CARIBE: Adapting Traditional IBE for the Modern Key-Covetous Appetite

Britta Hale Christopher Carr Danilo Gligoroski

britta.hale, chris.carr, danilo.gligoroski @item.ntnu.no

Norwegian University of Science and Technology,
Department of Telematics

Abstract. Current issues with mass surveillance and a lack of end-user encryption, coupled with a growing demand for key escrow under legal oversight and certificate authority security concerns, raises the question of the appropriateness of continued general dependency on PKI. Under this context, we examine Identity-Based Encryption (IBE) as an alternative to public-key encryption, contrasting the two in light of present interests. Balancing goals of encryption and (limited) key escrow, we compare IBE schemes involving multiple private-key generators (PKGs) where trust is not placed in a single entity, but spread evenly amongst many. Finally, we present CARIBE, a cascaded IBE scheme, and prove the security of it in the computational model. CARIBE combines the ease-of-use of IBE with key escrow, limited to the case when the entire set of participating PKGs collaborate. As a focal point, due to its construction, CARIBE also allows for maximum flexibility and user-side control of which PKGs should be allowed into the web of trust.

Keywords: Identity-based encryption, cascade ciphers, PKI, practice-oriented provable security, mass-surveillance

1 Introduction

Mass surveillance has undoubtedly formed one of the most contentious turning points in modern Internet history. Fueled by the revelation of the **PRISM** surveillance program in 2013 [17], which collected Internet data from some of the biggest operators, including Microsoft™, Google™, and Yahoo™, and subsequent knowledge of programs such as **xkeyscore** [16] for collecting data en-mass as it is transferred, the essential need for encryption has never been more salient. Simultaneously, the case for *backdoored* encryption is also being argued, as governments fight for control of, potentially vital, information for stanching terrorist threats. Under this context we raise the question of an optimal key management infrastructure, arguing for identity-based encryption (IBE) as an alternative to the current public key infrastructure (PKI) to increase the ease of encryption use, while also presenting a new prospective on IBE – CARIBE – which provides limited key escrow and allows for end-user control of encryption.

Fundamentally, a PKI system relies upon the receiver's precaution; whether or not they have set up a public/private key pair with the public key listed securely online for access by others who wish to communicate with them. Since powerful organizations can subvert these *secure* key directories [25] and most end users demonstrate a general apathy for establishing such keys to begin with [34], there is a clear need for a change to the conventional method. What the exigency of the situation demands is a system that provides secure ease-of-use encryption, does not rely upon a trusted third party, and yet allows for limited key escrow subject to the constraints of law. Notably, the need for limited key escrow is an axiomatically oblique path from the hither-to performed mass surveillance and pressure for backdoored products. As was recently stated in an open letter by U.S. congressmen Hurd and Lieu to the James Comey, Director of the FBI, "There is a difference between private companies assisting law enforcement and the government compelling companies to weaken their products..." [19].

Historically, IBE has been proposed and discussed as a means of encryption that is particularly user-friendly, as it is not necessary to have public or private keys established [32], yet it traditionally demands a trusted third party, namely a private-key generator (PKG). Developments in IBE have allowed for some preventative measures against a malicious PKG by distributing the key generation duties among multiple PKGs [23], but these designs demand that all involved PKGs must use the same IBE scheme which in turn could lead to a monopoly of the system. Realistically, it is even possible for groups of PKGs with IBE schemes in common to form coalitions, making collusion between PKGs easier and consequently increasing the risk of exposure for encrypted data. To our knowledge, no existing adaptation realizes the IBE ease-of-use, eliminates the demand for trust in a single

third party, allows free choice in the combination of PKGs and thereby encryption schemes, and only provides key escrow under collaboration of the entire set of PKGs involved.

In order to address cogent issues in key management for data encryption, we propose applying IBE through the interface of cascade encryption, galvanizing it into a realistic scheme for response to modern issues. Even while allowing key escrow in the most extreme circumstances, and demanding no less than the participation of all key generators to achieve it, the freshly-interfaced IBE provides far greater power to the end user for selecting a trust model than has been previously proposed. Merging such varied cryptographic areas precisely reflects the spirit of the IACR *Copenhagen Resolution* [20]:

> Population-wide surveillance threatens democracy and human dignity. We call for expediting research and deployment of effective techniques to protect personal privacy against governmental and corporate overreach.

Markedly other, similar, infrastructures also exist, such as attribute-based encryption (anyone in possession of the correct set of attributes can decrypt) and, more generally, predicate-based encryption (anyone in possession of the correct set of attributes and a decryption key corresponding to a certain predicate can decrypt). While these are certainly interesting, we focus on IBE due to its resemblance to PKI, with encryption based upon fixed identities. However, it is straightforward to envisage that our results can be adapted for both attribute-based encryption and predicate-based encryption, with the same freedom-of-choice benefit of the trust model maintained for the end user.

1.1 Outline and Contributions

Initializing our work, we consider underlying problems facing PKI for key management in §2.1, developed under the perspective of modern threats to Internet security. We provide a clear assessment of IBE as an alternative to PKI, with comparison on the strengths and weaknesses provided by each; for example, key escrow, effects from a compromised key management authority, etc.

After describing the existent pitfalls in PKI and the IBE alternative, attention is turned to the state of the art of IBE (§2.2), namely IBE composition schemes which aim to mitigate the disadvantages of IBE and the particular security guarantees they provide – such as hierarchical IBE (HIBE) [18, 15] and the distributed PKG system [10, 23]. Ultimately, current IBE schemes fall short of fully capitalizing on the potential of IBE over PKI, due largely to the handling of key escrow and real-life trust models. Essentially, HIBE places key escrow in the hands of one or more entities who may decrypt private communications at will or provide such power to a third party, amplifying the already existent problem of trust in the realm of Internet security. Meanwhile, distributed IBE attempts to spread the issue of key escrow across multiple PKGs by assuming a (possibly large) honest subset [10, 23].

In consequence of issues that still inhibit usable realization of IBE, we extend beyond our analysis by presenting a new scheme (§4) – Cascade-realized IBE (CARIBE). Succinctly, CARIBE applies a cascade of encryption via multiple PKGs to eliminate the single point of failure that is inherent in traditional IBE, while maintaining the potential for key escrow, reducing the mandated set of honest PKGs to size one, and not limiting the selection choice of schemes involved. CARIBE, unlike distributed IBE where all PKGs must operate under the same scheme, also allows for the interaction of multiple IBE schemes. Notably, this benefit is not to be under-estimated in the modern real-world context. Even as each PKG has freedom to use whatever IBE scheme it desires, a sender may either select PKGs based upon the combination of options given or upon a trust (or mutual distrust) foundation without regard to matching the IBE schemes being used. Thus, a sender may feasibly select rival PKGs to reduce the chances of collusion; for instance, PKGs implemented by and operating under the standards of competitive world powers [28, 30]. Still, the option for key escrow allows for a third party to obtain decryption keys under sufficiently compelling circumstances, such as through legal procedures. Thus all n PKGs selected by the sender would need to collaborate with such a third party to be able to decrypt. Similarly to emerging new ideas, such as Bitcoin, our proposal collects widely known and tested concepts and uses them in a novel way, expanding the horizons of IBE.

Additionally, we provide a clear analysis framework by presenting the IND-ID-CCA experiment for IBE in concise pseudo-code (§3). Furthermore, we define IND-ID-C.CCA for CARIBE and thereby provide a framework for cascade analysis for IBE (§4). We clearly state the C.CCA assumption for cascade encryption schemes, defining the conditions for CCA analysis that preclude the trivial cascade attacks of [11]. Utilizing this defined security structure, we analyze CARIBE, showing that security reduces to that of any one component scheme (§4).

1.2 Related Work

Identity-Based Cryptography Identity-based cryptography has seen a considerable amount of development since it was first envisaged by Shamir in the mid 1980's [32]. Initial examples of schemes for identity-based cryptography are due to Cocks [9], Boneh and Franklin [5], and Sakai, Ohgishi and Kasahara [31]. Today, there are entire books devoted to the subject [22, 8], as well as an extensive amount of research in various aspects of the field (for example, [5–7, 9, 10, 15, 18, 23, 31, 32, 35]).

Shamir argued for a public key encryption system conceptually similar to the postal system, albeit idealized, where a sender needs only the name and address of the recipient [32]. "IBE is a kind of public key encryption scheme where the public key of a user can be any arbitrary string – typically the e-mail address" [8], i.e. the recipient's identity, such as name, location, etc., becomes their public key. Enticingly, this approach offers some substantial advantages over the traditional PKI (see §2.1). Among the attributes the make IBE advantageous is its suitability for situations where network access is not continuous. Furthermore, and perhaps one of the more notable advantages of identity-based cryptography, is the nullification of the need for certificates and thereby the instantaneousness with which encryption can be performed without the requirement to obtain such a certificate. Joux [21] provides a broad, non-specific introduction to identity-based cryptography, relating it with other common public key practices. Examples of current IBE frameworks for scheme proposals include Full-Domain-Hash IBE, Exponent-Inversion IBE, and Commutative-Blinding IBE [6].

In 2008, some research was performed into the security of IBE when decryption keys are generated under the same ID from multiple PKGs [29]. Naturally, such investigation and the security results therefrom are highly relevant to this work, and would further support the security analysis of §4. However, as we do not limit our cascades to the use of one encryption scheme, the results of [29] would need to be greatly expanded before usage with CARIBE.

While the wider field of identity-based cryptography is of great interest, throughout this paper our focus will be on key management in the context and, in particular, identity-based encryption. Thus, we will generically refer to IBE and key management for IBE schemes. Identity-based signatures present another intriguing aspect of the cryptographic field that we foresee our work being highly relevant to, although we maintain a focus here on encryption. Specifically, CARIBE-esque cascaded identity-based signature schemes, under the motivation of Shamir's original identity-based signatures [32], could provide similar benefits as those that we have described for CARIBE (§1.1).

Cascade Encryption Cascade encryption, or sequential multiple encryption, is the concept of layering encryption such that the ciphertext from one encryption step is the plaintext of the next. Essentially, an n-fold cipher cascade of Encrypt algorithms takes in a message m and outputs

$$(\text{Encrypt}_{k_n} \circ \cdots \circ \text{Encrypt}_{k_1})(m) \ .$$

Ways of ealizing a cascade cipher include increasing the number of rounds of a cipher, cascading encryption under different keys, and cascading actual ciphers. In the context of IBE in this paper, the latter is of particular interest due to the potential benefits of employing various PKGs. Essentially, while multiple cipher rounds may be generally beneficial for security, it is the benefit of key escrow without mandatory trust in one PKG that makes cascade encryption attractive the context of IBE.

Previous work focusing on cascade encryption includes that of Even and Goldreich [12] which proved that in a 2-fold block-cipher cascade, the security of the cascade was reducible to the security of either of the component ciphers. Later work showed that the security of an n-cipher cascade was reducible to that of the first cipher in the cascade [27], that triple-encryption (3-fold cipher cascade) for block ciphers provides a security improvement over single- or double-encryption [4, 13], and describe generic CCA security for multiple encryption [11]. Furthermore, it has been information-theoretically demonstrated that an n-fold cascade of pseudo-random permutations (PRPs), for which the computational distinguishing advantage is bounded by $\epsilon < 1$ (ϵ-PRP), yields a $((n-(n-1)\epsilon)\epsilon^n+\nu)$-PRP for negligible function ν [33].

Despite the extensive research on multiple and cascade encryption, the application of an n-fold IBE-cipher cascade has not been addressed, nor have security considerations (CCA, etc.) been considered in this context. Since IBE already presents a possible alternative to PKI with some alluring benefits, the security of cascade encryption composed of IBE schemes, and the exact manner in which such a cascade can be realized, is particularly interesting. Moreover, cascade encryption with IBE goes beyond the encryption itself – in such a context, it is essential to consider collusion among the private key generators (PKGs) involved.

2 Public Key Management

Identity-based cryptography falls within the scope of public key cryptography. Currently, public key systems rely almost completely on *certificate authorities* (CAs), employing certificate chaining to distribute, assert, and prove ownership of public keys. Popularly, this system is referred to as PKI [24]. Mao [26], as well as Katz and Lindell [24], offers a good foundational overview of PKI. Structurally, identity-based cryptography diverges considerably from PKI and, as such, comes with certain advantages and disadvantages, particularly relating to key management.

2.1 Comparison of IBE with PKI

Table 1 summarizes a brief comparison of the main benefits and drawbacks of both IBE and PKI, followed by a discussion of the properties.

Architecture	PKI	IBE
Key management authority	CA	PKG
Key Escrow	No	Yes
Certificate management	Yes	No
Non-interactive authentication	No	Yes
Always encrypt	No	Yes
Compromise of management authority is fatal	No	Yes

Table 1: Comparison of properties between PKI and IBE.

Key management authority refers to the authority in control of managing, certifying, and/or distributing private and/or public keys. In PKI, key management is performed via a CA, of which there are many in the PKI system, with most users trusting some subset. For a user U to state ownership of their public key, the user requests a CA (of their choice) to sign their public key; with response computed under the signature algorithm $\text{Sign}(sk_{\text{CA}}, pk_U)$. When signing, the CA asserts that the identity of user U is what they claim it to be. Anyone who trusts the signing CA and knows the CA's public key verify this with a verification algorithm Verify. For correctness, it is required that $\text{Verify}(pk_{\text{CA}}, \text{Sign}(sk_{\text{CA}}, pk_U)) = 1$. In practice, there is a small selection of very high level CAs,called root CAs, who sign the public keys of other, intermediate, CAs. Continuing, intermediate CAs sign certificates of other, lower-level, intermediate CAs and so on, until the certificate of the end user is signed. This creates a chain of trust, where if someone trusts at least one CA in a signature chain, and has its public key, they can trace and verify down the chain of trust to the end node they wish to communicate with. One concern is obtaining the public keys of the trusted CAs in the first place. Practically, these public keys are often included in software, such as the operating system or browser [24].

In IBE, the PKG is responsible for generating and distributing the private key for the user U, derived from a master secret key kept solely by the PKG. The public key and identity of the receiver are one and the same [22]; thus no interaction with the PKG is required in order to send a message to a user.

Key Escrow effectively refers to the capacity of the key management authority to know the private keys of the users in their system. In PKI, there is no key escrow as the trusted CA signs the public key of the user and does not have access to the private key. Comparatively, IBE has key escrow as the PKG actually computes private keys for users. Generally, key escrow is considered as an undesirable property, as it places an enormous amount

of trust in the discretion of a single authority to not decrypt messages or pass on that ability to third parties. Thus, there are many scenarios where key escrow is anything but damning.

Avoiding key escrow in IBE is non-trivial (Chatterjee and Sarkar [8] devote a book chapter to the issue), with various methods developed that attempt to sidestep this drawback [2, 14, 18]. However, while there have been some attempts at avoiding it entirely, there is argument for the virtue of key escrow, particularly in the modern context when governments wish to legally access encrypted data. Broadly speaking, key escrow has a place so long as it is constrained and managed appropriately; for example, where the private data of an individual can only be accessed under juristical permission.

Certificate management refers specifically to the implementation, management, and verification of signed certificates, used for the certification of public key information of principals within the system. We refer to Adam, Farrell, Kause, and Mononen [1] for a full technical description. In the common PKI implementations, certificates need to be verified before the public key can be trusted. Thus sending a message requires two stages from the sender: a verification of the user's public key and encryption under that key. As IBE does not involve certificates, the processes of managing and checking them are not required.

Non-interactive authentication is closely related to certificate management. With IBE, the sender does not require external verification of the intended recipient Clearly, this approach is far more efficient over certificate management and interactive certificate verification of PKI, from the perspective of the sender. Additionally, it allows for ease-of-use for the recipient, as no set-up work is required to authenticated themselves to potential senders.

Always encrypt refers to the ability to encrypt a message to the receiver, even if they have not taken the preliminary step to set up a public/private key pair. With IBE, the ability to always encrypt allows a sender to immediately establish an encrypted communication channel without interaction [26]. Notably, it is even possible send an encrypted message to an identity that has not yet been registered with a PKG and consequently; in this case, the receiver would have to post-register with the PKG in order to decrypt the message. Currently the option to always encrypt for any identity is not possible in PKI, as public and private keys are created simultaneously and must be set up in advance. One attractive consequence of the *always encrypt* feature is the possibility that the sender of a message can choose a PKG that they trust, regardless of whether the receiver is managed by that PKG. Then the onus is on the receiver to prove their legitimacy to the PKG in order to decrypt the message. Allowing the sender of the message control over the choice of PKG, and by extension the IBE scheme, is an interesting consequence of IBE. For applications, this adds flexibility and control to the party wishing to transmit a message, and so is advantageous over current PKI which relies on the forethought of the receiver.

Compromise of management authority is fatal refers to the effect on the security of the past encryptions of the end user of the system, in the case of a compromised management authority. In IBE, a compromised PKG can generate any user's secret key for any time, meaning all encrypted communication to a user registered under the corrupted PKG is readable by that PKG, or anyone holding its keys. From a current PKI perspective, the consequences are not as dire, as all previous communications up until the compromise remain indecipherable. Once compromised, the CA can revoke the legitimate user's certificate and issue one for a potentially malicious imposter, declaring that this imposter is the legitimate user. The imposter can now intercept and read communications intended for the user, so long as the compromised CA remains trusted to the parties wishing to communicate. Consequently, deploying standard IBE, in the traditional single PKG context, is difficult to advocate for on a large scale, especially without a way of overcoming this attractive attack vector.

2.2 Comparison of IBE architectures

Schemes relying on a single PKG come with unfortunate drawbacks, as highlighted in § 2.1, making them unattractive in the context of global deployment. One possibility for mitigating this problem is to employ an IBE system that uses multiple PKGs. Here, the motivation is that by using multiple PKGs, the issue of single PKG compromise and key escrow is allayed simultaneously, whist still providing the advantageous properties of non-interactive authentication and always encrypt. Essentially, master or user decryption keys of a single PKG should be inadequate for decrypting past communications without keys from the other PKGs involved. Consequently this section examines current proposals for IBE schemes employing more than one PKG, providing an overview of these architectures and highlighting the properties they possess.

Hierarchical IBE One popular multi-PKG architecture is hierarchal identity-based encryption (HIBE). Originally envisaged by Horwitz and Lynn, HIBE has parallels with the hierarchical nature of current PKI [18]. Like PKI (§2.1), it is comprised of root nodes, intermediate nodes, and users. Informally, the root PKG holds the master secret key $\mathsf{masterkey}$, while an intermediate PKG holds its own identity ($\mathsf{ID}_{\text{PKG.i}}$) and must request their own secret key from the root PKG. Similarly, the user has an identity ($\mathsf{ID}_{\text{user}}$) and requests its secret key from the intermediate node. Keys at each stage are derived from functions on the keys at the higher level, as demonstrated in the following generalization for a hierarchy of n PKGs [18].

$$
\begin{aligned}
&\text{Root :} && f_1(\mathsf{masterkey}, \mathsf{ID}_{\text{PKG.2}}) = dkey_2 \rightarrow \text{Intermediate PKG.2} \\
&\text{Intermediate PKG.2 :} && f_2(dkey_2, \mathsf{ID}_{\text{PKG.3}}) = dkey_3 \rightarrow \text{Intermediate PKG.3} \\
&\vdots \\
&\text{Intermediate PKG.}n\text{ :} && f_n(dkey_n, \mathsf{ID}_{\text{User}}) = dkey_{\text{User}} \rightarrow \text{User}
\end{aligned}
$$

It is worth noting that the functions f_i, for $i \in \{1, n\}$, are known and that every PKG in the hierarchy, excepting the user, may have multiple descendants.

HIBE offers the advantage of reducing the workload of root PKGs. In a large infrastructure scenario, the number of requests for private keys could quickly grow too great for one PKG to manage; allowing for subordinate PKGs helps to alleviate that problem. Another aspect is that key escrow for the end user is available at every level [18]. Of course, from a modern perspective, this raises the fundamental concern that not only does the root PKG have the ability to generate the secret keys for all users, making it a prime target for coercion, but in fact any compromised PKG in the hierarchy can generate the secret key of all descendant users from itself.

Notably, Gentry and Silverberg offered an improved HIBE scheme, presenting an instantiation of HIBE that is CCA-secure under the Bilinear Diffie-Hellman Problem and collusion resistant [15]. Other extensions of hierarchical IBE schemes exist, such as Multi-HIBE and Anonymous-HIBE. Multi-hierarchical offers forward security [35] and anonymous-hierarchical offers anonymous communication between sender and receiver [7]. These techniques withstanding, HIBE still fails to address certain matters in focus. Multi-HIBE is restricted by demanding trust in a majority of root nodes, and requires that compromise or collusion of all ancestor PKGs for a user doesn't happen, as compromise of a user's ancestor PKGs implies complete exposure of their secret key and the secret keys of all users with a subset of those ancestors In a truly global system, the effects of this would be a devastating. Anonymous-HIBE is an interesting approach, but it still falls short of our goals. Anonymity from the PKGs is not an ideal attribute, instead, a system where the user can be accountable under certain circumstances, such as by a legitimate legal mandate, but is otherwise protected from scrutiny, is far more favorable.

Certificateless Public Key Cryptography Certificateless public key cryptography (CL-PKC), as explained by Al-Riyami and Paterson [2], was developed with the aim of finding public key schemes that are not dependent on certificates and do not have the key escrow property. As CL-PKC is claimed as an intermediate between standard PKI and the identity-based variant, we describe how it works on a high level.

CL-PKC requires two parties to generate public and private keys, where one is the end user. The PKG in this instance has a known public key ID_{PKG} and a master secret key. The user U has an identity $\mathsf{ID}_{\text{User}}$ and some secret information SecInfo–U known only to themselves, with the public key and the secret key generated from these parameters.

$$
\begin{aligned}
&\text{PKG :} && f_1(\mathsf{masterkey}, \mathsf{ID}_{\text{User}}) = dkey_{\tilde{U}} \rightarrow \text{User} \\
&\text{User :} && f_2(dkey_{\tilde{U}}, \text{SecInfo–}U) = dkey_U \\
&\text{User :} && f_3(\text{SecInfo–}U, \mathsf{ID}_{\text{PKG}}) = pkey_U
\end{aligned}
$$

Again, the functions f_i are assumed to be known for all i, and $dkey_{\tilde{U}}$ represents the partial decryption key for user U, which must be combined with the users secret information to form the decryption key. Generation of the public key can still be done prior to generation of the private key. Note that the public key is not computable from the identity of the user and therefore must also be pre-computed and made available publicly (though verification of the public key is no longer required). Consequently, due to the nature of this public key derivation, CL-PKC is no longer identity-based [2] and lacks the major advantage of the identity-based paradigm which allows any-time encryption without the receiver having to preform any set up. Another important issue with this system is that the user controls their own secret information and can therefore refuse to reveal the content of

their encrypted messages, even when juridically enjoined – an ability that matched the goals of CL-PKC, but falls short of contemporary considerations. As previously expressed, key escrow is an important piece of the public key deployment puzzle under present-day concerns, provided it is managed correctly.

Distributed IBE In parallel our argument in §2.1, Joux [21] advocates for a system with many, independent PKGs for nullifying the issue of compromise of a single PKG: "Such a scheme could mitigate the trust issues, at the cost of making the private key generation step heavier ..." Distributed IBE, as formally defined by Kate and Goldberg [23] does precisely that, with a form of threshold trust. Informally, an IBE scheme of n PKGs is (n, t)-distributed if no collusion of $x \leq t$ of the PKGs can compute the master key, for some threshold value $t \leq n$, where all n PKGs contain a share of a user's private key. In the distribution model, the n PKGs share parts of one master secret key.

While distributed IBE addresses the issue of key escrow and PKG compromise somewhat in-line with our goals a key issue with this method is that no allowance is made for PKGs using different IBE schemes. Consequently, as it would not be possible to share secrets across PKGs using different IBE schemes, the effectiveness of distributed IBE is limited. Applicability to the real world, and competing interests, means that the assumption of global agreement on one IBE scheme seems unlikely. Indeed, we conjecture that it is more likely that the organizations managing PKGs will use different schemes, many of which may be proprietary and not openly available for examination. Correspondingly, the forced use of specific, distributed IBE schemes may well lead to new, yet secure and efficient, IBE schemes being overlooked, even if their properties are highly desirable. In response to these concerns, we formally propose an IBE architecture, CARIBE, in the following section.

3 Identity-Based Encryption

Formally, we present the definition of an IBE scheme which serves as a grounding point for the work in this section. From IBE schemes, we build CARIBE – the scheme contribution of this paper – using generic, yet unspecified, number of n PKG.

Definition 1 (Identity-Based Encryption [10]). *Under a Private Key Generator (PKG), an identity-based encryption scheme $IBE.\mathcal{E}$ is a tuple of algorithms:*

- $\text{Setup}(\lambda) \xrightarrow{\$} (\mathsf{params}, \mathsf{masterkey})$*: A probabilistic setup generation algorithm that takes as input a security parameter λ and outputs parameters* params *and a PKG master key* masterkey*.*
- $\text{Extract}(\mathsf{params}, \mathsf{masterkey}, \mathsf{ID}) \xrightarrow{\$} dkey$*: A probabilistic extraction algorithm that takes as input system parameters* params*, a PKG master key* masterkey*, and a public identity string* ID*, and outputs a decryption key dkey.*
- $\text{Encrypt}(\mathsf{params}, \mathsf{ID}, m) \xrightarrow{\$} c$*: A probabilistic encryption algorithm that takes as input system parameters* params*, a public identity string* ID*, and a message $m \in \mathcal{M}$, and outputs a ciphertext $c \in \mathcal{C}$.*
- $\text{Decrypt}(\mathsf{params}, c, dkey) \xrightarrow{\$} m$*: A possibly probabilistic decryption algorithm that takes as input system parameters* params*, a ciphertext $c \in C$, and a private decryption key dkey, and outputs either a message $m \in \mathcal{M}$ or an error symbol $\perp$.*

In addition, it is required that if $dkey \leftarrow \text{Extract}(\mathsf{params}, \mathsf{masterkey}, \mathsf{ID})$*, then*

$$\forall m \in \mathcal{M} : \text{Decrypt}(\mathsf{params}, \text{Encrypt}(\mathsf{params}, \mathsf{ID}, m), dkey) = m \ .$$

As an accepted assessment of security for public key encryption schemes, IND-CCA is also the criterion for security in the layered PKG setting. While IND-CCA security has been extensively handled before [3], and even the particular case of IND-ID-CCA for IBE described [10], a clear, formalized, pseudo-code definition for IND-ID-CCA has been lacking. Consequently, we unambiguously delineate the IND-ID-CCA experiment and adversary win conditions, corresponding to Definition 2, in Fig. 1. Notationally, we let Π denote the protocol employed by the PKG.

Definition 2. *Let Π be an identity-based encryption scheme according to Definition 1 and let $\mathcal{A}$ be an adversary algorithm. Then, for the IND-ID-CCA experiment given in Fig. 1,*

$$\mathsf{Adv}^{\text{IND-ID-CCA}}_{\Pi,\mathcal{A}} = |\Pr[b = b'] - {}^1\!/_2| \ .$$

$\underline{\mathsf{Exp}^{IND-ID-CCA}_{\Pi,\mathcal{A}}():}$

1: $(\mathsf{params}, \mathsf{masterkey}) \xleftarrow{\$} \text{Setup}(\lambda)$
2: $\mathsf{ID.list_{ext}} \leftarrow \perp$
3: $\mathsf{ID.list_{enc}} \leftarrow \perp$
4: $S \leftarrow \emptyset$
5: $b \xleftarrow{\$} \{0,1\}$
6: $\mathcal{A}^{\text{Extract}(\cdot),\text{Encrypt}(\cdot),\text{Decrypt}(\cdot),\mathsf{params}}()$
7: $b' \xleftarrow{\$} \mathcal{A}^{\text{Extract}(\cdot),\text{Encrypt}(\cdot),\text{Decrypt}(\cdot),\mathsf{params}}$

$\underline{\text{Oracle Extract}(\mathsf{ID}_i):}$

1: **if** $\mathsf{ID}_i \in \mathsf{ID.list_{enc}}$ **then**
2: **return** $\perp$
3: $dkey_i \leftarrow \text{Extract}(\mathsf{params}, \mathsf{masterkey}, \mathsf{ID}_i)$
4: $\mathsf{ID.list_{ext}} \leftarrow \mathsf{ID.list_{ext}}|\mathsf{ID}_i$
5: **return** $dkey_i$

$\underline{\text{Oracle Decrypt}(\mathsf{ID}_i, c_i):}$

1: **if** $(\mathsf{ID}_i, c_i) \in S$ **then**
2: **return** $\perp$
3: $dkey_i \leftarrow \text{Extract}(\mathsf{params}, \mathsf{masterkey}, \mathsf{ID}_i)$
4: $m_i \leftarrow \text{Decrypt}(\mathsf{params}, c_i, dkey_i)$
5: **return** m_i

$\underline{\text{Oracle Encrypt}(\mathsf{ID}, m_0, m_1):}$

1: **if** $\mathsf{ID} \in \mathsf{ID.list_{ext}}$ **then**
2: **return** $\perp$
3: $c^{(0)} \leftarrow \text{Encrypt}(\mathsf{params}, \mathsf{ID}, m_0)$
4: $c^{(1)} \leftarrow \text{Encrypt}(\mathsf{params}, \mathsf{ID}, m_1)$
5: **if** $c^{(0)} = \perp$ or $c^{(1)} = \perp$ **then**
6: **return** $\perp$
7: $c_u := c^{(b)}$
8: $S \leftarrow S \cup \{(\mathsf{ID}, c_u)\}$
9: $\mathsf{ID.list_{enc}} \leftarrow \mathsf{ID.list_{enc}}|\mathsf{ID}$
10: **return** c_u

Fig. 1: IND-ID-CCA Experiment for $IBE.\mathcal{E}$.

4 Cascade-realized IBE – CARIBE

Employing a generic, finite number of n IBE schemes, as defined above, we describe Cascade-realized Identity-Based Encryption (CARIBE), an n-fold IBE-cipher cascade, in Definition 3. Saliently, the CARIBE definition does not restrict the type of encryption schemes being used. As a result, a CARIBE scheme $CARIBE.\mathcal{E}$ could literally consist of PKGs using n distinct IBE schemes, if such variation is desired or deemed necessary. Essentially, a CARIBE scheme works with cascaded encryption; however, unlike general cascade encryption schemes, multiple PKGs are involved and encryption is sequentially performed using the parameters params generated by each of them.

In the following definition it is required that each generated ciphertext is in the plaintext space of the next cipher in the encryption cascade. Simply interpreting plaintext and ciphertext spaces as $\{0,1\}^*$ allows for convenient completion of this requirement.

Definition 3 (Cascade-realized Identity-Based Encryption (CARIBE)). *Under n Private Key Generators (PKG), a cascade-realized identity-based encryption scheme $CARIBE.\mathcal{E}$ is a tuple of algorithms that takes n IBE encryption schemes Π_i, for $i \in \{1, \ldots n\}$, with the restriction that $\mathcal{C}_i \subseteq \mathcal{M}_{i+1}$ for $i \in \{1, \ldots n-1\}$, i.e. that the ciphertext of each IBE scheme is in the plaintext space of the next IBE scheme in the cascade. Algorithms for $CARIBE.\mathcal{E} := C(\Pi_1, \ldots, \Pi_n)$ are as follows:*

- $\text{Setup}_S(\lambda_1, \ldots, \lambda_n)$:
  1: **for** $i \in \{1, \ldots, n\}$ **do**
  2: $(\mathsf{params}_i, \mathsf{masterkey}_i) \leftarrow \text{Setup}_i(\lambda_i)$
  3: **return** $(\{\mathsf{params}_i\}, \{\mathsf{masterkey}_i\})$

  A probabilistic setup generation algorithm that takes as input an ordered sequence of n security parameters λ_i and outputs an ordered set of n parameters $\{\mathsf{params}_i\}$ and an ordered set of n PKG master keys $\{\mathsf{masterkey}_i\}$, for $i \in \{1, \ldots n\}$.
- $\text{Extract}_S((\{\mathsf{params}_i\}, \{\mathsf{masterkey}_i\}), \mathsf{ID})$:
  1: **for** $i \in \{1, \ldots, n\}$ **do**
  2: $dkey_i \leftarrow \text{Extract}_i(\mathsf{params}_i, \mathsf{masterkey}_i, \mathsf{ID})$
  3: **return** $\{dkey_i\}$

  A probabilistic extraction algorithm that takes as input a set of n system parameters $\{\mathsf{params}_i\}$, a set of n PKG master keys $\{\mathsf{masterkey}_i\}$, and a public identity string ID, and outputs a set of n decryption keys $\{dkey_i\}$, for $i \in \{1, \ldots n\}$.

– $\text{Encrypt}_S(\{\mathsf{params}_i\}, \mathsf{ID}, m)$:

1: $c_2 \leftarrow \text{Encrypt}_1(\mathsf{params}_1, \mathsf{ID}, m)$
2: **for** $i \in \{2, \ldots, n\}$ **do**
3: $\quad c_{i+1} \leftarrow \text{Encrypt}_i(\mathsf{params}_i, \mathsf{ID}, c_i)$
4: $c \leftarrow c_{n+1}$
5: **return** c

A probabilistic encryption algorithm that takes as input an ordered set of n system parameters $\{\mathsf{params}_i\}$*, for* $i \in \{1, \ldots n\}$*, a public identity string* ID*, and a message* $m \in \mathcal{M}$*, and outputs a ciphertext* $c \in \mathcal{C}$*. The plaintext space of* Encrypt_S *is* $\mathcal{M} := \mathcal{M}_1$*, the plaintext space of* Encrypt_1*, while the ciphertext space of* Encrypt_S *is* $\mathcal{C} := \mathcal{C}_n$*, the ciphertext space of* Encrypt_n*.*

– $\text{Decrypt}_S(\{\mathsf{params}_i\}, c, \{dkey_i\})$:

1: $c_n \leftarrow c$
2: $i \leftarrow n$
3: **while** $i > 0$ **do**
4: $\quad c_{i-1} \leftarrow \text{Decrypt}_i(\mathsf{params}_i, c_i, dkey_i)$
5: $\quad i \leftarrow i - 1$
6: $m \leftarrow c_0$
7: **return** m

A possibly probabilistic decryption algorithm that takes as input an ordered set of n system parameters $\{\mathsf{params}_i\}$*, a ciphertext* $c \in C$*, and an ordered set of n private decryption keys* $\{dkey_i\}$*, and outputs either a message* $m \in \mathcal{M}$ *or an error symbol* $\perp$*.*

In addition, it is required that if $\{dkey_i\} \leftarrow \text{Extract}_S(\{\mathsf{params}_i\}, \{\mathsf{masterkey}_i\}, \mathsf{ID})$*, then*

$$\forall m \in \mathcal{M} : \text{Decrypt}_S(\{\mathsf{params}_i\}, \text{Encrypt}(\{\mathsf{params}_i\}, \mathsf{ID}, m), \{dkey_i\}) = m \ .$$

4.1 Security of CARIBE

CARIBE is essentially a type of cascade encryption scheme and as such presents a challenge in the context of CCA security. As noted in [11], any encryption cascade fails to provide CCA security, since an adversary that is possession of the key of the outermost encryption layer could simply decrypt that layer and re-encrypt, yielding a new ciphertext to call the decryption oracle on, trivially breaking the CCA security. However, this is based upon the assumption that an adversary already has the key for the outermost encryption, and assuming such key possession is generally non-standard when considering CCA security. To avoid this fairly trivial break, we propose the following assumption when analyzing the CCA security of cascade schemes; a CCA security game based upon this assumption will be termed Cascade CCA (C.CCA):

C.CCA Assumption for Cascaded Encryption *For a cascaded encryption scheme, with ciphertexts generated as* $c := \text{E}_{K_n}(\ldots(\text{E}_{K_2}(\text{E}_{K_1}(m))))$*, and an adversary* $\mathcal{A}$*, CCA security is analyzed under the assumption that* $\mathcal{A}$ *does not possess* K_n*.*

Note that if $\mathcal{A}$ possesses the encryption keys for the outermost r layers, and can win a CCA security game on an encryption scheme cascade of the remaining $n - r$ layers, then $\mathcal{A}$ can certainly win the CCA game on the entire scheme. Thus, the layers can be "peeled back" until $\mathcal{A}$ does not possess the outermost decryption key. While making this assumption is not ideal for security analysis, it is necessary to avoid the trivial break mentioned above and allows for realistic analyses to still be performed for cascaded schemes.

In addition to the CARIBE scheme description, we present the tailored IND-ID-C.CCA experiment in Fig. 2. While most schemes are analyzed under a general security experiment, an experiment for CARIBE itself is required as any CARIBE scheme is in fact a particular cascade selection of other IBE schemes. Adversarial advantage for the experiment is described in Definition 4.

Definition 4. *Let C be an algorithm taking as input n identity-based encryption schemes* Π_j*, for* $j \in \{1, \ldots, n\}$*, according to Definition 1 and yielding a cascade-realized identity-based encryption scheme* $C(\Pi_1, \ldots, \Pi_n)$*, per Definition 3, comprised of a new tuple of algorithms* $(\text{Setup}_S, \text{Extract}_S, \text{Encrypt}_S,$
$\text{Decrypt}_S)$*. Let* $\mathcal{A}$ *be an adversary algorithm. Then, for the IND-ID-C.CCA experiment given in Fig. 2,*

$$\mathsf{Adv}^{\text{IND-ID-C.CCA}}_{C(\Pi_1,\ldots,\Pi_n),\mathcal{A}} = |\Pr[b = b'] - {}^1\!/_2| \ .$$

$\mathsf{Exp}^{IND-ID-C.CCA}_{C(\Pi_1,\ldots,\Pi_n),\mathcal{A}}()$:

1: $(\{\mathsf{params}_j\}, \{\mathsf{masterkey}_j\}) \xleftarrow{\$} \text{Setup}(\lambda_1, \ldots, \lambda_n)$
2: $\mathsf{ID.list_{ext}} \leftarrow \perp$
3: $\mathsf{ID.list_{enc}} \leftarrow \perp$
4: $S \leftarrow \emptyset$
5: $b \xleftarrow{\$} \{0, 1\}$
6: $\mathcal{A}^{\text{Extract}_S(\cdot),\text{Encrypt}_S(\cdot),\text{Decrypt}_S(\cdot),\{\mathsf{params}_j\}}()$
7: $b' \xleftarrow{\$} \mathcal{A}^{\text{Extract}_S(\cdot),\text{Encrypt}_S(\cdot),\text{Decrypt}_S(\cdot),\{\mathsf{params}_j\}}$

Oracle $\mathsf{Extract}_S(\mathsf{ID}_i, \{1, \ldots, n\})$:

1: **if** $\mathsf{ID}_i \in \mathsf{ID.list_{enc}}$ **then**
2: **return** $\perp$
3: $\{dkey_j\} \leftarrow \text{Extract}_S(\{\mathsf{params}_j\}, \{\mathsf{masterkey}_j\}, \mathsf{ID}_i)$
4: $\mathsf{ID.list_{ext}} \leftarrow \mathsf{ID.list_{ext}}|\mathsf{ID}_i$
5: **return** $\{dkey_j\}_i$

Oracle $\mathsf{Decrypt}_S(\mathsf{ID}_i, c_i, \{1, \ldots, n\})$:

1: **if** $(\mathsf{ID}_i, c_i) \in S$ **then**
2: **return** $\perp$
3: $\{dkey_j\}$
$\leftarrow \text{Extract}_S(\{\mathsf{params}_j\}, \{\mathsf{masterkey}_j\}, \mathsf{ID}_i)$
4: $m_i \leftarrow \text{Decrypt}_S(\{\mathsf{params}_j\}, c_i, \{dkey_j\})$
5: **return** m_i

Oracle $\mathsf{Encrypt}_S(\mathsf{ID}, m_0, m_1)$:

1: **if** $\mathsf{ID} \in \mathsf{ID.list_{ext}}$ **then**
2: **return** $\perp$
3: $c^{(0)} \leftarrow \text{Encrypt}_S(\{\mathsf{params}_j\}, \mathsf{ID}, m_0)$
4: $c^{(1)} \leftarrow \text{Encrypt}_S(\{\mathsf{params}_j\}, \mathsf{ID}, m_1)$
5: **if** $c^{(0)} = \perp$ or $c^{(1)} = \perp$ **then**
6: **return** $\perp$
7: $c := c^{(b)}$
8: $S \leftarrow S \cup \{(\mathsf{ID}, c)\}$
9: $\mathsf{ID.list_{enc}} \leftarrow \mathsf{ID.list_{enc}}|\mathsf{ID}$
10: **return** c

Fig. 2: IND-ID-C.CCA Experiment for $CARIBE.\mathcal{E}$.

Security for CARIBE depends upon the constituent IBE schemes involved, as each contributes to the security of the final ciphertext. As previously mentioned, it has been show that the encryption security of a 2-fold cascade is reducible to the security of either of the composite ciphers [12]. Consequently, not only do we focus on the expanded n-fold case, but take into consideration the possibility of collusion. On a logical level, the ciphertext cannot be decrypted even if $n-1$ of the n PKGs collude, so long as one PKG is honest. Essentially, this worst-case scenario would then be precisely equivalent to a basic, secure IBE scheme under an honest PKG. Thus, the distinction between a CARIBE and IBE scheme becomes one of existence; for IBE we demand that the IBE scheme in use is secure with an honest PKG, while for CARIBE it is only required that one such IBE scheme exists in the cascade. Succinctly, Theorem 1 provides a proof of this analysis, similar to that of [12] yet given in detail for IBE.

To enable the analysis, we operate in an execution environment with the following standard list of allowed adversarial queries:

- $\mathsf{Extract}_S(\mathsf{ID}_i, \{1, \ldots, n\})$: Operates as in Fig. 2.
- $\mathsf{Encrypt}_S(\mathsf{ID}, m_0, m_1)$: Operates as in Fig. 2.
- $\mathsf{Decrypt}_S(\mathsf{ID}_i, c_i, \{1, \ldots, n\})$: Operates as in Fig. 2.
- $\mathsf{Corrupt}_S(\Pi_j)$: This query returns $\mathsf{masterkey}_j$. As an adversary $\mathcal{A}$ already has access to params_j, corrupting Π_j allows $\mathcal{A}$ to extract decryption keys $dkey_j$ for any ID_i. This query models $\mathcal{A}$'s ability to request the collusion of the PKG operating Π_j.

Note that we prove Theorem 1 for a static adversary which, although it may adaptively corrupt nodes at will, must commit to at least one honest PKG before the protocol run. Notably, this follows similarly to the chain of past IBE work, where a static adversarial model is common [23].

Theorem 1. *If $IBE.\mathcal{E}$ protocols Π_j, for $j \in \{0, \ldots, n\}$, are combined to form $CARIBE.\mathcal{E} = C(\Pi_1, \ldots, \Pi_n)$, the resulting $CARIBE.\mathcal{E}$ protocol will be IND-ID-C.CCA provided that there exists $\Pi_t \in \{\Pi_j\}$ that is IND-ID-C.CCA secure.*

Specifically, for any efficient adversary $\mathcal{A}$ that runs in time $\mathtt{t}$ and asks $q = q_{ext} + q_{enc} + q_{dec}$ queries, where q_{ext} are extraction queries, q_{enc} are encryption queries, and q_{dec} are decryption queries, there exists adversaries $\mathcal{B}_j$ that run in time $\mathtt{t}_{\mathcal{B}} \approx \mathtt{t}$ and asks $q_{\mathcal{B}}$ queries, such that

$$\mathsf{Adv}^{\text{IND-ID-C.CCA}}_{C(\Pi_1,\ldots,\Pi_n)}(\mathcal{A}) \leq \mathsf{Adv}^{\text{IND-ID-C.CCA}}_{\Pi_t}(\mathcal{B}_t) \ .$$

Proof. Let $\mathcal{A}$ be an challenger against the IND-ID-CC.CA security of $CARIBE.\mathcal{E} = C(\Pi_1, \ldots, \Pi_n)$ and let $\mathcal{B}$ be an adversary against the IND-ID-C.CCA of $IBE.\mathcal{E}$ for Π_t, with $\mathsf{Extract}_t$, $\mathsf{Encrypt}_t$, and $\mathsf{Decrypt}_t$ oracles running on params_t and $\mathsf{masterkey}_t$, corresponding to Π_t, as well as Extract_j, Encrypt_j, and Decrypt_j algorithms running on params_j and $\mathsf{masterkey}_j$ for $j \in \{1, \ldots, n\}/\{t\}$ which he uses to answer $\mathcal{A}$'s queries. Let $S_{\mathcal{B}}$ be a list of pairs (ID, c) which $\mathcal{B}$ sends back to $\mathcal{A}$ in response to $\mathsf{Encrypt}_S$ queries, maintained by $\mathcal{B}$ and initialized to $\perp$. If at any time $\mathcal{A}$ makes a $\mathsf{Corrupt}(\Pi_t)$ query, $\mathcal{B}$ gives up.

When $\mathcal{A}$ asks an $\mathsf{Extract}_S$ query on (ID_i, c_i), $\mathcal{B}$ calls his $\mathsf{Extract}_t$ oracle and Extract_j algorithms, and sends the agglomerated responses $\{dkey_j\}$ to $\mathcal{A}$.

When $\mathcal{A}$ asks a $\mathsf{Decrypt}_S$ query on a ciphertext c, $\mathcal{B}$ sequentially uses his Decrypt_j algorithms – starting with Decrypt_n on c, followed with Decrypt_{n-1} on the output of Decrypt_n, and so forth for $t < j$. $\mathcal{B}$ then call $\mathsf{Decrypt}_t$ on the output of Decrypt_{t+1}. Thereafter, $\mathcal{B}$ continues with his Decrypt_j calculations, starting with Decrypt_{t-1} on the output of $\mathsf{Decrypt}_t$, for $j < t$. Finally, the result of Decrypt_1 is returned to $\mathcal{A}$.

Should $\mathcal{A}$ query $\mathsf{Encrypt}_S$ on (ID, m_0, m_1), $\mathcal{B}$ behaves as follows: $\mathcal{B}$ calculates $\text{Encrypt}_1(\mathsf{params}_1, \mathsf{ID}, m_0) = c_1^{(0)}$ and $\text{Encrypt}_1(\mathsf{params}_1, \mathsf{ID}, m_1) = c_1^{(1)}$, then $\text{Encrypt}_j(\mathsf{params}_j, \mathsf{ID}, c_{j-1}^{(0)}) = c_j^{(0)}$ and $\text{Encrypt}_j(\mathsf{params}_j, \mathsf{ID}, c_{j-1}^{(1)}) = c_j^{(1)}$ for $1 < j < t$. Then, $\mathcal{B}$ calls $\mathsf{Encrypt}_t(\mathsf{ID}, c_{t-1}^{(0)}, c_{t-1}^{(1)})$ to get c_t. Thereafter $\mathcal{B}$ continues calculating $\text{Encrypt}_j(\mathsf{params}_j, \mathsf{ID}, c_{j-1}) = c_i$ for $t < j \leq n$. Finally, $\mathcal{B}$ sets $c := c_n$, $S \leftarrow S \cup (\mathsf{ID}, c)$, and passes c back to $\mathcal{A}$.

By the IND-ID-C.CCA success of $\mathcal{A}$ against $CARIBE.\mathcal{E}$, at some point $\mathcal{A}$ returns a correct bit guess b'. Hence, $\mathcal{B}$ also wins the IND-ID-C.CCA game against the $IBE.\mathcal{E}$ protocol Π_t with b'. □

From Theorem 1 it can be observed that the CARIBE scheme is at least as secure as the strongest IBE protocol in the cascade. As mentioned before, should multiple PKG collude to determine the decryption key or plaintext, this implies that a CARIBE scheme is secure as long as at most $n - 1$ of the n PKGs collude. Briefly, Table 2 contrasts CARIBE with some of the other IBE-composite schemes discussed in §2.2, under standard structure (special-case HIBE or Distributed IBE schemes may aberrate from the standard in handling of key escrow). Of particular note is the fact that CARIBE allows for composition across multiple IBE platforms – namely, each PKG can use a different IBE scheme for extraction, encryption, etc. In the interest of security, this a logical real-world benefits in the case where preferred PKGs, known to be adverse to mutual collusion, insist on utilizing different IBE schemes.

Topic	Hierarchical IBE	Distributed IBE	CARIBE
Key management authority	n PKGs, hierarchy	n PKGs, t-threshold	n PKGs
Key Escrow	Yes	Limited $t+1$ collusions	Limited n collusions
Certificates	No	No	No
Ciphertext Expansion	Yes	Yes	Yes
Management authority has access to private keys	Yes for all PKGs	Under $t+1 \leq n$ n collusions	Under n collusions
Compromise of management authority is fatal	Yes for any PKG in hierarchy	Under $t+1 \leq n$ collusions	Under n collusions
Incorporation of various encryption methods across PKGs possible	No	No	Yes

Table 2: Comparison of properties among composition IBE schemes.

4.2 Ciphertext Expansion in CARIBE

One natural consequence of cascaded encryption is the amplification of ciphertext expansion. Historically, ciphertext expansion would be a formidable concern under slow transmission rates, and since it is already common in IBE schemes, further expansion from cascades would hardly have been welcomed. However, with increasing improvements in IBE scheme developments involving less expansion, as well as faster transmission rates than used to be available, this issue is not as imposing as it once was. Moreover, the context of modern internet privacy raises the possibility that users may be willing to trade some transmission time to obtain security with convenience.

Since CARIBE is a composition of ciphers of the sender's choice, it is not possible to predict the relative ciphertext expansion in advance without knowing which ciphers, and in what order, the sender will select. Still, some basic observations can be noted. Unquestionably, the expansion involved in a CARIBE of several compositions of Cocks' IBE schemes [9] would be enormous, as ciphertext length in under single encryption is already several times larger than the plaintext length. Yet the ciphertext expansion for a regular, single Cocks scheme is daunting enough at the outset to be naturally prohibitive in practice. Meanwhile, in a CARIBE of n Boneh–Franklin [10] ciphers, a plaintext length $|m|$ would expand to $n \cdot |P| + |m|$, where $|P|$ is the length of a pre-selected element of a group $\mathbb{G}$, of large prime order q, which is part of a bilinear map. Markedly, such a linear expansion is hardly imposing. Aside from classic IBE ciphers, ciphertext expansion in modern IBE proposals lie spread on the scale between these examples, yet far closer to Boneh–Franklin efficiency than Cocks. With simultaneously increasing efficiency and security awareness, it is reasonable that expansion for CARIBE will be of limited concern, particularly as it is based upon the selection of IBE components made and is therefore under the purview of the sender.

5 Conclusion

Combining work on IBE and cascade encryption, CARIBE synthesizes diverse ideas to provide more benefits and a higher level of security than have been achieved in other schemes. Particularly, it provides the key escrow and lack of pre-computation benefits of IBE with the PKI restriction that any given key authority cannot read the encrypted information. One interesting open question relating to this for investigation is relative changes to security and key escrow in the instance that a end-user selects itself as a PKG. However, the security implications of such a scenario are not unique to CARIBE or even identity-based cryptography, but are interesting in the context and comparison of all key management schemes.

Even though there is an inherent time-cost in multiple encryptions for CARIBE, the added security coupled with ever-increasing computing power dilutes this drawback. Additionally, while the onus is on the receiver to obtain decryption keys from multiple authorities, the sender's ability to virtually select a security level for encryption through the PKGs of their choice – possibly based on the encryption type provided by the PKG – may commonly be seen as a sufficient time/security trade-off. Compared with other, contemporary multi-PKG IBE variations, this end-user security selection power provides the catalyst for re-visiting cascade encryption in a previous un-investigated context. In terms of the modern world, where desire to legally access encrypted messages is paired with hostility among internet powers, the consolidation of key escrow and leveraged distrust makes CARIBE particularly appetizing.

References

1. C. Adams, S. Farrell, T. Kause, and T. Mononen. Internet X. 509 Public Key Infrastructure Certificate Management Protocol (CMP). *Request for Comments (RFC)*, 4210, 2005.
2. Sattam S Al-Riyami and Kenneth G Paterson. Certificateless public key cryptography. In *Advances in Cryptology-ASIACRYPT 2003*, pages 452–473. Springer, 2003.
3. M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among Notions of Security for Public-Key Encryption Schemes. In *In Proceedings of CRYPTO'98, Lecture Notes in Computer Science, vol. 1462*, pages 26–45. Springer–Berlin–Heidelberg, 1998.
4. M. Bellare and P. Rogaway. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. In *Advances in Cryptology - EUROCRYPT 2006, Lecture Notes in Computer Science vol. 4004*, pages 409–426. Springer Berlin Heidelberg, 2006.
5. D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003.
6. X. Boyen. A Tapestry of Identity-Based Encryption: Practical Frameworks Compared. *International Journal of Applied Cryptography, vol.1, number 1*, pages 3–21.
7. X. Boyen and B. Waters. Anonymous Hierarchical Identity-Based Encryption (without Random Oracles). In *Advances in Cryptology-CRYPTO 2006*, pages 290–307. Springer, 2006.
8. S. Chatterjee and P. Sarkar. *Identity-Based Encryption*. Springer Science & Business Media, 2011.
9. C. Cocks. An Identity Based Encryption Scheme Based on Quadratic Residues. In *Cryptography and Coding*, pages 360–363. Springer, 2001.
10. D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. In *SIAM Journal of Computing, vol. 32, issue 3*, pages 586–615. Society for Industrial and Applied Mathematics, 2003.
11. Y. Dodis and J. Kratz. Chosen-Ciphertext Security of Multiple Encryption. In *In Proceedings of TCC 2005, Lecture Notes in Computer Science, vol. 3378*, pages 188–209. Springer–Berlin–Heidelberg, 2005.
12. S. Even and O. Goldreich. On the Power of Cascade Ciphers. In *In Proceedings of CRYPTO'83, Advances in Cryptology*, pages 43–50. Springer–Verlag US, 1984.
13. P. Gaži and U. Maurer. Cascade Encryption Revisited. In *Advances in Cryptology – Proceedings of ASIACRYPT 2009, Lecture Notes in Computer Science, vol. 5912*, pages 37–51. Springer–Verlag–Berlin–Heidelberg, 2009.
14. C. Gentry. Certificate-Based Encryption and the Certificate Revocation Problem. In *Advances in Cryptology—EUROCRYPT 2003*, pages 272–293. Springer, 2003.
15. C. Gentry and A. Silverberg. Hierarchical ID-Based Cryptography. In *Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, ASIACRYPT '02, pages 548–566. Springer-Verlag, 2002.
16. G. Greenwald. XKeyscore: NSA tool collects 'nearly everything a user does on the internet'. http://www.theguardian.com/world/2013/jul/31/nsa-top-secret-program-online-data, July 31, 2013. Accessed 2 June, 2015.
17. G. Greenwald and E. MacAskill. NSA Prism program taps in to user data of Apple, Google and others. http://www.theguardian.com/world/2013/jun/06/us-tech-giants-nsa-data, June 7, 2013. Accessed 2 June, 2015.
18. J. Horwitz and B. Lynn. Toward Hierarchical Identity-Based Encryption. In *Advances in Cryptology—EUROCRYPT 2002*, pages 466–481. Springer, 2002.
19. W. Hurd and T.W. Lieu. Congressman Lieu Letter to FBI Director Comey on Encryption "Backdoor" Proposal. https://lieu.house.gov/media-center/, June 1, 2015. Accessed 2 June, 2015.
20. IACR. IACR Statement On Mass Surveillance: Copenhagen Resolution. http://www.iacr.org/misc/statement-May2014.html, May 14, 2014. Accessed 2 June, 2015.
21. A. Joux. Introduction to Identity-Based Cryptography. *Identity- Based Cryptography*, 2009.
22. M. Joye and G. Neven. *Identity-Based Cryptography*, volume 2. IOS press, 2009.
23. A. Kate and I. Goldberg. Distributed private-key generators for identity-based cryptography. In *Security and Cryptography for Networks*, volume 6280 of *Lecture Notes in Computer Science*, pages 436–453. Springer Berlin Heidelberg, 2010.
24. J. Katz and Y. Lindell. *Introduction to Modern Cryptography*. CRC Press, 2014.
25. N. Leavitt. Internet Security under Attack: The Undermining of Digital Certificates. *Computer*, 44(12):17–20, Dec 2011.
26. W. Mao. *Modern Cryptography: Theory and Practice*. Prentice Hall PTR, 2004.
27. M. Maurer and J. Massey. Cascade Ciphers: The Importance of Being First. In *Journal of Cryptology, vol. 6, Issue 1*, pages 55–61. Springer–Verlag, 1993.
28. National Institute of Standards and Technology. http://www.nist.gov/, Accessed 2 June, 2015.
29. K. G. Paterson and S. Srinivasan. Security and Anonymity of Identity-Based Encryption with Multiple Trusted Authorities. In *In Proceedings of Pairing-Based Cryptography–Pairing 2008, Lecture Notes in Computer Science, vol. 5209*, pages 354–375. Springer–Berlin–Heidelberg, 2008.
30. V. Popov, I. Kurepkin, and S. Leontiev. RFC 4357: Additional Cryptographic Algorithms for Use with GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94 Algorithms. Technical report, January 2006.

31. R. Sakai, K. Ohgishi, and M Kasahara. Cryptosystems Based on Pairing. In *The 2000 Symposium on Cryptography and Information Security, Okinawa, Japan*, pages 135–148, 2000.
32. A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In *Advances in cryptology*, pages 47–53. Springer, 1985.
33. S. Tessaro. Security Amplification for the Cascade of Arbitrarily Weak PRPs: Tight Bounds via the Interactive Hardcore Lemma. In *In Proceedings of Theory of Cryptography TCC 2011, Lecture Notes in Computer Science vol. 6597*, pages 37–54. Springer Berlin Heidelberg, 2011.
34. A. Whitten and J.D. Tygar. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In *Usenix Security*, volume 1999, 1999.
35. D. Yao, N. Fazio, Y. Dodis, and A. Lysyanskaya. ID-Based Encryption for Complex Hierarchies with Applications to forward Security and Broadcast Encryption. In *Proceedings of the 11th ACM conference on Computer and communications security*, pages 354–363. ACM, 2004.