phan, Moti Yung, and Marcin Zawada. The self-anti-censorship nature of encryption: On the prevalence of anamorphic cryptography. *PoPETs*, 2023(4):170–183, July 2023. doi:10.56553/popets-2023-0104.

[KPP+23b] Miroslaw Kutylowski, Giuseppe Persiano, Duong Hieu Phan, Moti Yung, and Marcin Zawada. Anamorphic signatures: Secrecy from a dictator who only permits authentication! In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 759–790. Springer, Heidelberg, August 2023. doi:10.1007/978-3-031-38545-2_25.

[LW10] Chung Ki Li and Duncan S. Wong. Signcryption from randomness recoverable public key encryption. *Information Sciences*, 180(4):549–559, 2010. URL: https://www.sciencedirect.com/science/article/pii/S0020025509004514, doi:10.1016/j.ins.2009.10.015.

[Mic93] Silvio Micali. Fair public-key cryptosystems. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 113–138. Springer, Heidelberg, August 1993. doi:10.1007/3-540-48071-4_9.

[PPY22] Giuseppe Persiano, Duong Hieu Phan, and Moti Yung. Anamorphic encryption: Private communication against a dictator. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part II*, volume 13276 of *LNCS*, pages 34–63. Springer, Heidelberg, May / June 2022. doi:10.1007/978-3-031-07085-3_2.

[R+98] Ronald L Rivest et al. Chaffing and winnowing: Confidentiality without encryption. *CryptoBytes (RSA laboratories)*, 4(1):12–17, 1998.

[RTYZ16] 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. Springer, Heidelberg, December 2016. doi:10.1007/978-3-662-53890-6_2.

[RTYZ17] 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. ACM Press, October / November 2017. doi:10.1145/3133956.3133993.

[Sim83] Gustavus J. Simmons. The prisoners' problem and the subliminal channel. In David Chaum, editor, *CRYPTO'83*, pages 51–67. Plenum Press, New York, USA, 1983.

[vH04] Luis von Ahn and Nicholas J. Hopper. Public-key steganography. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 323–341. Springer, Heidelberg, May 2004. `doi:10.1007/978-3-540-24676-3_20`.

[WCHY23] 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. Springer, Heidelberg, December 2023. `doi:10.1007/978-981-99-8736-8_5`.

[YY96] 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. Springer, Heidelberg, August 1996. `doi:10.1007/3-540-68697-5_8`.

[YY97] 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. Springer, Heidelberg, August 1997. `doi:10.1007/BFb0052241`.

[YY98] Adam Young and Moti Yung. Auto-recoverable auto-certifiable cryptosystems. In Kaisa Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 17–31. Springer, Heidelberg, May / June 1998. `doi:10.1007/BFb0054114`.

[YY10] Adam Young and Moti Yung. Kleptography from standard assumptions and applications. In Juan A. Garay and Roberto De Prisco, editors, *SCN 10*, volume 6280 of *LNCS*, pages 271–290. Springer, Heidelberg, September 2010. `doi:10.1007/978-3-642-15317-4_18`.

[YY18] Adam L. Young and Moti Yung. Semantically secure anonymity: Foundations of re-encryption. In Dario Catalano and Roberto De Prisco, editors, *SCN 18*, volume 11035 of *LNCS*, pages 255–273. Springer, Heidelberg, September 2018. `doi:10.1007/978-3-319-98113-0_14`.

```
$\boxed{\mathsf{G}^{\mathsf{sec\text{-}0}}_{\Pi,\Sigma_1}}$ $\dbox{\mathsf{G}^{1}_{\Pi,\Sigma_1}}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\boxed{\mathsf{K} \xleftarrow{\$} \mathcal{K}}$
    $\dbox{\mathsf{f} \xleftarrow{\$} \mathcal{R}^{[\tau] \times \hat{\mathcal{M}}}}$
    $\mathtt{ctr} := 0$
    return $(sk, pk)$
  $\textsc{AEnc}(\mathsf{m}, \hat{\mathsf{m}})$:
    $\boxed{\mathsf{r} := \mathsf{F}(\mathsf{K}, (\mathtt{ctr}, \hat{\mathsf{m}}))}$
    $\dbox{\mathsf{r} := \mathsf{f}((\mathtt{ctr}, \hat{\mathsf{m}}))}$
    $\mathsf{c} := \mathsf{Enc}(pk, \mathsf{m}; \mathsf{r})$
    $\mathtt{ctr} := \mathtt{ctr} + 1$
    return $\mathsf{c}$

$\mathsf{G}^{2}_{\Pi,\Sigma_1}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
  $\textsc{AEnc}(\mathsf{m}, \hat{\mathsf{m}})$:
    $\mathsf{r} \xleftarrow{\$} \mathcal{R}$
    $\mathsf{c} \leftarrow \mathsf{Enc}(pk, \mathsf{m}; \mathsf{r})$
    return $\mathsf{c}$

$\mathsf{G}^{\mathsf{sec\text{-}1}}_{\Pi,\Sigma_1}$
  $\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
  $\textsc{AEnc}(\mathsf{m}, \hat{\mathsf{m}})$:
    $\mathsf{c} \leftarrow \mathsf{Enc}(pk, \mathsf{m})$
    return $\mathsf{c}$

$\mathrm{D}'^{\textsc{Init},\textsc{Eval}}$
    $\mathsf{b} \leftarrow \mathrm{D}^{\textsc{Init}^*,\textsc{AEnc}^*}$
    return $\mathsf{b}$
  $\textsc{Init}^*()$:
    $\textsc{Init}()$
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\mathtt{ctr} := 0$
    return $(sk, pk)$
  $\textsc{AEnc}^*(\mathsf{m}, \hat{\mathsf{m}})$:
    $\mathsf{r} := \textsc{Eval}((\mathtt{ctr}, \hat{\mathsf{m}}))$
    $\mathsf{c} := \mathsf{Enc}(pk, \mathsf{m}; \mathsf{r})$
    $\mathtt{ctr} := \mathtt{ctr} + 1$
    return $\mathsf{c}$
```

Figure 15: Games $\mathrm{G}^{\mathsf{sec\text{-}0}}_{\Pi,\Sigma_1}$, $\mathrm{G}^{1}_{\Pi,\Sigma_1}$, $\mathrm{G}^{2}_{\Pi,\Sigma_1}$, $\mathrm{G}^{\mathsf{sec\text{-}1}}_{\Pi,\Sigma_1}$, and distinguisher D' for the proof of Lemma 4.1.

A Proofs

A.1 Proofs for the Σ_1 Construction

Lemma 4.1. *Let Σ_1 be the anamorphic extension from Construction 1 for an arbitrary PKE scheme Π. There exists an efficient transformation of any* sec *distinguisher D into a* prf *distinguisher D' with $\mathrm{q}(\mathrm{D}') = \mathrm{q}(\mathrm{D}) \leq \tau$ such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_1}(\mathrm{D}) = \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathrm{D}').$$

Proof. Define games $\mathrm{G}^{\mathsf{sec\text{-}0}}_{\Pi,\Sigma_1}$, $\mathrm{G}^{1}_{\Pi,\Sigma_1}$, $\mathrm{G}^{2}_{\Pi,\Sigma_1}$, $\mathrm{G}^{\mathsf{sec\text{-}1}}_{\Pi,\Sigma_1}$, and distinguisher D' as in Figure 15. D' is such that if it is interacting with $\mathrm{G}^{\mathsf{prf\text{-}0}}_{\mathsf{F}}$, it perfectly emulates $\mathrm{G}^{\mathsf{sec\text{-}0}}_{\Pi,\Sigma_1}$ towards D, and if it is interacting with $\mathrm{G}^{\mathsf{prf\text{-}1}}_{\mathsf{F}}$, it perfectly emulates $\mathrm{G}^{1}_{\Pi,\Sigma_1}$ towards D. Since the counter $\mathtt{ctr}$ used as part of the input to the uniform random function f is never repeating, r is effectively uniformly distributed in $\mathrm{G}^{1}_{\Pi,\Sigma_1}$, and therefore $\mathrm{G}^{1}_{\Pi,\Sigma_1}$ is perfectly indistinguishable from $\mathrm{G}^{2}_{\Pi,\Sigma_1}$. Moreover, $\mathrm{G}^{2}_{\Pi,\Sigma_1}$ is just a more explicit

```
$\boxed{G^{\text{rob-0}}_{\Pi,\Sigma_1}}$ $\boxed{G^{1}_{\Pi,\Sigma_1}}$
INIT():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\boxed{K \xleftarrow{\$} \mathcal{K}}$
    $\boxed{f \xleftarrow{\$} \mathcal{R}^{[\tau]\times\hat{\mathcal{M}}}}$
ENCADEC(m, ctr):
    $c := \mathsf{Enc}(pk, m)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        $\boxed{r' := F(K, (ctr, \hat{m}))}$
        $\boxed{r' := f((ctr, \hat{m}))}$
        $c' := \mathsf{Enc}(pk, m; r')$
        if $c' = c$ then
            return $\hat{m}$
    return $\perp$
```

```
$G^{2}_{\Pi,\Sigma_1}$
INIT():
    bad := false
ENCADEC(m, ctr):
    $r \xleftarrow{\$} \mathcal{R}$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        $r' \xleftarrow{\$} \mathcal{R}$
        if $r' = r$ then
            bad := true
            return $\hat{m}$
    return $\perp$

$G^{\text{rob-1}}_{\Pi,\Sigma_1}$
INIT():
    // Do nothing
ENCADEC(m, ctr):
    return $\perp$
```

```
$D'^{\text{INIT},\text{EVAL}}$
    $b \leftarrow D^{\text{INIT}^*,\text{ENCADEC}^*}$
    return b
INIT*():
    INIT()
    $(sk, pk) \leftarrow \mathsf{Gen}()$
ENCADEC*(m, ctr):
    $c := \mathsf{Enc}(pk, m)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        $r := \text{EVAL}((ctr, \hat{m}))$
        $c' := \mathsf{Enc}(pk, m; r')$
        if $c' = c$ then
            return $\hat{m}$
    return $\perp$
```

Figure 16: Games $G^{\text{rob-0}}_{\Pi,\Sigma_1}$, G^{1}_{Π,Σ_1}, G^{2}_{Π,Σ_1}, $G^{\text{rob-1}}_{\Pi,\Sigma_1}$, and distinguisher D' for the proof of Lemma 4.2.

description of $G^{\text{sec-1}}_{\Pi,\Sigma_1}$, and thus they too are perfectly indistinguishable. Therefore, we have

$$\begin{aligned}
\mathbf{Adv}^{\text{sec}}_{\Pi,\Sigma_1}(D) &= \Pr[G^{\text{sec-0}}_{\Pi,\Sigma_1}(D)] - \Pr[G^{\text{sec-1}}_{\Pi,\Sigma_1}(D)] \\
&= (\Pr[G^{\text{sec-0}}_{\Pi,\Sigma_1}(D)] - \Pr[G^{1}_{\Pi,\Sigma_1}(D)]) \\
&\quad + (\Pr[G^{1}_{\Pi,\Sigma_1}(D)] - \Pr[G^{2}_{\Pi,\Sigma_1}(D)]) \\
&\quad + (\Pr[G^{2}_{\Pi,\Sigma_1}(D)] - \Pr[G^{\text{sec-1}}_{\Pi,\Sigma_1}(D)]) \\
&= (\Pr[G^{\text{prf-0}}_{F}(D')] - \Pr[G^{\text{prf-1}}_{F}(D')]) + 0 + 0 \\
&= \mathbf{Adv}^{\text{prf}}_{F}(D').
\end{aligned}$$

□

Lemma 4.2. *Let Σ_1 be the anamorphic extension from Construction 1 for an arbitrary PKE scheme Π. There exists an efficient transformation of any* rob *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \tau$ such that*

$$\mathbf{Adv}^{\text{rob}}_{\Pi,\Sigma_1}(D) \leq \mathbf{Adv}^{\text{prf}}_{F}(D') + \frac{q\ell}{\rho}.$$

Proof. Define games $G^{\text{rob-0}}_{\Pi,\Sigma_1}$, G^{1}_{Π,Σ_1}, G^{2}_{Π,Σ_1}, $G^{\text{rob-1}}_{\Pi,\Sigma_1}$, and distinguisher D' as in Figure 16. D' is such that if it is interacting with $G^{\text{prf-0}}_{F}$, it perfectly emulates $G^{\text{rob-0}}_{\Pi,\Sigma_1}$ towards D, and if it is interacting with $\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}$, it perfectly emulates $\mathsf{G}^{1}_{\Pi,\Sigma_1}$ towards D. Without loss of generality,[8] we can assume that D never repeats counters, hence r' is effectively uniformly distributed in $\mathsf{G}^{1}_{\Pi,\Sigma_1}$. Then, since $c := \mathsf{Enc}(pk, m)$ corresponds to $r \xleftarrow{\$} \mathcal{R}$ followed by $c := \mathsf{Enc}(pk, m; r)$, and since for any pk, m, r, and r' we have that $\mathsf{Enc}(pk, m; r) \neq \mathsf{Enc}(pk, m; r')$, it follows that $\mathsf{G}^{1}_{\Pi,\Sigma_1}$ is perfectly indistinguishable from $\mathsf{G}^{2}_{\Pi,\Sigma_1}$. Moreover, $\mathsf{G}^{2}_{\Pi,\Sigma_1}$ and $\mathsf{G}^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma_1}$ are identical until bad is set to true, which happens with probability $q\ell/\rho$. Therefore, using the fundamental lemma of game playing, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_1}(\mathrm{D}) &= \Pr[\mathsf{G}^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma_1}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma_1}(\mathrm{D})] \\
&= (\Pr[\mathsf{G}^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma_1}(\mathrm{D})] - \Pr[\mathsf{G}^{1}_{\Pi,\Sigma_1}(\mathrm{D})]) \\
&\quad + (\Pr[\mathsf{G}^{1}_{\Pi,\Sigma_1}(\mathrm{D})] - \Pr[\mathsf{G}^{2}_{\Pi,\Sigma_1}(\mathrm{D})]) \\
&\quad + (\Pr[\mathsf{G}^{2}_{\Pi,\Sigma_1}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma_1}(\mathrm{D})]) \\
&\leq (\Pr[\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}(\mathrm{D}')] - \Pr[\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}(\mathrm{D}')]) + 0 + \Pr[\mathsf{bad}] \\
&\leq \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathrm{D}') + \frac{q\ell}{\rho}.
\end{aligned}$$

□

A.2 Proofs for the Σ_2 Construction

Lemma 4.3. *Let Σ_2 be the anamorphic extension from Construction 2 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* sec *distinguisher D into a* prf *distinguisher D' with $q(\mathrm{D}') = q(\mathrm{D}) \leq \tau$ such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_2}(\mathrm{D}) = \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathrm{D}').$$

Proof. Define games $\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_2}$, $\mathsf{G}^{1}_{\Pi,\Sigma_2}$, $\mathsf{G}^{2}_{\Pi,\Sigma_2}$, $\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_2}$, and distinguisher D' as in Figure 17. D' is such that if it is interacting with $\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}$, it perfectly emulates $\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_2}$ towards D, and if it is interacting with $\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}$, it perfectly emulates $\mathsf{G}^{1}_{\Pi,\Sigma_2}$ towards D. Since the counter $\mathtt{ctr}$ used as input to the uniform random function f is never repeating, t is effectively uniformly distributed in $\mathsf{G}^{1}_{\Pi,\Sigma_2}$, and since so is $r = \hat{m} \oplus t$, $\mathsf{G}^{1}_{\Pi,\Sigma_2}$ is perfectly indistinguishable from $\mathsf{G}^{2}_{\Pi,\Sigma_2}$. Moreover, $\mathsf{G}^{2}_{\Pi,\Sigma_2}$ is just a more explicit description of $\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_2}$, and thus they too are perfectly indistinguishable. Therefore, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_2}(\mathrm{D}) &= \Pr[\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_2}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_2}(\mathrm{D})] \\
&= (\Pr[\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_2}(\mathrm{D})] - \Pr[\mathsf{G}^{1}_{\Pi,\Sigma_2}(\mathrm{D})]) \\
&\quad + (\Pr[\mathsf{G}^{1}_{\Pi,\Sigma_2}(\mathrm{D})] - \Pr[\mathsf{G}^{2}_{\Pi,\Sigma_2}(\mathrm{D})]) \\
&\quad + (\Pr[\mathsf{G}^{2}_{\Pi,\Sigma_2}(\mathrm{D})] - \Pr[\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_2}(\mathrm{D})]) \\
&= (\Pr[\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}0}(\mathrm{D}')] - \Pr[\mathsf{G}_{\mathsf{F}}^{\mathsf{prf}\text{-}1}(\mathrm{D}')]) + 0 + 0 \\
&= \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathrm{D}').
\end{aligned}$$

□

---

[8] Whether D repeats counters or not, the probability of a collision between r and r', over all of D's queries, remains the same.

```
$\boxed{G^{\text{sec-0}}_{\Pi,\Sigma_2}}$ $\dbox{G^{1}_{\Pi,\Sigma_2}}$
INIT():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\boxed{K \xleftarrow{\$} \mathcal{K}}$
    $\dbox{f \xleftarrow{\$} \mathcal{R}^{[\tau]}}$
    $\mathsf{ctr} := 0$
    return $(sk, pk)$
AENC$(m, \hat{m})$:
    $\boxed{t := F(K, \mathsf{ctr})}$
    $\dbox{t := f(\mathsf{ctr})}$
    $r := \hat{m} \oplus t$
    $c := \mathsf{Enc}(pk, m; r)$
    $\mathsf{ctr} := \mathsf{ctr} + 1$
    return $c$
```

```
$G^{2}_{\Pi,\Sigma_2}$
INIT():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
AENC$(m, \hat{m})$:
    $r \xleftarrow{\$} \mathcal{R}$
    $c \leftarrow \mathsf{Enc}(pk, m; r)$
    return $c$

$G^{\text{sec-1}}_{\Pi,\Sigma_2}$
INIT():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
AENC$(m, \hat{m})$:
    $c \leftarrow \mathsf{Enc}(pk, m)$
    return $c$
```

```
$D'^{\text{INIT},\text{EVAL}}$
    $b \leftarrow D^{\text{INIT}^*,\text{AENC}^*}$
    return $b$
INIT*():
    INIT()
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\mathsf{ctr} := 0$
    return $(sk, pk)$
AENC*$(m, \hat{m})$:
    $t := \text{EVAL}(\mathsf{ctr})$
    $r := \hat{m} \oplus t$
    $c := \mathsf{Enc}(pk, m; r)$
    $\mathsf{ctr} := \mathsf{ctr} + 1$
    return $c$
```

Figure 17: Games $G^{\text{sec-0}}_{\Pi,\Sigma_2}$, G^{1}_{Π,Σ_2}, G^{2}_{Π,Σ_2}, $G^{\text{sec-1}}_{\Pi,\Sigma_2}$, and distinguisher D' for the proof of Lemma 4.3.

Lemma 4.4. *Let Σ_2 be the anamorphic extension from Construction 2 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* rob *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \tau$ such that*

$$\mathsf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_2}(D) \leq \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(D') + \frac{q\ell}{\rho}.$$

Proof. Define games $G^{\text{rob-0}}_{\Pi,\Sigma_2}$, G^{1}_{Π,Σ_2}, G^{2}_{Π,Σ_2}, $G^{\text{rob-1}}_{\Pi,\Sigma_2}$, and distinguisher D' as in Figure 18. D' is such that if it is interacting with $G^{\text{prf-0}}_{\mathsf{F}}$, it perfectly emulates $G^{\text{rob-0}}_{\Pi,\Sigma_2}$ towards D, and if it is interacting with $G^{\text{prf-1}}_{\mathsf{F}}$, it perfectly emulates G^{1}_{Π,Σ_2} towards D. Without loss of generality,[9] we can assume that D never repeats counters, hence t is effectively uniformly distributed in G^{1}_{Π,Σ_2}. Then, since $(c_1, c_2) := \mathsf{Enc}(pk, m)$ corresponds to $r \xleftarrow{\$} \mathcal{R}$ followed by $(c_1, c_2) := \mathsf{Enc}(pk, m; r)$, and since $\gamma(\beta(r), t) = \beta(\hat{m})$ if and only if $r = \hat{m} \oplus t$, it follows that G^{1}_{Π,Σ_2} is perfectly indistinguishable from G^{2}_{Π,Σ_2}. Moreover, G^{2}_{Π,Σ_2} and $G^{\text{rob-1}}_{\Pi,\Sigma_2}$ are identical until bad is set to true, which happens with

---

[9] Whether D repeats counters or not, the probability of a collision between r and $\hat{m} \oplus t$, over all of D's queries, remains the same.

```
$\boxed{G^{\text{rob-0}}_{\Pi,\Sigma_2}}$ $\dbox{G^{1}_{\Pi,\Sigma_2}}$
Init():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\boxed{K \xleftarrow{\$} \mathcal{K}}$
    $\dbox{f \xleftarrow{\$} \mathcal{R}^{[\tau]}}$
EncADec(m, ctr):
    $(c_1, c_2) := \mathsf{Enc}(pk, m)$
    $\boxed{t := F(K, ctr)}$
    $\dbox{t := f(ctr)}$
    $s := \gamma(c_2, t)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}) = s$ do
            return $\hat{m}$
    return $\perp$
```

```
$G^{2}_{\Pi,\Sigma_2}$
Init():
    bad := false
EncADec(m, ctr):
    $r \xleftarrow{\$} \mathcal{R}$
    $t \xleftarrow{\$} \mathcal{R}$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $r = \hat{m} \oplus t$ then
            bad := true
            return $\hat{m}$
    return $\perp$
```

```
$G^{\text{rob-1}}_{\Pi,\Sigma_2}$
Init():
    // Do nothing
EncADec(m, ctr):
    return $\perp$
```

```
$D'^{\textsc{Init},\textsc{Eval}}$
    $b \leftarrow D^{\textsc{Init}^*,\textsc{EncADec}^*}$
    return b
Init*():
    Init()
    $(sk, pk) \leftarrow \mathsf{Gen}()$
EncADec*(m, ctr):
    $(c_1, c_2) := \mathsf{Enc}(pk, m)$
    $t := \textsc{Eval}(ctr)$
    $s := \gamma(c_2, t)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}) = s$ do
            return $\hat{m}$
    return $\perp$
```

Figure 18: Games $G^{\text{rob-0}}_{\Pi,\Sigma_2}$, G^{1}_{Π,Σ_2}, G^{2}_{Π,Σ_2}, $G^{\text{rob-1}}_{\Pi,\Sigma_2}$, and distinguisher D' for the proof of Lemma 4.4.

probability $q\ell/\rho$. Therefore, using the fundamental lemma of game playing, we have

$$\begin{aligned}
\mathbf{Adv}^{\text{rob}}_{\Pi,\Sigma_2}(D) &= \Pr[G^{\text{rob-0}}_{\Pi,\Sigma_2}(D)] - \Pr[G^{\text{rob-1}}_{\Pi,\Sigma_2}(D)] \\
&= (\Pr[G^{\text{rob-0}}_{\Pi,\Sigma_2}(D)] - \Pr[G^{1}_{\Pi,\Sigma_2}(D)]) \\
&\quad + (\Pr[G^{1}_{\Pi,\Sigma_2}(D)] - \Pr[G^{2}_{\Pi,\Sigma_2}(D)]) \\
&\quad + (\Pr[G^{2}_{\Pi,\Sigma_2}(D)] - \Pr[G^{\text{rob-1}}_{\Pi,\Sigma_2}(D)]) \\
&\leq (\Pr[G^{\text{prf-0}}_{F}(D')] - \Pr[G^{\text{prf-1}}_{F}(D')]) + 0 + \Pr[\text{bad}] \\
&\leq \mathbf{Adv}^{\text{prf}}_{F}(D') + \frac{q\ell}{\rho}.
\end{aligned}$$

□

A.3 Proofs for the Σ_3 Construction

Lemma 4.6. *Let Σ_3 be the anamorphic extension from Construction 3 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* cor *distinguisher D into a* prf *distinguisher D' with $q \doteq q(D') = q(D) \leq \sigma\tau$ such that*

$$\mathbf{Adv}^{\text{cor}}_{\Pi,\Sigma_3}(D) \leq \mathbf{Adv}^{\text{prf}}_{F}(D') + \frac{q\sigma\ell}{\rho}.$$

```
$\boxed{G^{\text{cor-0}}_{\Pi,\Sigma_3,m}}$ $\dbox{G^{1}_{\Pi,\Sigma_3,m}}$
$\textsc{Init}()$:
$(sk, pk) \leftarrow \mathsf{Gen}()$
$\boxed{K \xleftarrow{\$} \mathcal{K}}$
$\dbox{f \xleftarrow{\$} \mathcal{R}^{[\sigma]\times[\tau]}}$
$(x, y) := (0, 0)$
$\textsc{AEncADec}(\hat{m})$:
repeat
    $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
    $\boxed{t := \mathsf{F}(K, (x, y))}$
    $\dbox{t := f((x, y))}$
    $r := \hat{m} \oplus t$
until $\delta(\beta(r)) = y$
$(c_1, c_2) := \mathsf{Enc}(pk, m; r)$
$y' := \delta(c_2)$    $/\!\!/ = \delta(\beta(r)) = y$
foreach $x' \in [\sigma]$ do
    $\boxed{t' := \mathsf{F}(K, (x', y'))}$
    $\dbox{t' := f((x', y'))}$
    $s := \gamma(c_2, t')$    $/\!\!/ = \gamma(\beta(r), t')$
    foreach $\hat{m}' \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}') = s$ then
            return $\hat{m}'$
return $\perp$
```

```
$G^{2}_{\Pi,\Sigma_3,m}$
$\textsc{Init}()$:
$(x, y) := (0, 0)$
bad := true
$\textsc{AEncADec}(\hat{m})$:
repeat
    $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
    $t_x \xleftarrow{\$} \mathcal{R}$
    $r := \hat{m} \oplus t_x$
until $\delta(\beta(r)) = y$
foreach $x' \in [\sigma]$ do
    if $x' \neq x$
        $t_{x'} \xleftarrow{\$} \mathcal{R}$
    $s := \gamma(\beta(r), t_{x'})$
    foreach $\hat{m}' \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}') = s$ then
            if $\hat{m}' \neq \hat{m}$ then
                bad := true
            return $\hat{m}'$
$/\!\!/$ Unreachable
```

```
$G^{\text{cor-1}}_{\Pi,\Sigma_3,m}$
$\textsc{Init}()$:
$/\!\!/$ Do nothing
$\textsc{AEncADec}(\hat{m})$:
return $\hat{m}$
```

```
$D'^{\textsc{Init},\textsc{Eval}}$
$b \leftarrow D^{\textsc{Init}^*,\textsc{AEncADec}^*}$
return $b$
$\textsc{Init}^*()$:
$\textsc{Init}()$
$(sk, pk) \leftarrow \mathsf{Gen}()$
$(x, y) := (0, 0)$
$\textsc{AEncADec}^*(m)$:
repeat
    $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
    $t := \textsc{Eval}((x, y))$
    $r := \hat{m} \oplus t$
until $\delta(\beta(r)) = y$
foreach $x' \in [\sigma]$ do
    $t' := \textsc{Eval}((x', y))$
    $s := \gamma(\beta(r), t')$
    foreach $\hat{m}' \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}') = s$ then
            return $\hat{m}'$
$/\!\!/$ Unreachable
```

Figure 19: Games $G^{\text{cor-0}}_{\Pi,\Sigma_3,m}$, $G^{1}_{\Pi,\Sigma_3,m}$, $G^{2}_{\Pi,\Sigma_3,m}$, $G^{\text{cor-1}}_{\Pi,\Sigma_3,m}$, and distinguisher D' for the proof of Lemma 4.6.

Proof. Define games $G^{\text{cor-0}}_{\Pi,\Sigma_3,m}$, $G^{1}_{\Pi,\Sigma_3,m}$, $G^{2}_{\Pi,\Sigma_3,m}$, $G^{\text{cor-1}}_{\Pi,\Sigma_3,m}$ and distinguisher D' as in Figure 19. Note that for convenience we define game $G^{\text{cor-0}}_{\Pi,\Sigma_3,m}$ without pre-processing. D' is such that if it is interacting with $G^{\text{prf-0}}_{F}$, it perfectly emulates $G^{\text{cor-0}}_{\Pi,\Sigma_3}$ towards D, and if it is interacting with $G^{\text{prf-1}}_{F}$, it perfectly emulates G^{1}_{Π,Σ_3} towards D. Since the state (x, y) used as input to the uniform random function f is never repeating, t and every value t' computed in the for loop for each $x' \in [\sigma]$ are effectively uniformly distributed in G^{1}_{Π,Σ_3}. Then it follows that G^{1}_{Π,Σ_3} is perfectly indistinguishable from G^{2}_{Π,Σ_3}. Moreover, G^{2}_{Π,Σ_3} and $G^{\text{cor-1}}_{\Pi,\Sigma_3}$ are identical until bad is set to true. For each queried covert message $\hat{m}$, this happens if there exists an $x' \neq x$ and a $\hat{m}' \neq \hat{m}$ such that $\beta(\hat{m}') = s$. To compute the probability of bad being set to true for a fixed $\hat{m}$, let $\mathcal{X} \doteq [\sigma] \setminus \{x\}$ and $\mathcal{N} \doteq \hat{\mathcal{M}} \setminus \{\hat{m}\}$. Then, since $\beta(\hat{m}') = \gamma(\beta(\hat{m} \oplus t_x), t_{x'})$ if and only if $\hat{m} \oplus t_x = \hat{m}' \oplus t_{x'}$, we have

$$\begin{aligned}
&\Pr[\exists x' \in \mathcal{X}, \hat{m}' \in \mathcal{N} : \beta(\hat{m}') = \gamma(\beta(r), t_{x'})] \\
&\quad = \Pr[\exists x' \in \mathcal{X}, \hat{m}' \in \mathcal{N} : \beta(\hat{m}') = \gamma(\beta(\hat{m} \oplus t_x), t_{x'})] \\
&\quad = \Pr[\exists x' \in \mathcal{X}, \hat{m}' \in \mathcal{N} : \hat{m} \oplus t_x = \hat{m}' \oplus t_{x'}] \\
&\quad = \Pr[\exists x' \in \mathcal{X}, \hat{m}' \in \mathcal{N} : t_x \oplus t_{x'} = \hat{m} \oplus \hat{m}'] \\
&\quad \leq \sum_{x' \in \mathcal{X}} \sum_{\hat{m}' \in \mathcal{N}} \Pr[t_x \oplus t_{x'} = \hat{m} \oplus \hat{m}'] \\
&\quad \leq \frac{\sigma \ell}{\rho}.
\end{aligned}$$

Therefore, using the fundamental lemma of game playing, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathsf{cor}}_{\Pi,\Sigma_3}(D) &= \Pr[G^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma_3}(D)] - \Pr[G^{\mathsf{cor}\text{-}1}_{\Pi,\Sigma_3}(D)] \\
&= (\Pr[G^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma_3}(D)] - \Pr[G^{1}_{\Pi,\Sigma_3}(D)]) \\
&\qquad + (\Pr[G^{1}_{\Pi,\Sigma_3}(D)] - \Pr[G^{2}_{\Pi,\Sigma_3}(D)]) \\
&\qquad + (\Pr[G^{2}_{\Pi,\Sigma_3}(D)] - \Pr[G^{\mathsf{cor}\text{-}1}_{\Pi,\Sigma_3}(D)]) \\
&\leq (\Pr[G^{\mathsf{prf}\text{-}0}_{F}(D')] - \Pr[G^{\mathsf{prf}\text{-}1}_{F}(D')]) + 0 + \Pr[\mathsf{bad}] \\
&\leq \mathbf{Adv}^{\mathsf{prf}}_{F}(D') + \frac{q\sigma\ell}{\rho}.
\end{aligned}$$

□

Lemma 4.7. *Let* Σ_3 *be the anamorphic extension from* Construction 3 *for an SRR PKE scheme* Π *satisfying* Definition 4.1. *There exists an efficient transformation of any* sec *distinguisher* D *into a* prf *distinguisher* D' *with* $q(D') = q(D) \leq \sigma\tau$ *such that*

$$\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_3}(D) = \mathbf{Adv}^{\mathsf{prf}}_{F}(D').$$

Proof. Define games $G^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}$, G^{1}_{Π,Σ_3}, G^{2}_{Π,Σ_3}, $G^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}$, and distinguisher D' as in Figure 20. D' is such that if it is interacting with $G^{\mathsf{prf}\text{-}0}_{F}$, it perfectly emulates $G^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}$ towards D, and if it is interacting with $G^{\mathsf{prf}\text{-}1}_{F}$, it perfectly emulates G^{1}_{Π,Σ_3} towards D. Since the state pair (x, y) used as input to the uniform random function f is never repeating, t is effectively uniformly distributed in G^{1}_{Π,Σ_3}, and since so is $r = \hat{m} \oplus t$, G^{1}_{Π,Σ_3} is perfectly indistinguishable from G^{2}_{Π,Σ_3}. Moreover, since by Lemma 4.5 we know that $\mathbb{E}[T] \approx \tau < \infty$, where T denotes the number of iterations in AEnc, G^{2}_{Π,Σ_3} is perfectly indistinguishable from $G^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}$. Therefore, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathsf{sec}}_{\Pi,\Sigma_3}(D) &= \Pr[G^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}(D)] - \Pr[G^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}(D)] \\
&= (\Pr[G^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}(D)] - \Pr[G^{1}_{\Pi,\Sigma_3}(D)]) \\
&\qquad + (\Pr[G^{1}_{\Pi,\Sigma_3}(D)] - \Pr[G^{2}_{\Pi,\Sigma_3}(D)]) \\
&\qquad + (\Pr[G^{2}_{\Pi,\Sigma_3}(D)] - \Pr[G^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}(D)]) \\
&= (\Pr[G^{\mathsf{prf}\text{-}0}_{F}(D')] - \Pr[G^{\mathsf{prf}\text{-}1}_{F}(D')]) + 0 + 0 \\
&= \mathbf{Adv}^{\mathsf{prf}}_{F}(D').
\end{aligned}$$

□

```
$\boxed{\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}}$ $\dbox{\mathsf{G}^{1}_{\Pi,\Sigma_3}}$
$\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $\boxed{K \xleftarrow{\$} \mathcal{K}}$
    $\dbox{f \xleftarrow{\$} \mathcal{R}^{[\sigma]\times[\tau]}}$
    $(x, y) := (0, 0)$
    return $(sk, pk)$
$\textsc{AEnc}(m, \hat{m})$:
    repeat
        $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
        $\boxed{t := \mathsf{F}(K, (x, y))}$
        $\dbox{t := f((x, y))}$
        $r := \hat{m} \oplus t$
    until $\delta(\beta(r)) = y$
    $c := \mathsf{Enc}(pk, m; r)$
    return $c$
```

```
$\mathsf{G}^{2}_{\Pi,\Sigma_3}$
$\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $(x, y) := (0, 0)$
    return $(sk, pk)$
$\textsc{AEnc}(m, \hat{m})$:
    repeat
        $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
        $r \xleftarrow{\$} \mathcal{R}$
    until $\delta(\beta(r)) = y$
    $c := \mathsf{Enc}(pk, m; r)$
    return $c$
```

```
$\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}$
$\textsc{Init}()$:
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
$\textsc{AEnc}(m, \hat{m})$:
    $c := \mathsf{Enc}(pk, m)$
    return $c$
```

```
$\mathsf{D}'^{\textsc{Init},\textsc{Eval}}$
    $b \leftarrow \mathsf{D}^{\textsc{Init}^*,\textsc{AEnc}^*}$
    return $b$
$\textsc{Init}^*()$:
    $\textsc{Init}()$
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    $(x, y) := (0, 0)$
    return $(sk, pk)$
$\textsc{AEnc}^*(m, \hat{m})$:
    repeat
        $(x, y) := \mathsf{IL}_{\sigma,\tau}(x, y)$
        $\textsc{Eval}((x, y))$
        $r := \hat{m} \oplus t$
    until $\delta(\beta(r)) = y$
    $c := \mathsf{Enc}(pk, m; r)$
    return $c$
```

Figure 20: Games $\mathsf{G}^{\mathsf{sec}\text{-}0}_{\Pi,\Sigma_3}$, $\mathsf{G}^{1}_{\Pi,\Sigma_3}$, $\mathsf{G}^{2}_{\Pi,\Sigma_3}$, $\mathsf{G}^{\mathsf{sec}\text{-}1}_{\Pi,\Sigma_3}$, and distinguisher D' for the proof of Lemma 4.7.

Lemma 4.8. *Let Σ_3 be the anamorphic extension from Construction 3 for an SRR PKE scheme Π satisfying Definition 4.1. There exists an efficient transformation of any* rob *distinguisher D into a* prf *distinguisher D' with $q \doteq q(\mathsf{D}') = q(\mathsf{D}) \leq \sigma\tau$ such that*

$$\mathbf{Adv}^{\mathsf{rob}}_{\Pi,\Sigma_3}(\mathsf{D}) \leq \mathbf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathsf{D}') + \frac{q\sigma\ell}{\rho}.$$

Proof. Define games $\mathsf{G}^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma_3}$, $\mathsf{G}^{1}_{\Pi,\Sigma_3}$, $\mathsf{G}^{2}_{\Pi,\Sigma_3}$, $\mathsf{G}^{\mathsf{rob}\text{-}1}_{\Pi,\Sigma_3}$, and distinguisher D' as in Figure 21. Note that for convenience we define game $\mathsf{G}^{\mathsf{cor}\text{-}0}_{\Pi,\Sigma_3,m}$ without pre-processing. D' is such that if it is interacting with $\mathsf{G}^{\mathsf{prf}\text{-}0}_{\mathsf{F}}$, it perfectly emulates $\mathsf{G}^{\mathsf{rob}\text{-}0}_{\Pi,\Sigma_3}$ towards D, and if it is interacting with $\mathsf{G}^{\mathsf{prf}\text{-}1}_{\mathsf{F}}$, it perfectly emulates $\mathsf{G}^{1}_{\Pi,\Sigma_3}$ towards D. Without loss of generality,[10] we can assume that state pairs (x, y) are never repeated, hence t is effectively uniformly distributed in $\mathsf{G}^{1}_{\Pi,\Sigma_3}$. Then, since $(c_1, c_2) := \mathsf{Enc}(pk, m)$ corresponds to $r \xleftarrow{\$} \mathcal{R}$ followed by $(c_1, c_2) := \mathsf{Enc}(pk, m; r)$, and since $\gamma(\beta(r), t) = \beta(\hat{m})$ if and only if $r = \hat{m} \oplus t$, it follows that $\mathsf{G}^{1}_{\Pi,\Sigma_3}$ is perfectly indistinguishable from $\mathsf{G}^{2}_{\Pi,\Sigma_3}$. Moreover, $\mathsf{G}^{2}_{\Pi,\Sigma_3}$ and

---

[10] Whether a state pair (x, y) is repeated or not, the probability of a collision between r and $\hat{m} \oplus t$, over all of D's queries, remains the same.

```
$\boxed{G^{\text{rob-0}}_{\Pi,\Sigma_3}}$ $\dbox{G^{1}_{\Pi,\Sigma_3}}$
INIT():
$(sk, pk) \leftarrow \mathsf{Gen}()$
$\boxed{K \xleftarrow{\$} \mathcal{K}}$
$\dbox{f \xleftarrow{\$} \mathcal{R}^{[\tau]}}$
ENCADEC($m$):
$(c_1, c_2) := \mathsf{Enc}(pk, m)$
$y := \delta(c_2)$
foreach $x \in [\sigma]$ do
    $\boxed{t := F(K, (x, y))}$
    $\dbox{t := f((x, y))}$
    $s := \gamma(c_2, t)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}) = s$ do
            return $\hat{m}$
return $\perp$
```

```
$G^{2}_{\Pi,\Sigma_3}$
INIT():
bad := false
ENCADEC($m$):
$r \xleftarrow{\$} \mathcal{R}$
foreach $x \in [\sigma]$ do
    $t \xleftarrow{\$} \mathcal{R}$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $r = \hat{m} \oplus t$ then
            bad := true
            return $\hat{m}$
return $\perp$

$G^{\text{rob-1}}_{\Pi,\Sigma_3}$
INIT():
// Do nothing
ENCADEC($m$):
return $\perp$
```

```
$D'^{\text{INIT},\text{EVAL}}$
$b \leftarrow D^{\text{INIT}^*,\text{ENCADEC}^*}$
return $b$
INIT*():
INIT()
$(sk, pk) \leftarrow \mathsf{Gen}()$
ENCADEC*($m$):
$(c_1, c_2) := \mathsf{Enc}(pk, m)$
$y := \delta(c_2)$
foreach $x \in [\sigma]$ do
    $t := \text{EVAL}((x, y))$
    $s := \gamma(c_2, t)$
    foreach $\hat{m} \in \hat{\mathcal{M}}$ do
        if $\beta(\hat{m}) = s$ do
            return $\hat{m}$
return $\perp$
```

Figure 21: Games $G^{\text{rob-0}}_{\Pi,\Sigma_3}$, G^{1}_{Π,Σ_3}, G^{2}_{Π,Σ_3}, $G^{\text{rob-1}}_{\Pi,\Sigma_3}$, and distinguisher D' for the proof of Lemma 4.8.

$G^{\text{rob-1}}_{\Pi,\Sigma_3}$ are identical until bad is set to true, which happens with probability $q\sigma\ell/\rho$. Therefore, using the fundamental lemma of game playing, we have

$$\begin{aligned}
\mathbf{Adv}^{\text{rob}}_{\Pi,\Sigma_2}(D) &= \Pr[G^{\text{rob-0}}_{\Pi,\Sigma_2}(D)] - \Pr[G^{\text{rob-1}}_{\Pi,\Sigma_2}(D)] \\
&= (\Pr[G^{\text{rob-0}}_{\Pi,\Sigma_2}(D)] - \Pr[G^{1}_{\Pi,\Sigma_2}(D)]) \\
&\quad + (\Pr[G^{1}_{\Pi,\Sigma_2}(D)] - \Pr[G^{2}_{\Pi,\Sigma_2}(D)]) \\
&\quad + (\Pr[G^{2}_{\Pi,\Sigma_2}(D)] - \Pr[G^{\text{rob-1}}_{\Pi,\Sigma_2}(D)]) \\
&\le (\Pr[G^{\text{prf-0}}_{F}(D')] - \Pr[G^{\text{prf-1}}_{F}(D')]) + 0 + \Pr[\text{bad}] \\
&\le \mathbf{Adv}^{\text{prf}}_{F}(D') + \frac{q\sigma\ell}{\rho}.
\end{aligned}$$

□

B IND-CPA Security of Anamorphic Ciphertexts

For a PKE scheme Π with anamorphic extension Σ, [PPY22] additionally defines security in terms of *indistinguishability of anamorphic ciphertexts under a chosen-plaintext attack* (ind-anam-cpa). More specifically, they require that for a fixed (normal) message m, anamorphic encryptions of covert messages $\hat{m}_0$ and $\hat{m}_1$ with m be indistinguishable. We reformulate this notion as *real-or-random* rather than *left-or-right* (cf. [BDJR97]).

```
Game $G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}$
    INIT():
        $(sk, pk) \leftarrow \mathsf{Gen}()$
        $dk \leftarrow \mathsf{aGen}(sk, pk)$
        return $(sk, pk)$
    AENC($\hat{m}$):
        $c \leftarrow \mathsf{aEnc}(dk, m, \hat{m})$
        return $c$
```

```
Game $G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}$
    INIT():
        $(sk, pk) \leftarrow \mathsf{Gen}()$
        $dk \leftarrow \mathsf{aGen}(sk, pk)$
        return $(sk, pk)$
    AENC($\hat{m}$):
        $\tilde{m} \xleftarrow{\$} \hat{\mathcal{M}}$
        $c \leftarrow \mathsf{aEnc}(dk, m, \tilde{m})$
        return $c$
```

Figure 22: Games defining ind-anam-cpa security of an anamorphic extension $\Sigma = (\mathsf{aGen}, \mathsf{aEnc}, \mathsf{aDec})$ for PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

Definition B.1. *For a PKE scheme Π with anamorphic extension Σ and arbitrary message $m \in \mathcal{M}$, we define the advantage of an* ind-anam-cpa *distinguisher D as*

$$\mathbf{Adv}^{\text{ind-anam-cpa}}_{\Pi,\Sigma,m}(D) \doteq \Pr[G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}(D)] - \Pr[G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}(D)],$$

with games $G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}$ and $G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}$ as defined in Figure 22. We let $q(D)$ denote the total number of messages queried to ENC *by D.*

As shown in [PPY22], the sec notion for anamorphic extensions implies ind-anam-cpa security, which roughly speaking means that in order to show that anamorphic ciphertexts are indistinguishable from one another, it suffices to show that anamorphic ciphertexts are indistinguishable from regular ones. We next reprove this simple result using our new formalism.

Theorem B.1. *For a PKE scheme Π with anamorphic extension Σ, let and $m \in \mathcal{M}$ be arbitrary. There exists an efficient transformation of any* ind-anam-cpa *distinguisher D into an* sec *distinguisher D'_m with $q(D'_m) = q(D)$ such that*

$$\mathbf{Adv}^{\text{ind-anam-cpa}}_{\Pi,\Sigma,m}(D) \leq 2 \cdot \mathbf{Adv}^{\text{sec}}_{\Pi,\Sigma}(D'_m).$$

Proof. For an arbitrary message $m \in \mathcal{M}$, define game $G_{\Pi,\Sigma,m}$ and adversaries $D'_{m,1}$, $D'_{m,2}$ as in Figure 23. $D'_{m,1}$ is such that if it is interacting with $G^{\text{sec-0}}_{\Pi,\Sigma}$, it perfectly emulates $G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}$ towards D, and if it is interacting with $G^{\text{sec-1}}_{\Pi,\Sigma}$, it perfectly emulates $G^{1}_{\Pi,\Sigma}$ towards D. $D'_{m,2}$ is such that if it is interacting with $G^{\text{sec-1}}_{\Pi,\Sigma}$, it perfectly emulates $G^{1}_{\Pi,\Sigma}$ towards D, and if it is interacting

```
$G_{\Pi,\Sigma,m}$
Init():
    $(sk, pk) \leftarrow \mathsf{Gen}()$
    return $(sk, pk)$
AEnc($\hat{m}$):
    $c \leftarrow \mathsf{Enc}(pk, m)$
    return $c$
```

```
$\boxed{D'^{\textsc{Init},\textsc{AEnc}}_{m,1}}$  $\dbox{D'^{\textsc{Init},\textsc{AEnc}}_{m,2}}$
    $b \leftarrow D^{\textsc{Init}^*,\textsc{AEnc}^*}$
    $\boxed{\textbf{return } b}$
    $\dbox{\textbf{return } 1 - b}$
Init*():
    $(sk, pk) \leftarrow \textsc{Init}()$
    return $(sk, pk)$
AEnc*($\hat{m}$):
    $\boxed{c \leftarrow \textsc{AEnc}(m, \hat{m})}$
    $\dbox{\tilde{m} \xleftarrow{\$} \mathcal{M}}$
    $\dbox{c \leftarrow \textsc{AEnc}(m, \tilde{m})}$
    return $c$
```

Figure 23: Game $G_{\Pi,\Sigma,m}$ and adversaries $D'_{m,1}$, $D'_{m,2}$ for the proof of Theorem B.1.

with $G^{\text{sec-0}}_{\Pi,\Sigma}$, it perfectly emulates $G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}$ towards D. Therefore, we have

$$\begin{aligned}
\mathbf{Adv}^{\text{ind-anam-cpa}}_{\Pi,\Sigma,m}(D) &= \Pr[G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}(D)] - \Pr[G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}(D)] \\
&= (\Pr[G^{\text{ind-anam-cpa-0}}_{\Pi,\Sigma,m}(D)] - \Pr[G_{\Pi,\Sigma,m}(D)]) \\
&\quad + (\Pr[G_{\Pi,\Sigma,m}(D)] - \Pr[G^{\text{ind-anam-cpa-1}}_{\Pi,\Sigma,m}(D)]) \\
&= (\Pr[G^{\text{sec-0}}_{\Pi,\Sigma}(D'_{m,1})] - \Pr[G^{\text{sec-1}}_{\Pi,\Sigma}(D'_{m,1})]) \\
&\quad + (\Pr[G^{\text{sec-0}}_{\Pi,\Sigma}(D'_{m,2})] - \Pr[G^{\text{sec-1}}_{\Pi,\Sigma}(D'_{m,2})]) \\
&= \mathbf{Adv}^{\text{sec}}_{\Pi,\Sigma}(D'_{m,1}) + \mathbf{Adv}^{\text{sec}}_{\Pi,\Sigma}(D'_{m,2}) \\
&= 2 \cdot \mathbf{Adv}^{\text{sec}}_{\Pi,\Sigma}(D'),
\end{aligned}$$

where D'_m is the distinguisher that initially flips a uniform coin, and depending on the outcome it either behaves as $D'_{m,1}$ or as $D'_{m,2}$. ☐

C ElGamal's Σ_3 Anamorphic Extension Test Code

We implemented the synchronized robustly anamorphic extension AnamElGamal $\doteq$ (aGen, aEnc, aDec) with pre-computation for the SRR PKE scheme ElGamal from Construction 9 in Python, with F = `AES`, to test the four scenarios mentioned in Section 1.2. Note that this code requires the package `PyCryptodome`.

```
import random
from Crypto.Cipher import AES


class PublicParams:
```

```
    def __init__(self, p, q, g):
        self.p = p
        self.q = q
        self.g = g

class AnamParams:
    def __init__(self, l, s, t):
        self.F = lambda pp, K, x, y: \
                    int.from_bytes(AES.new(K, AES.MODE_ECB) \
                    .encrypt(x.to_bytes(8, 'little') \
                    + y.to_bytes(8, 'little')), "little") % pp.p
        self.d = lambda ap, x: x % ap.t
        self.l = l
        self.s = s
        self.t = t

class KeyPair:
    def __init__(self, sk, pk):
        self.sk = sk
        self.pk = pk

class DoubleKey:
    def __init__(self, K, T, pk):
        self.K = K
        self.T = T
        self.pk = pk

def Gen(pp):
    sk = random.randint(0, pp.q - 1)
    pk = pow(pp.g, sk, pp.p)
    return KeyPair(sk, pk)

def Enc(pp, pk, msg):
    r = random.randint(0, pp.q - 1)
    c0 = (msg * pow(pk, r, pp.p)) % pp.p
    c1 = pow(pp.g, r, pp.p)
    return c0, c1

def Dec(pp, sk, c):
    return (c[0] * pow(c[1], -sk, pp.p)) % pp.p

def aGen(pp, ap, pk):
    K = random.randbytes(16)
```

```
    T = dict()
    for i in range(ap.l):
        T[pow(pp.g, i, pp.p)] = i
    return DoubleKey(K, T, pk)

def aEncCtr(pp, ap, dk, msg, cm, ctr):
    found = False
    for x in range(ctr[0], ap.s):
        for y in range(ctr[1], ap.t):
            t = ap.F(pp, dk.K, x, y)
            r = (cm + t) % pp.q
            if ap.d(ap, pow(pp.g, r, pp.p)) == y:
                found = True
                break
        if found:
            break
        ctr[1] = 0
    ctr[0] = (x + (1 if y == ap.t - 1 else 0)) % ap.s
    ctr[1] = (y + 1) % ap.t
    c0 = (msg * pow(dk.pk, r, pp.p)) % pp.p
    c1 = pow(pp.g, r, pp.p)
    ctx = (c0, c1)
    return ctx, ctr

def aEnc(pp, ap, dk, msg, cm):
    while True:
        x = random.randint(0, ap.s - 1)
        y = random.randint(0, ap.t - 1)
        t = ap.F(pp, dk.K, x, y)
        r = (cm + t) % pp.q
        if ap.d(ap, pow(pp.g, r, pp.p)) == y:
            break
    c0 = (msg * pow(dk.pk, r, pp.p)) % pp.p
    c1 = pow(pp.g, r, pp.p)
    ctx = (c0, c1)
    return ctx

def aDec(pp, ap, dk, ctx):
    y = ap.d(ap, ctx[1])
    for x in range(ap.s):
        t = ap.F(pp, dk.K, x, y)
        s = (ctx[1] * pow(pp.g, -t, pp.p)) % pp.p
        if s in dk.T:
```

```
            return dk.T[s]
    return -1

# Settings
runs = 50

# Public Parameters (safe prime, pow(g, (p - 1) // 2, p) != 1)
#p, g = int("0xFFFFFFFFFFFFFFFFC90FDAA22168C234C4C6628B80DC1CD1\
#29024E088A67CC74020BBEA63B139B22514A08798E3404DD\
#EF9519B3CD3A431B302B0A6DF25F14374FE1356D6D51C245\
#E485B576625E7EC6F44C42E9A637ED6B0BFF5CB6F406B7ED\
#EE386BFB5A899FA5AE9F24117C4B1FE649286651ECE65381\
#FFFFFFFFFFFFFFFF", 0), 5 # Oakley group (RFC 2409)
p, g = 1000000007, 5
q = p - 1
pp = PublicParams(p, q, g)
print("p =", pp.p)
print("q =", pp.q)
print("g =", pp.g)

# Anamorphic Parameters
l = 100
s = 100
t = 100
ap = AnamParams(l, s, t)
print("l =", ap.l)
print("s =", ap.t)
print("t =", ap.s)

# Keys Generation
kp = Gen(pp)
dk = aGen(pp, ap, kp.pk)
print("(sk, pk) = (%d, %d)" % (kp.sk, kp.pk))
print("K =", dk.K)
print("T = [", ", ".join(str(a) + "->" + str(b) for (a,b) in \
    sorted([((pp.g ** i) % pp.p, i) for i in range(l)])), ’]’)

# Testing aEnc -> Dec and aEnc -> aDec
msg = random.randint(1, pp.p - 1)
cm = random.randint(0, l - 1)
#ctr = [0, 0]
for i in range(runs):
    #c, ctr = aEncCtr(pp, dk, msg, cm, ctr)
```

```
    ctx = aEnc(pp, ap, dk, msg, cm)
    msg_ = Dec(pp, kp.sk, ctx)
    cm_ = aDec(pp, ap, dk, ctx)
    print("(%d, %d) -> aEnc -> (%d, %d) -> Dec -> %d" \
        % (msg, cm, ctx[0], ctx[1], msg_))
    print("(%d, %d) -> aEnc -> (%d, %d) -> aDec -> %d" \
        % (msg, cm, ctx[0], ctx[1], cm_))

# Testing Enc -> Dec and Enc -> aDec
for i in range(runs):
    m = random.randint(1, pp.p - 1)
    ctx = Enc(pp, kp.pk, m)
    msg_ = Dec(pp, kp.sk, ctx)
    cm_ = aDec(pp, ap, dk, ctx)
    print("%d -> Enc -> (%d, %d) -> Dec -> %d" \
        % (m, ctx[0], ctx[1], msg_))
    print("%d -> Enc -> (%d, %d) -> aDec -> %d" \
        % (m, ctx[0], ctx[1], cm_), "(!)" if cm_ != -1 else "")
```