# Bounds on Ad Hoc Threshold Encryption*

Ivan Damgård, Sophia Yakoubov

Aarhus University, {ivan, sophia.yakoubov}@cs.au.dk

**Abstract.** Threshold encryption is encryption to a group of $n$ intended recipients in such a way that any $t+1$ out of the $n$ recipients together can decrypt, but $t$ or fewer learn nothing about the message. *Ad hoc* threshold encryption (ATE) is threshold encryption with no trusted setup beyond a PKI (that is, all keys are generated independently). The techniques known in the ad hoc setting suffer either from ciphertexts linear in $(n-t)$ (Daza *et. al*), or from reliance on cumbersome primitives like indistinguishability obfuscation (Reyzin *et. al*).

In this paper, we set out to determine whether we can get ATE with short ciphertexts from standard primitives. We therefore work in a model where we limit reliance on computational assumptions. We do this by demanding information theoretic security given black-box access to limited cryptographic tools such as non-interactive key exchange and pseudorandom generators.

We show that, with access only to idealized two-party key exchange, any secure ATE scheme must produce ciphertexts of size at least $(n-t-1)l$ (where $l$ is the length of the message). If access is additionally given to an idealized PRG, the lower bound on ciphertext size becomes $\lceil \frac{n-t}{2} \rceil \lambda + l$ (where $\lambda$ is the length of the input to the PRG).

If idealized $q$-party key exchange for $q > 2$ is availabe, then we can achieve a constant-size ciphertext, at the cost of invoking the key exchange an exponential number of times. We also prove that, if the size of the ciphertext is optimal (that is, equal to the size of the message), the exponential overhead is unavoidable. Finally, we give some alternative constructions demonstrating that the overhead can be reduced at the cost of slightly larger ciphertext size.

* This research was supported by: the European Research Council (ERC) under the European Unions's Horizon 2020 research and innovation programme under grant agreement No 669255 (MPCPRO); the NSF MACS project.

# Table of Contents



## 1 Introduction

The concept of public-key threshold encryption is very well known. It goes back at least to Desmedt *et. al*[DDFY94], and has since then been studied in a very long line of research. For this type of scheme, the key generation outputs a public key $\mathsf{pk}$ and a set of secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_n$ which are generated with respect to a threshold value $t$, where $0 \leq t < n$. Informally, the important security properties are that given any set of at least $t + 1$ secret keys, one can decrypt a ciphertext encrypted under $\mathsf{pk}$, while the encryption remains secure even given any set of $t$ secret keys. For efficiency, ciphertexts should have size independent of $n$.

Requiring a single trusted execution of key generation can be very limiting, particularly in a system where parties may join at any point, or where senders want to dynamically choose subsets of the parties to be the recipients of

a particular message. *Dynamic* threshold public-key encryption, introduced by Delerablée and Pointcheval [DP08], has a reduced setup requirement where the sender can pick the set of $n$ recipients at encryption time; however, each recipient's secret key must be derived from a common master secret key, so a trusted authority is still necessary. *Ad hoc threshold encryption* (ATE), first introduced by Daza *et. al*[DHMR08] as *threshold broadcast encryption*[1] (motivated by its applicability to mobile ad hoc networks), requires no trusted setup beyond the absolute minimum — a PKI.

ATE considers a universe of users, where each user $i$ has a public key $\mathsf{pk}_i$ and corresponding secret key $\mathsf{sk}_i$, and where all key pairs are independently generated. A sender can select a set $\mathcal{R}$ of $n$ users and a threshold value $t$ at the time at which he decides to send a message $\mathsf{m}$. He can then construct a ciphertext $\mathsf{C} = E_{\mathsf{pk}_\mathcal{R},t}(\mathsf{m})$, where $\mathsf{pk}_\mathcal{R}$ is the set of public keys belonging to parties in $\mathcal{R}$. ATE requires properties similar to those of standard threshold encryption: namely, that any $t+1$ parties in $\mathcal{R}$ can decrypt, while the encryption remains semantically secure even given the secret keys of any $t$ parties in $\mathcal{R}$.

Clearly, ATE has a number of attractive properties that standard threshold encryption lacks: no trusted authority, and the ability to decide on the set of receivers and the threshold on the fly. On the other hand, it is not clear that an ATE ciphertext can be as small as a standard one. The easiest approach to building an ATE solution is to secret-share the message with threshold $t$ to get shares $\mathsf{m}_1, \ldots, \mathsf{m}_n$ and encrypt the shares, setting $E_{\{\mathsf{pk}_1,\ldots,\mathsf{pk}_n\},t}(\mathsf{m}) = (E_{\mathsf{pk}_1}(\mathsf{m}_1), \ldots, E_{\mathsf{pk}_n}(\mathsf{m}_n))$. This clearly works, but leads to ciphertext size linear in $n$. Daza *et. al*[DHMR08] show how to decrease the ciphertext size to be linear in $n-t$, which, while better, may still be unmanageable for large $n$ and small $t$. Reyzin *et. al*[RSY18] show that using indistinguishability obfuscation, as well as few standard primitives, it is possible to get ciphertext size independent of $n$. There are several reasons, however, why this is not a very satisfactory answer. For one thing, the construction requires senders to have public and secret keys, which is not usually assumed for ATE. Moreover, obfuscation requires strong assumptions; with current state of the art techniques, it comes at the price of a huge loss of efficiency in practice; and finally, it is well known that candidate obfuscation constructions have a somewhat shaky security status.

An important question therefore is to determine whether we can get ATE with short ciphertexts using standard cryptographic techniques. This question is completely open, except for the case of almost maximal threshold $t$. When $t = n-1$ (that is, where everyone must collaborate in decryption), it is easy to see that we can get constant size ciphertexts based on the DDH assumption. One can simply multiply together $n$ parties' El Gamal public keys, to get a new key that is also a valid El Gamal public key; the corresponding secret key is now effectively additively secret shared among the $n$ original parties. One can then

[1] One should note that ATE for $t = 0$ is very similar to broadcast encryption: each party can decrypt on his own. However, in broadcast encryption, centralized key generation is usually allowed (or at least key generation is coordinated between receivers). This is exactly what is not allowed in ATE.

produce the ATE ciphertext by encrypting using El Gamal under the public key product; this ciphertext will have size independent of $n$. For $t = n - c$ (for constant $c$), one can use the techniques of Daza *et. al*[DHMR08] to get constant-size ciphertexts (since $n-t = c$), also under DDH or other standard assumptions. However, there is no known way to generalize any of this to significantly smaller thresholds, which makes the problem all the more tantalizing.

### 1.1 Our Contribution

In this paper, our goal is to understand the feasibility of ATE with short ciphertexts from standard cryptographic techniques. Clearly, indistinguishability obfuscation (which enables constant-size ciphertexts [RSY18]) is not standard; an important question is finding a suitable restriction that precludes the use of indistinguishability obfuscation, while allowing the use of more standard tools.

We take the approach of giving protocols access to ideal functionalities for certain basic cryptographic primitives such as key exchange and pseudorandom generators. We then require that the ATE constructions that use those primitives be information theoretically secure assuming the idealized versions of the primitives. Intuitively, this ensures that the protocols we consider can use the underlying primitives, but cannot use anything else that requires computational assumptions, since such assumptions cannot lead to information theoretic security.

Lower bounds in such a model are, on the one hand, limited — they only hold for restricted black-box use of the underlying primitives. On the other hand, they are stronger than usual, in two ways. First, when proving lower bounds, we do not assume that the idealized primitives or the protocols themselves are efficient[2]. Second, the communication cost of implementing the idealized primitives is not considered as part of the ciphertext size.

Our results in this setting are summarized in Figure 1.

We show, using information theoretic methods, that if the only cryptographic tool you have is key exchange (a functionality that outputs the same random string to two parties), then it is impossible to achieve a constant size ciphertext. In fact, the ciphertext must have size at least $(n-t-1)l$, where $l$ is the entropy of the message (Theorem 2). The construction of Daza *et. al*[DHMR08] (which is in our model) leads to an upper bound of $(n-t)l$ on the ciphertext size.

As mentioned above, we do not charge the protocols for the communication cost of implementing the available primitives, which makes our lower bound stronger. As for the upper bounds, we note that key exchange between a sender and several receivers can be implemented with very little communication overhead based on DDH: the sender provides a group element $g^s$ as part of the

[2] Note that in a model where protocols do not have to be efficient, indistinguishability obfuscation exists: one just takes the lexicographically first circuit with the desired input-output behavior [MR13]. However, the construction from [RSY18] is still excluded in our model because it makes essential non-black-box use of several standard primitives.

| | | | Bounds | | | | |
|---|---|---|---|---|---|---|---|
| | | | Lower Bounds | | Upper Bounds | | |
| Threshold | Assumptions | Security | Ctext Size | Number of NIKE Calls | Ctext Size | Number of NIKE Calls | NIKE Output Size |
| any $t$ | 2-party NIKE | Perfect | $(n-t-1)l$ (Theorem 2) | | Constr 1 $(n-t)l$ | $n$ | $l$ |
| any $t$ | 2-party NIKE | Statistical | $(n-t-1)l$ $- negl(\lambda)$ (Theorem 4) | | Constr 1 $(n-t)l$ | $n$ | $l$ |
| any $t$ | 2-party NIKE and PRG | Statistical | $\lceil \frac{n-t}{2} \rceil \lambda + l -$ $negl(\lambda)$ (Theorem 5) | | Constr 2 $(n-t)\lambda + l$ | $n$ | $\lambda$ |
| any $t$ | $(n-t+1)$-party NIKE | Perfect | | | Constr 3 $l$ | $\binom{n}{n-t}$ | $l$ |
| any $t = \Theta(n)$ | $q$-party NIKE for any $q$ | Perfect | $l$ | exponential (Theorem 7) | | | |
| $t = 0$ (broadcast) | $q$-party NIKE for any $q$ | Perfect | | | Constr 4 $\lceil \frac{n}{q-1} \rceil l$ | $\lceil \frac{n}{q-1} \rceil$ | $l$ |
| $t = 1$ | $q$-party NIKE for $q = \lceil \frac{n}{2} \rceil + 1$ | Perfect | | | Constr 5 $\log(n)l$ | $2\log(n)$ | $l$ |
| any $t$ | $q$-party NIKE for any $q$ | Perfect | | | Constr 5 $2(t+1)^3 \cdot \log(n)l$ | $2(t+1)^3 \cdot \log(n)$ | $l$ |

Fig. 1: Summary of Results. In this table, $n$ denotes the size of the recipient set, $t$ denotes the threshold of parties (such that any subset of recipients of size greater than $t$ is able to decrypt), $l$ denotes the entropy of the message (often equivalent to its length), and $\lambda$ denotes the security parameter. NIKE stands for Non Interactive Key Exchange.

ciphertext, and for receiver $i$ with public key $g^{x_i}$, the shared key is $g^{sx_i}$. Thus, the communication overhead for the key exchange is independent of $n$, so our upper bound can indeed be very close to the lower bound even for a practical implementation.

In the same spirit, we show that if we grant access to both key exchange and a PRG, then the lower bound is $\lceil \frac{n-t}{2} \rceil \lambda + l$, where $\lambda$ is the seed length of the PRG (Theorem 5). This says that the obvious approach where we send a seed using ATE, and use the output of the PRG as a one-time pad on the message, is optimal up to a factor of two in this model.

We next explore what is possible if we give access to a more general form of key exchange – multiparty non-interactive key exchange, or *q-NIKE* – where $q$ (the sender and $q-1$ receivers) can agree on the same randomness non-interactively (from their respective public keys). Multiparty NIKE can be constructed from multilinear maps, indistinguishability obfuscation [BZ14], universal samplers [HJK+16,GPSZ17] (which can be built from indistinguishability obfuscation or functional encryption), or encryption combiners satisfying perfect independence [MZ17] (which can be built from universal samplers).

We show that leveraging multiparty NIKE allows for ATE with optimal ciphertext size, equal to the message length (Constr 3); however, our construction requires an exponential number (in $n$) of invocations of a $(n-t+1)$-NIKE. We also show that if the goal is to have the ciphertext be of the same length as the message (which is optimal), then the hurdle of having to invoke the multiparty NIKE an exponential number of times is unavoidable (Theorem 7).

Finally, we explore other ways to leverage multiparty NIKE. In particular, when $t$ is small enough — that is, when $t^3 \log n$ is $o(n)$ — Constr 5 shows that we can have ciphertext size sublinear in $n$, while invoking the NIKE polynomially many (in $n$) times. This demonstrates a tradeoff that is fundamentally different from our other results: using multiparty NIKE, we can break the lower bound on ciphertext size for 2-NIKE, but still avoid the exponential overhead required for minimal ciphertext size.

### 1.2 Open Problems

This is a very rich space of problems to explore. In particular, it is natural to ask what becomes of lower bounds on the ciphertext size given $q$-NIKE for various values of $q$. For $q = n - t$ we can obtain optimal-size ciphertexts (Constr 3), but for $q = 2$ we have $(n-t-1)l$ as a lower bound (Theorem 2). Lower bounds on ciphertext size for other values of $q$ are wide open.

Another natural question to ask is, for general $q$, what becomes of the tradeoff between number of invocations of $q$-NIKE and ciphertext size. Is there a lower bound on the number of $q$-NIKE invocations for constant-size (but sub-optimal) ciphertexts, like there is for optimal ciphertexts (Theorem 7)?

### 1.3 Notation

We use lowercase variables — $\mathsf{pk}, \mathsf{sk}, \mathsf{m}, \mathsf{c}$ — to refer to concrete values, and uppercase variables — $\mathsf{PK}, \mathsf{SK}, \mathsf{M}, \mathsf{C}$ — to refer to distributions.

## 2 Definitions

In this section, we give some rudimentary definitions of ad hoc homomorphic encryption (ATE).

### 2.1 ATE Syntax

An ATE scheme consists of three algorithms, described below.

$KG(1^\lambda) \rightarrow (\mathsf{pk}, \mathsf{sk})$ is a randomized key generation algorithm that takes in the security parameter $\lambda$ and returns a public-private key pair.

$E_{\mathsf{pk}_\mathcal{R}, t}(\mathsf{m}) \rightarrow \mathsf{c}$ is an encryption algorithm that encrypts a message $\mathsf{m}$ to a set of public keys $\mathsf{pk}_\mathcal{R} = \{\mathsf{pk}_i\}_{i \in \mathcal{R}}$ belonging to the parties in the intended recipient set $\mathcal{R}$ in such a way that any size-$(t+1)$ subset of the recipient set should jointly be able to decrypt.

$D_{\mathsf{pk}_\mathcal{R}, \mathsf{sk}_\mathcal{A}}(\mathsf{c}) \rightarrow \mathsf{m}$ is a decryption algorithm that uses secret keys $\mathsf{sk}_\mathcal{A} = \{\mathsf{sk}_i\}_{i \in \mathcal{A}}$ belonging to a subset $\mathcal{A}$ of the intended recipient set $\mathcal{R}$ (where $|\mathcal{A}| > t$) to decrypt the ciphertext $\mathsf{c}$ and recover the message $\mathsf{m}$.

In this paper, we want to model giving the parties black-box access to an idealized cryptographic primitive $F$. All three algorithms are then augmented with oracle access to $F$.

When $F$ is a primitive like key exchange that has no inputs (apart from party identities) — only outputs — we can get away with a simplification. Because there are no inputs, it doesn't really matter *when* $F$ is called (since inputs cannot be adaptively chosen), so we simply assume that $F$ has been called a sufficient number of times before encryption (say, at key generation time). (Of course, actually implementing ATE this way would have serious efficiency issues; key exchange would have to be run among all sets of parties many times, even among those who might not have a future need to send messages to one another. However, even though ATE should not be implemented this way, modeling it this way is valid for arguing lower bounds on ciphertext size.)

In the rest of this paper, we abstract away from using public-private key pairs which we assume are only used to implement key exchange[3]. Instead, we directly model the outputs of key exchange as a set $\mathsf{s} = \{\mathsf{s}_i\}$ of shared keys. In the idealized key exchange model, $\mathsf{s}$ is a random value whose components are shared by multiple parties (usually by the sender and one of the receivers). Under this simplification, the syntax changes as follows. $KG(1^\lambda)$ is replaced by

[3] This follows from the fact that our first goal is to analyze information theoretically secure ATE based on idealized key exchange.

invocations of the idealized key exchange functionality, distributing shared keys in $\mathsf{s}$ to the appropriate parties Namely, $\mathsf{s}_i$ to party $i$ and the entire set $\mathsf{s}$ to the sender. Encryption becomes $E_{\mathsf{s},t}(\mathsf{m}) \to \mathsf{c}$, and decryption becomes $D_{\mathsf{s}_\mathcal{A}}(\mathsf{c}) \to \mathsf{m}$ (where $\mathsf{s}_\mathcal{A} = \{\mathsf{s}_i\}_{i\in\mathcal{A}}$, $|\mathcal{A}| > t$).

### 2.2 ATE Security in the Pairwise Key Exchange Model

For our purposes, an ad hoc threshold encryption (ATE) scheme in the pairwise key exchange model is characterized by the joint distribution of the following random variables: $\mathsf{S}_1, \ldots, \mathsf{S}_n$ (where $\mathsf{S}_i$ is a random variable shared by the sender and receiver $i$), the message $\mathsf{M}$, and the ciphertext $\mathsf{C}$. Informally, an ATE scheme is secure if any $t$ parties in the designated set of receivers $\mathcal{R}$ can learn nothing about a message from a ciphertext, but any $t+1$ parties in $\mathcal{R}$ can recover the message. More precisely:

**Definition 1 (ATE Perfect Security).** *An ATE scheme $(E, D)$ in the pairwise key exchange model is* perfectly secure with threshold $t$ *if for any set of receivers $\mathcal{R}$ of size $n$, for $\mathsf{C} = E_{\mathsf{S},t}(\mathsf{M})$, the following two properties hold:*

**Security** *For any $\mathcal{A} \subset \mathcal{R}$ of size at most $t$, we have $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_\mathcal{A}) = H(\mathsf{M})$, where $\mathsf{S}_\mathcal{A} = \{\mathsf{S}_i\}_{i\in\mathcal{A}}$ is the set of keys shared by the sender and parties in $\mathcal{A}$.*

**Correctness** *For any $\mathcal{A} \subset \mathcal{R}$ of size greater than $t$, we have $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_\mathcal{A}) = 0$. Furthermore, $\mathsf{M} = D_{\mathsf{S}_\mathcal{R}}(\mathsf{C})$.*

We can define statistical security similarly, where we assume that the distribution of the variables may also depend on a security parameter $\lambda$, but we always assume that the parameters $l, n, t$ are polynomial in $\lambda$.

**Definition 2 (ATE Statistical Security).** *An ATE scheme $(E, D)$ in the pairwise key exchange model is* statistically secure with threshold $t$ *if for any set of receivers $\mathcal{R}$ of size $n$, for $\mathsf{C} = E_{\mathsf{S},t}(\mathsf{M})$, the following two properties hold:*

**Security** *If for any $\mathcal{A} \subset \mathcal{R}$ of size at most $t$, we have $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_\mathcal{A}) \geq H(\mathsf{M}) - negl(\lambda)$, where $\mathsf{S}_\mathcal{A} = \{\mathsf{S}_i\}_{i\in\mathcal{A}}$ is the set of keys shared by the sender and parties in $\mathcal{A}$.*

**Correctness** *For any $\mathcal{A} \subset \mathcal{R}$ of size greater than $t$, we have $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_\mathcal{A}) \leq negl(\lambda)$. Furthermore, $\mathsf{M} = D_{\mathsf{S}_\mathcal{R}}(\mathsf{C})$ with overwhelming probability.*

### 2.3 Relationship to Other Definitions

In this section, we consider the relationship between our definitions and some other flavors of threshold encryption definitions.

*Relationship to Semantic Security* For those readers used to dealing with computational notions of security, Definition 1 might not look quite like a typical definition for encryption. In the security requirement, instead of demanding the inability of an adversary to distinguish between encryptions of two messages (as in semantic security), we demand that the entropy of the message not diminish in the presence of the ciphertext and insufficiently many keys. However, the entropy of the message remaining the same in the presence of the ciphertext clearly implies that even an unbounded adversary cannot distinguish between encryptions of two messages.

*Relationship to Adaptive Security* Because we require security against *any* subset of corrupt parties $\mathcal{A} \subset \mathcal{R}$ of size at most $t$, we get an equivalent of adaptive security (where the adversary can choose the subset adaptively based on the challenge ciphertext), which is stronger than selective security (where the adversary must choose the subset before seeing the ciphertext).

*Modeling Encryption Oracles* Definition 1 does not explicitly model additional information that may be available to an adversary, such as known message / ciphertext pairs, or access to encryption oracles. We do not model this information because, in our idealized key exchange setting, it cannot be of any possible use. We assume that every message is encrypted with fresh outputs $\mathsf{S}$ of the idealized key exchange, so an adversary can simulate known message / ciphertext pairs and encryption oracles by picking the values $\mathsf{S}$ for other messages and ciphertexts at random. We do not consider decryption oracles or CCA security in this paper.

## 3 Bounds Assuming Idealized Key Exchance

In this section, we assume that a sender has access to pairwise shared randomness with each of the $n$ receivers (distributed, e.g., by an idealized key exchange functionality). Thus each receiver $i$ gets a random variable (or *key*) $\mathsf{S}_i$, while the sender has access to the full set of keys $\{\mathsf{S}_1, \ldots, \mathsf{S}_n\}$. The keys $\mathsf{S}_i$ are independent, since they are the result of separate calls to the key exchange functionality. The choice of message is represented by random variable $\mathsf{M}$, and we let $l$ denote the entropy of $\mathsf{M}$, which we assume (for simplicity) is fixed. The sender computes a ciphertext $\mathsf{C}$ from $\mathsf{M}$, $\{\mathsf{S}_1, \ldots, \mathsf{S}_n\}$ and possibly some local randomness.

### 3.1 Lower Bound on Ciphertext Size in the Perfect Security Setting

We begin with Theorem 1, which gives a lower bound on ciphertext size specifically for the broadcast ($t = 0$) setting. We then use Theorem 1 to prove Theorem 2, which deals with general $t$.

**Theorem 1.** *For any ATE scheme in the pairwise key exchange model that is perfectly secure with threshold $t = 0$, it holds that*

$$H(\mathsf{C}) \geq (n-1)l.$$

*Proof.* In this proof, we refer to the entropy diagram in Figure 2.

1. $H(\mathsf{M}|\mathsf{C}) = H(\mathsf{M})$, by perfect security. It follows that $b + c = 0$ in Figure 2 (that is, $\mathsf{M}$ and $\mathsf{C}$ are independent). (Note also that $g = 0$, since $\mathsf{C}$ together with $\mathsf{S}_i$ completely determine $\mathsf{M}$.)
2. For $i \in \{1, \ldots, n\}$, $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_i) = 0$, by the fact that any one key together with the ciphertext can be used to recover the message. It follows that $H(\mathsf{M}) = a + b + c = a$ (by Step 1, since $b + c = 0$).
3. $H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) = H(\mathsf{S}_i) - (a + b + d) \leq H(\mathsf{S}_i) - a$ (since $b + d$ must be non-negative) $= H(\mathsf{S}_i) - H(\mathsf{M})$ (by Step 2, since $H(\mathsf{M}) = a$).
4. 

$$H(\mathsf{S}_1, \ldots, \mathsf{S}_n|\mathsf{M}, \mathsf{C}) \leq \sum_{i=1}^{n} H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) \tag{1}$$

$$\leq \sum_{i=1}^{n} \big(H(\mathsf{S}_i) - H(\mathsf{M})\big) \qquad \text{(by Step 3)} \tag{2}$$

$$= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nH(\mathsf{M}) \tag{3}$$

5. On the other hand, we have

$$H(\mathsf{S}_1, \ldots, \mathsf{S}_n|\mathsf{M}, \mathsf{C}) \geq H(\mathsf{S}_1, \ldots, \mathsf{S}_n) - H(\mathsf{C}, \mathsf{M}) \tag{4}$$

$$= H(\mathsf{S}_1, \ldots, \mathsf{S}_n) - H(\mathsf{C}) - H(\mathsf{M}) \tag{5}$$

$$= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - H(\mathsf{C}) - H(\mathsf{M}), \tag{6}$$

   since the keys are all independent, and $\mathsf{C}$ and $\mathsf{M}$ are independent (Step 1).
6. By combining Step 4 and Step 5, we get

$$\Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - H(\mathsf{C}) - H(\mathsf{M}) \leq \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nH(\mathsf{M})$$

   from which we conclude that

$$H(\mathsf{C}) \geq (n-1)H(\mathsf{M}) = (n-1)l.$$

■

**Theorem 2.** *For any ATE scheme in the pairwise key exchange model that is perfectly secure with threshold $t$, it holds that*

$$H(\mathsf{C}) \geq (n - t - 1)l.$$

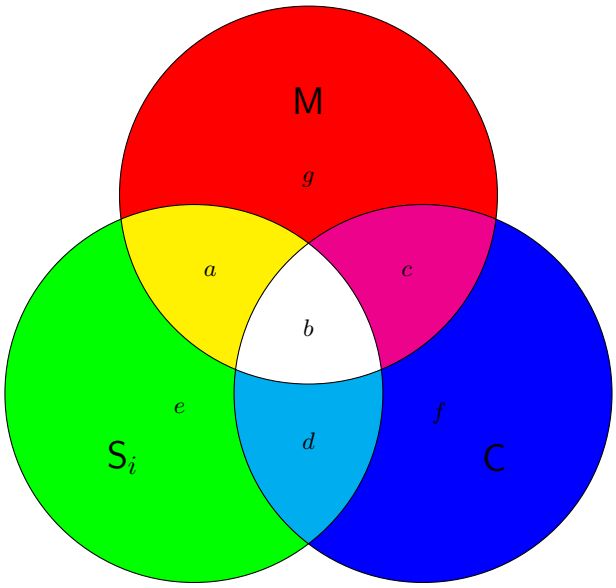


Fig. 2: Entropy Diagram. Each circle represents the entropy of an object $(\mathsf{M}, \mathsf{C}, \mathsf{S}_i)$. The overlap areas represent the mutual information of two objects. For instance $I(C; M) = b + c$. All subareas represent amounts of information and are therefore non-negative, except $b$ which may be negative.

*Proof.* Fix arbitrary values $\mathsf{s}_1, \ldots, \mathsf{s}_t$. Let $E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}$ be the event that $\mathsf{S}_1 = \mathsf{s}_1, \ldots, \mathsf{S}_t = \mathsf{s}_t$. Clearly, it is sufficient to show that for any such event, $H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \geq (n-t-1)l$, since this implies

$$H(\mathsf{C}) \geq H(\mathsf{C}|\mathsf{S}_1, \ldots, \mathsf{S}_t) = \sum_{\mathsf{s}_1,\ldots,\mathsf{s}_t} H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \Pr[E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}] \geq (n-t-1)l.$$

We do this in the following way. We fix keys $\mathsf{s}_1, \ldots, \mathsf{s}_t$ belonging to notional parties as public parameters. It is then straightforward to see that we can build an ATE with $t = 0$ by always including the notional parties $1, \ldots, t$ in the recipient set $\mathcal{R}$; any of the remaining $n - t$ parties will be able to decrypt on their own. Thus, we can leverage Theorem 1 (with a modified recipient set size $n' = n - t$) to conclude that

$$H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \geq (n'-1)l = (n-t-1)l.$$

■

### 3.2 Lower Bound on Ciphertext Size in the Statistical Security Setting

We have a very similar result for statistical security. Like in the perfect security setting, we begin with Theorem 3, which gives a lower bound on ciphertext size specifically for the broadcast ($t = 0$) setting.

**Theorem 3.** *For any ATE scheme in the pairwise key exchange model that is statistically secure with threshold $t = 0$, it holds that*

$$H(\mathsf{C}) \geq (n-1)H(\mathsf{M}) - \delta(\lambda)$$

*for a negligible function $\delta(\lambda)$.*

*Proof.* Like the proof of Theorem 1, this proof refers to the entropy diagram in Figure 2.

1. By the statistical security of the ATE scheme, $H(\mathsf{M}) - H(\mathsf{M}|\mathsf{C}) = \epsilon(\lambda)$ for a negligible function $\epsilon(\lambda)$. So,

   $$b + c = \epsilon(\lambda)$$

   in Figure 2 (or, in other words, $\mathsf{M}$ and $\mathsf{C}$ are nearly independent; $H(\mathsf{M}, \mathsf{C}) = H(\mathsf{M}) + H(\mathsf{C}) - \epsilon(\lambda)$).
2. For $i \in \{1, \ldots, n\}$, statistical security implies that $g$ (in Figure 2) $= H(\mathsf{M}|\mathsf{C}, \mathsf{S}_i) = \epsilon(\lambda)$ for a negligible function $\epsilon(\lambda)$.

   $$H(\mathsf{M}) = a + b + c + g = a + 2\epsilon(\lambda)$$

   by Step 1, since we can assume $b + c = \epsilon(\lambda)$.

3. For $i \in \{1, \ldots, n\}$, $H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) = H(\mathsf{S}_i) - (a+b+d) \leq H(\mathsf{S}_i) - a$, since $b+d$ must be non-negative. Then by Step 2, we get

$$H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) \leq H(\mathsf{S}_i) - H(\mathsf{M}) + 2\epsilon(\lambda).$$

4. We now use Step 3 to conclude:

$$\begin{aligned} H(\mathsf{S}_1, \ldots, \mathsf{S}_n|\mathsf{M}, \mathsf{C}) &\leq \sum_{i=1}^{n} H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) && (7)\\ &\leq \sum_{i=1}^{n} \big(H(\mathsf{S}_i) - H(\mathsf{M}) + 2\epsilon(\lambda)\big) && (8)\\ &= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nH(\mathsf{M}) + 2n\epsilon(\lambda) && (9) \end{aligned}$$

5. On the other hand, we have

$$\begin{aligned} H(\mathsf{S}_1, \ldots, \mathsf{S}_n|\mathsf{M}, \mathsf{C}) &\geq H(\mathsf{S}_1, \ldots, \mathsf{S}_n) - H(\mathsf{M}, \mathsf{C}) && (10)\\ &= H(\mathsf{S}_1, \ldots, \mathsf{S}_n) - (H(\mathsf{M}) + H(\mathsf{C}) - \epsilon(\lambda)) && (11)\\ &= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - H(\mathsf{M}) - H(\mathsf{C}) + \epsilon(\lambda), && (12) \end{aligned}$$

   by Step 1 and since the keys are all independent.
6. By combining Step 4 and Step 5, we get

$$\Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - H(\mathsf{M}) - H(\mathsf{C}) \leq \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nH(\mathsf{M}) + (2n-1)\epsilon(\lambda)$$

   from which we conclude that

$$H(\mathsf{C}) \geq (n-1)H(\mathsf{M}) - \delta(\lambda),$$

   where we define the negligible function $\delta(\lambda) = (2n-1)\epsilon(\lambda)$.

■

**Theorem 4.** *For any ATE scheme in the pairwise key exchange model that is statistically secure with threshold $t$, it holds that*

$$H(\mathsf{C}) \geq (n-t-1)H(\mathsf{M}) - \delta(\lambda),$$

*for a negligible function $\delta(\lambda)$.*

To prove Theorem 4, we first need the following lemma, which is quite well known, but we include a proof for completeness.

**Lemma 1.** *For any random variable $X$ with $X \geq 0$, we have* $\Pr[X \geq \sqrt{E(X)}] \leq \sqrt{E(X)}$.

*Proof.* Writing $P[X = x] = p_x$, and $E(X) = E$, we have

$$E = \sum_x xp_x = \sum_{x \le \sqrt{E}} xp_x + \sum_{x > \sqrt{E}} xp_x \tag{13}$$

$$\geq \sum_{x \le \sqrt{E}} 0p_x + \sum_{x > \sqrt{E}} \sqrt{E}p_x \tag{14}$$

$$= \sqrt{E} \sum_{x \geq \sqrt{E}} p_x \tag{15}$$

$$= \sqrt{E} \Pr[x \geq \sqrt{E}] \tag{16}$$

from which the lemma follows immediately. ■

*Proof (Theorem 4).* Fix arbitrary values $\mathsf{s}_1, \ldots, \mathsf{s}_t$ and define the event $E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}$ be the event that $\mathsf{S}_1 = \mathsf{s}_1, \ldots, \mathsf{S}_t = \mathsf{s}_t$. It is sufficient to show the following claim.

*Claim.* We can define a set of *good* values $\mathsf{s}_1, \ldots, \mathsf{s}_t$, such that

1. it holds that for good values we have
$$H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \geq (n - t - 1)H(\mathsf{M}) - \eta(\lambda)$$
for a negligible function $\eta(\lambda)$;
2. and the probability that $\mathsf{s}_1, \ldots, \mathsf{s}_t$ is bad is a negligible function $\nu(\lambda)$.

Namely, if the claim holds, we have:

$$H(\mathsf{C}) \geq H(\mathsf{C}|\mathsf{S}_1, \ldots, \mathsf{S}_t) \tag{17}$$

$$= \sum_{\mathsf{s}_1,\ldots,\mathsf{s}_t} H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \Pr[E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}] \tag{18}$$

$$\geq \sum_{good\ \mathsf{s}_1,\ldots,\mathsf{s}_t} H(\mathsf{C}|E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}) \Pr[E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}] \tag{19}$$

$$\geq ((n - t - 1)H(\mathsf{M}) - \eta(\lambda)) \Pr[\mathsf{s}_1, \ldots, \mathsf{s}_t \text{ is good}] \tag{20}$$

$$= ((n - t - 1)H(\mathsf{M}) - \eta(\lambda))(1 - \nu(\lambda)) \tag{21}$$

from which the conclusion of the theorem follows easily.

To show the above claim, we write $\tilde{H}$ for entropies of all variables as they are distributed when we condition on $E_{\mathsf{s}_1,\ldots,\mathsf{s}_t}$. Since the keys are chosen independently of $\mathsf{M}$, we have $H(\mathsf{M}) = \tilde{H}(\mathsf{M})$.

We define $\mathsf{s}_1, \ldots, \mathsf{s}_t$ to be *good* if, when $\mathsf{s}_1, \ldots, \mathsf{s}_t$ are fixed as public parameters, the $t$ of $n$ ATE scheme becomes a statistically secure 0 of $n-t$ ATE scheme. The first part of the claim then follows by Theorem 3.

By the statistical security of the ATE scheme, $H(\mathsf{M}) - H(\mathsf{M}|\mathsf{C}) = \epsilon(\lambda)$ for negligible $\epsilon(\lambda)$. We can think of $\tilde{H}(\mathsf{M}) - \tilde{H}(\mathsf{M}|\mathsf{C})$ as a random variable that takes a value for each fixed choice of $\mathsf{s}_1, \ldots, \mathsf{s}_t$. Its expectation is then clearly $\epsilon(\lambda)$. Further since $\tilde{H}()$ stands for entropy (of distributions conditioned on the

choice of $\mathsf{s}_1, \ldots, \mathsf{s}_t$) we have $\tilde{H}(\mathsf{M}) - \tilde{H}(\mathsf{M}|\mathsf{C}) \geq 0$ by the standard inequality for entropy conditioned on a variable.

Therefore Lemma 1 gives us that $\tilde{H}(\mathsf{M}) - \tilde{H}(\mathsf{M}|\mathsf{C}) \leq \sqrt{\epsilon(\lambda)}$, except with probability $\sqrt{\epsilon(\lambda)}$.

Likewise, statisitical security also says that $H(\mathsf{M}|\mathsf{C}, \mathsf{S}_1, \ldots, \mathsf{S}_t, \mathsf{S}_i) = \epsilon(k)$ for any $i > t$. In the same way as above, we can use Lemma 1 to argue that this means that $\tilde{H}(\mathsf{M}|\mathsf{S}_i) \leq \sqrt{\epsilon(\lambda)}$, except with probability $\sqrt{\epsilon(\lambda)}$.

It follows that, except with negligible probability, $\mathsf{s}_1, \ldots, \mathsf{s}_t$ are good, which proves the second part of the claim.

■

### 3.3 Upper Bound

The bound in Theorem 2 is tight, up to an additive $H(\mathsf{M}) = l$. It is achieved in Constr 1, which is based on the work of Daza *et. al* [DHMR08]. (In the work of Daza *et. al*, specific assumptions are used; Constr 1 is expressed more generally.)

**Construction 1** *Let the message and each of the shared keys $\mathsf{S}_i$ be a field element. The ciphertext consists of an additional $n - t$ points on the degree-$n$ polynomial interpolating the message and the secret keys. Decryption is then done by interpolation.*

The above construction has $H(\mathsf{C}) = (n - t)l$.

## 4 Bounds Additionally Assuming an Idealized PRG

In this section, we add an idealized pseudorandom generator (PRG); an idealized functionality that takes in a random length-$\lambda$ seed, and outputs a longer random value. (As long as the output is at least one bit longer than the input, we can bootstrap the PRG to give arbitrarily long outputs. In our case, the output length that most often makes sense is $l$, the length of the message.) Our ATE algorithms are augmented with oracle access to the idealized PRG.

We make some assumptions on how the ATE protocol may use the idealized PRG:

**Definition 3.** *An* admissible *ATE-protocol satisfies the following:*

- *For any subset of receivers, any PRG-seed chosen by the sender can either be computed using what that subset of receivers knows, or has full entropy (possibly up to a negligible loss).*
- *During encryption, the sender chooses all seeds that are input to PRG uniformly, independently of anything else.*
- *The idealized PRG is not called during key generation $KG$.*

In the following we will only consider admissible protocols. The motivation for this is as follows:

– We want to make sure that an admissible protocol can be turned into a construction in the real world by replacing the idealized PRG by a real PRG construction. Now, if a seed has (essentially) full entropy in the view of the adversary then (and only then) can we use the standard security of a real PRG to conclude that the output is pseudorandom. Seeds for which the adversary has partial information are not useful in this sense, and we may as well give the adversary full information on that seed for free.
  This is why we assume that in the view of a subset of receivers, any seed that the sender chose can either be computed or has (essentially) full entropy. However, for a seed to be potentially useful it must have full entropy in the first place, which is why we assume that the sender chooses all seeds uniformly, independently of anything else.
– We assume that the idealized PRG is not called during key generation $KG$ for simplicity, because this does not cost us any generality: calls to the PRG on inputs that are independent of the resulting shared keys $\mathsf{S}$ may as well be performed later. On the other hand, calls to the PRG using shared keys as input may as well be replaced with additional calls to the idealized key exchange functionality itself (resulting, either way, in a greater amount of shared randomness).

### 4.1 Lower Bound on Ciphertext Size

**Theorem 5.** *Consider any statistically secure threshold $t$ ATE scheme in the pairwise key exchange model with additional access to an idealized PRG which takes inputs of size $\lambda$. Assume the scheme encrypts a message $\mathsf{M}$ with entropy $l$. If the scheme is admissible it holds that*

$$H(\mathsf{C}) \geq \left\lceil \frac{n-t}{2} \right\rceil \lambda + l - \delta(\lambda)$$

*for a negligible function $\delta(\lambda)$.*

For the proof of the above theorem, we define a modified broadcast ATE scheme (that is, a modified ATE scheme with $t = 0$) which we call a *multi-message random broadcast ATE (MMRB ATE) scheme*. In a multi-message random broadcast ATE scheme, different members of the set of recipients recover different independent uniformly random messages of length $l$ (where the sender can specify that more than one party recover the same message, for a total of $x$ distinct messages). The statistical security of such a scheme is defined analogously to that of a regular ATE scheme; each of the messages should have (nearly) full entropy in the presence of just the ciphertext, and in the additional presence of one of the secret keys, one or more messages loses (nearly) all entropy.

**Corollary 1 (of Theorem 3).** *For any multi-message random broadcast ATE scheme in the pairwise key exchange model that is statistically secure, for a set of independent uniformly random messages $\mathsf{M} = \{\mathsf{M}_1, \ldots, \mathsf{M}_x\}$ each with length $l$, it holds that*

$$H(\mathsf{C}) \geq \left\lceil \frac{n}{2} \right\rceil l - \delta(\lambda)$$

*for a negligible function* $\delta(\lambda)$.

*Proof.* We prove Corollary 1 in three steps:

1. It is easy to see that $H(\mathsf{C}) \geq xl - \delta_1(\lambda)$ for a negligible function $\delta_1(\lambda)$, since the ciphertext must carry information about all $x$ messages of length $l$.
2. We prove below that $H(\mathsf{C}) \geq (n-x)l - \delta_2(\lambda)$ for a negligible function $\delta_2(\lambda)$.
3. Given those two facts, it follows that $H(\mathsf{C}) \geq \lceil \frac{n}{2} \rceil l - \delta(\lambda)$ for a negligible function $\delta(\lambda)$, since the maximum of the two bounds above is clearly minimized at $x = \frac{n}{2}$.

The proof that $H(\mathsf{C}) \geq (n-x)l - \delta_2(\lambda)$ closely follows the proof of Theorem 3, with only minor modifications. In the entropy diagram, the entire set of messages $\mathsf{M}$ (where $H(\mathsf{M}) = xl$) is considered instead of a single message. We include the details below for completeness.

1. By the statistical security of the multi-message random broadcast ATE scheme, $H(\mathsf{M}) - H(\mathsf{M}|\mathsf{C}) = \epsilon(\lambda)$ for a negligible function $\epsilon$. So,

$$b + c = \epsilon(\lambda)$$

   in Figure 2.
2. For $i \in \{1, \ldots, n\}$, $g$ (in Figure 2) $= H(\mathsf{M}|\mathsf{C}, \mathsf{S}_i) \leq (x-1)l + \epsilon(\lambda)$ for a negligible function $\epsilon(\lambda)$ (since $\mathsf{C}$ together with $\mathsf{S}_i$ enable the recovery of at least one message with entropy $l$).

$$H(\mathsf{M})xl = a + b + c + g \leq a + (x-1)l + 2\epsilon(\lambda)$$

$$\Rightarrow a \geq l - 2\epsilon(\lambda)$$

   by Step 1, since we can assume $b + c = \epsilon(\lambda)$.
3. For $i \in \{1, \ldots, n\}$, $H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) = H(\mathsf{S}_i) - (a+b+d) \leq H(\mathsf{S}_i) - a$, since $b+d$ must be non-negative. Then by Step 2, we get

$$H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) \leq H(\mathsf{S}_i) - l + 2\epsilon(\lambda).$$

4. We now conclude:

$$H(\mathsf{S}_1, \ldots, \mathsf{S}_n|\mathsf{M}, \mathsf{C}) \leq \sum_{i=1}^{n} H(\mathsf{S}_i|\mathsf{M}, \mathsf{C}) \tag{22}$$

$$\leq \sum_{i=1}^{n} \big(H(\mathsf{S}_i) - l + 2\epsilon(\lambda)\big) \tag{23}$$

$$= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nl + 2n\epsilon(\lambda) \tag{24}$$

5. On the other hand, we have

$$\begin{aligned} H(\mathsf{S}_1,\ldots,\mathsf{S}_n|\mathsf{M},\mathsf{C}) &\geq H(\mathsf{S}_1,\ldots,\mathsf{S}_n) - H(\mathsf{M},\mathsf{C}) && (25)\\ &= H(\mathsf{S}_1,\ldots,\mathsf{S}_n) - (H(\mathsf{M}) + H(\mathsf{C}) - \epsilon(\lambda)) && (26)\\ &= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - H(\mathsf{M}) - H(\mathsf{C}) + \epsilon(\lambda) && (27)\\ &= \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - xl - H(\mathsf{C}) + \epsilon(\lambda), && (28) \end{aligned}$$

since $\mathsf{M}$ and $\mathsf{C}$ are independent and since the keys are all independent.

6. By combining the previous two steps, we get

$$\Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - xl - H(\mathsf{C}) \leq \Big(\sum_{i=1}^{n} H(\mathsf{S}_i)\Big) - nl + (2n-1)\epsilon(\lambda)$$

from which we conclude that

$$H(\mathsf{C}) \geq (n-x)l - \eta(\lambda),$$

where we define the negligible function $\eta(\lambda) = (2n-1)\epsilon(\lambda)$.

■

*Proof (of Theorem 5).* If the PRG is not utilized at all, then we inherit the lower bound from Theorem 4. In order for the PRG to be useful, it is clear that at least one $\lambda$-size input to that PRG must be communicated to the recipient set $\mathcal{R}$. This is because if the sender and receivers don't know any common inputs to the PRG, invocations of the PRG may as well be replaced by picking fresh random values.

Let a set of seeds for the PRG be *useful* if invoking the PRG with those seeds (in addition to performing operations unrelated to the PRG) is sufficient to decrypt.

No useful set of seeds can be communicated in such a way that an unqualified set of parties can recover it. This is because if an unqualified set of parties can learn a useful set of seeds, then the PRG once more becomes useless, since instead of using the PRG we could model the PRG output on those seeds as public information and achieve the same results.

Of course, it must be true that some useful set of seeds is recoverable by any qualified set.

Imagine a fixed maximally unqualified set of parties; fix the keys belonging to that set as public parameteres. Each remaining party must enable the reconstruction of at least one additional seed, so that the resulting qualified set of parties knows a useful set of seeds. Let $\mathcal{S}$ be the set of those additional seeds that remaining parties enable the reconstruction of.

Notice that, with overwhelming probability, we are left with a statistically secure multi-message random broadcast ATE to $n' = n - t$ parties with message length $l' = \lambda$. By Corollary 1, we have

$$H(\mathsf{C}) \geq \left\lceil \frac{n'}{2} \right\rceil l' - \delta'(\lambda) = \left\lceil \frac{n-t}{2} \right\rceil \lambda - \delta'(\lambda)$$

for a negligible function $\delta'(\lambda)$.

The ciphertext must carry an additional $l$ bits corresponding to the actual message, which must be independent of all other values.

■

### 4.2 Upper Bound

Constr 2 describes how, using an idealized PRG in addition to pairwise shared randomness, we can achieve

$$H(\mathsf{C}) = (n-t)\lambda + l,$$

which is only a factor of two off from the lower bound.

**Construction 2** *The sender chooses a random PRG seed, uses the ATE scheme from Constr 1 to encrypt this seed to the receivers, and uses the PRG output on this seed to one-time-pad-encrypt the message.*

## 5 Bounds Assuming Idealized Multiparty Key Exchange

In this section, we explore what happens when we allow the sender to agree on a shared key with more than one receiver at a time (that is, if we assume *multiparty* key exchange). As we mentioned in the introduction, multiparty non interactive key exchange (multiparty NIKE) can be constructed from multilinear maps, indistinguishability obfuscation [BZ14], universal samplers [HJK$^+$16,GPSZ17], or encryption combiners [MZ17].

*Remark 1.* It might seem that allowing access to NIKE defeats the point of ATE, which is to avoid the need for correlated randomness; however, just like two-party non-interactive key exchange, multiparty NIKE is something that can be used to bootstrap independently generated public-private key pairs into a limited notion of correlated randomness without requiring interaction.

### 5.1 Building Block: Pseudorandom Secret Sharing

Both our positive and our negative results in the multiparty key exchange model leverage *pseudorandom secret sharing*, which is a technique for the local (that is, non-interactive) conversion of a replicated secret sharing to a Shamir secret sharing.

A *replicated secret sharing* for the $(t+1)$-out-of-$n$ threshold access structure proceeds as follows. First, the dealer splits the secret $\mathsf{M}$ into $\binom{n}{t}$ additive secret shares, where each share $\mathsf{r}_{\mathcal{A}}$ corresponds to a different maximally unqualified set $\mathcal{A}$ of size $t$. Then, the complement of each set $\mathcal{A}$ (that is, the $n-t$ parties that are *not* in $\mathcal{A}$) are all given $\mathsf{r}_{\mathcal{A}}$. It is then clear that any maximally unqualified set $\mathcal{A}$ is only missing knowledge of one share $\mathsf{r}_{\mathcal{A}}$, which any additional party holds.

Pseudorandom secret sharing [CDI05] locally converts such a replicated secret sharing into a Shamir secret sharing (a degree-$t$ polynomial $f$ with $f(0) = \mathsf{M}$ as the secret, and $f(i) = \mathsf{s}_i$ as party $i$'s share for $i \in [1, \ldots, n]$). Pseudorandom secret sharing proceeds as follows: let $f_{\mathcal{A}}$ be the degree-$t$ polynomial such that $f_{\mathcal{A}}(0) = 1$, and $f_{\mathcal{A}}(i) = 0$ for all $i \in [n] \backslash \mathcal{A}$. Each player $\mathcal{P}_i$ can then compute their Shamir share as

$$\mathsf{s}_i = \sum_{\mathcal{A} \subseteq [n] : |\mathcal{A}| = n-t, i \in \mathcal{A}} \mathsf{r}_{\mathcal{A}} f_{\mathcal{A}}(i).$$

Cramer, Damgård and Ishai [CDI05] also prove a lower bound, stated in Theorem 6.

**Theorem 6 (From [CDI05]).** *Fewer than $\binom{n}{t}$ independent random values shared among various subsets of parties cannot be locally converted into a $(t+1)$-out-of-$n$ threshold secret sharing.*

### 5.2 Lower Bounding the Key Size When $H(\mathsf{C}) = H(\mathsf{M})$

**Theorem 7.** *For any ATE scheme with threshold $t = \theta(n)$ in the multiparty key exchange model that is perfectly secure, if $H(\mathsf{C}) = H(\mathsf{M})$, then keys of exponential size are necessary.*

*Proof.* If $H(\mathsf{C}) = H(\mathsf{M})$, then for any distribution of keys, there is exactly one ciphertext that corresponds to any given message. Therefore, choosing a ciphertext at random (without considering the keys) will always give a valid ciphertext that consistently decrypts to some message. Choosing the keys and ciphertext simultaneously independently at random thus produces a random $(t+1)$-out-of-$n$ secret sharing (where the ciphertext is simply an additional random value given to all parties). So, the exponential lower bound by Cramer *et. al* [CDI05] (Theorem 6) on amount of independent randomness that can be converted into a $(t+1)$-out-of-$n$ secret sharing applies. ■

### 5.3 Upper Bounds

*Upper Bounds with Optimal Ciphertext Size* Constr 3 achieves ciphertext size independent of the number of parties by leveraging the techniques of replicated or pseudorandom secret sharing. However, the price it pays is exponential key size (that is, the sender and each of the receivers must use an exponential number of shared random values).

**Construction 3** *The sender shares a different independently random field element with the compliment of each maximally unqualified subset (that is, with each set of size $n-t$). This is a $(t+1)$-out-of-$n$ replicated secret sharing of a random value (which, if desired, can then be converted to a Shamir secret sharing of the same random value using the pseudorandom secret sharing technique [CDI05]). This shared random value can then be used directly for encryption (e.g. as a one time pad).*

Constr 3 produces ciphertexts of optimal size; $H(\mathsf{C}) = H(\mathsf{M})$. The number $q$ of parties (including the sender) sharing any given random value is fixed at $q = n - t + 1$.

Next, we consider other constructions leveraging $q$-NIKE that do not achieve optimal ciphertext size, and thus avoid paying the exponential overhead we have demonstrated is necessary for such optimality.

*Upper Bounds with $t = 0$* In Constr 4, we show that for broadcast (that is, $t = 0$), we can use any slight increase in the number $q$ of parties sharing randomness to decrease the required ciphertext size.

**Construction 4** *The sender shares a different independently random value with enough size-$(q-1)$ subsets of receivers to cover the entire size-$n$ set of receivers. (We will need $\lceil \frac{n}{q-1} \rceil$ such values.) To broadcast a message, the sender uses each of those random values individually to encrypt the message (e.g. as a one time pad).*

Constr 4 produces ciphertexts of size $\lceil \frac{n}{q-1} \rceil l$.

*Upper Bounds with $t > 0$* Next, we turn to a completely different approach that works for $t > 0$, as long as $t$ is small compared to $n$. Let $c = t + 1$, so that the goal is to allow any $c$ of the $n$ recipients to decrypt while $t$ or fewer cannot.

We first need a couple of technical results. Some notation: a *$k$-split* $A$ of the $n$ recipients will be a randomly chosen partitioning of $[n] = [1, \ldots, n]$ into disjoint subsets $A = [A_1, \ldots, A_k]$. (Equivalently, put each recipient into a bucket chosen randomly from among $k$ buckets.) Consider a subset $C \subset [n]$ of size $c$. We say that $C$ is happy about a split $A$ if no two members of $C$ are in the same bucket.

**Lemma 2.** *For $k = 2c^2$, for each fixed $C \subset [n]$ of size $c$, the probability that $C$ is not happy about a random $k$-split is at most $\frac{1}{2}$.*

*Proof.* For $i \in C$, let $E_i$ be the event that recipient $i$ lands in the same bucket as some other member of $C$. Clearly $\Pr[E_i] \leq \frac{c}{k} = \frac{c}{2c^2} = \frac{1}{2c}$. $C$ is unhappy about the split if and only if $E_i$ occurs for some $i \in C$. By the union bound, the probability that $C$ is not happy is at most $\frac{c}{2c} = \frac{1}{2}$. ■

Consider a set $S$ of $s$ random splits. We say $S$ is *$c$-good* if it holds that *every* subset of size $c$ is happy about at least one of the splits.

**Lemma 3.** *For any number $n$ of recipients, $c = t + 1$ and $k = 2c^2$, there exists a set $S$ of $c\log(n)$ $k$-splits that is $c$-good.*

*Proof.* Consider a random set $S$ of $s = c\log(n)$ splits. Let $C \subset [n]$ be a fixed subset of size $c$. The probability that $C$ is unhappy about every split in $S$ is at most $2^{-s}$ by Lemma 2. The number of subsets of size $c$ is $\binom{n}{c} \leq n^c$, so by the union bound the probability that some $C$ is unhappy — and thus that $S$ is not $c$-good — is is less than $n^c \cdot 2^{-s} = n^c \cdot 2^{-c\log(n)} = n^c \cdot n^{-c} = 1$. It follows that there must exist a set $S$ of splits that is $c$-good. ■

**Lemma 4.** *For any number $n$ of recipients and $c = 2$, we can get a set $S$ of $\log(n)$ 2-splits that is 2-good.*

*Proof.* Represent each recipient $i \in [n]$ in binary. Let the $j$th bit of that representation denote whether $i$ is in $A_0$ (equivalently $A_2$) or $A_1$ in the $j$th 2-split. Clearly this set of splits is 2-good, since the binary representation of every two parties must differ in at least one place. ■

Finally, in Constr 5 we observe that from a ($c$-good) set of $s$ splits $S$, we can easily obtain an $c$-of-$n$ ATE.

**Construction 5** *Given a $c$-good set of splits $S$, for each split $A = [A_1, \ldots, A_k] \in S$, consider the $A_i$'s as virtual receivers. Use Constr 1 to build an ATE where the $A_i$'s are receivers and any $c$ of the $k$ virtual receivers can decrypt (i.e., $t = c-1$). The only change is that where Constr 1 would call a 2-NIKE to share a key with $A_i$, instead we invoke a multiparty NIKE to share a key with all recipients in $A_i$. The ciphertext is the concatenation of all the $s$ ciphertexts, one from each of the splits.*

Since a subset $C$ can decrypt a ciphertext induced by a split if $C$ is happy about the split, and all subsets of size $c$ are happy about at least one split in a $c$-good set of splits, it is straightforward to see that Constr 5 gives a $c$-of-$n$ ATE.

Using the lemmas above, we can get a $c$-good set $S$ of $s$ $k$-splits with $s = c\log(n)$ and $k = 2c^2$. Constr 1 gives ciphertexts of size $(k-(c-1))l$, so Constr 5 gives ciphertexts of size $s(k-(c-1))l = c\log(n)(2c^2 - c + 1)l = O(c^3\log(n)l)$, where $l$ is the length of the message.

One can also see that the expected size of a subset in a split is $\frac{n}{k} = \frac{n}{2c^2}$, so we can use this construction with a $q$-NIKE, where $q$ is roughly $\frac{n}{2c^2}$. Of course, even if this construction in principle works for any $n$ and $c$, it is only really interesting when $c^3\log(n)$ is significantly smaller than $n$, since otherwise it is not better than using Constr 1 directly.

For $c = 2$ ($t = 1$), Lemma 4 gives a 2-good set of $\log(n)$ 2-splits, which allows us to get ciphertexts of size $s(k-(c-1))l = \log(n)(2-2+1)l = \log(n)l$.

## References

BZ14. Dan Boneh and Mark Zhandry. Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation. In Juan A. Garay

and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 480–499. Springer, Heidelberg, August 2014.

CDI05. Ronald Cramer, Ivan Damgård, and Yuval Ishai. Share conversion, pseudorandom secret-sharing and applications to secure computation. In Joe Kilian, editor, *TCC 2005*, volume 3378 of *LNCS*, pages 342–362. Springer, Heidelberg, February 2005.

DDFY94. Alfredo De Santis, Yvo Desmedt, Yair Frankel, and Moti Yung. How to share a function securely. In *26th ACM STOC*, pages 522–533. ACM Press, May 1994.

DHMR08. Vanesa Daza, Javier Herranz, Paz Morillo, and Carla Ràfols. Ad-hoc threshold broadcast encryption with shorter ciphertexts. *Electron. Notes Theor. Comput. Sci.*, 192(2):3–15, May 2008.

DP08. Cécile Delerablée and David Pointcheval. Dynamic threshold public-key encryption. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 317–334. Springer, Heidelberg, August 2008.

GPSZ17. Sanjam Garg, Omkant Pandey, Akshayaram Srinivasan, and Mark Zhandry. Breaking the sub-exponential barrier in obfustopia. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part II*, volume 10211 of *LNCS*, pages 156–181. Springer, Heidelberg, April / May 2017.

HJK+16. Dennis Hofheinz, Tibor Jager, Dakshita Khurana, Amit Sahai, Brent Waters, and Mark Zhandry. How to generate and use universal samplers. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 715–744. Springer, Heidelberg, December 2016.

MR13. Tal Moran and Alon Rosen. There is no indistinguishability obfuscation in pessiland. Cryptology ePrint Archive, Report 2013/643, 2013. `http://eprint.iacr.org/2013/643`.

MZ17. Fermi Ma and Mark Zhandry. Encryptor combiners: A unified approach to multiparty NIKE, (H)IBE, and broadcast encryption. Cryptology ePrint Archive, Report 2017/152, 2017. `http://eprint.iacr.org/2017/152`.

RSY18. Leonid Reyzin, Adam Smith, and Sophia Yakoubov. Turning hate into love: Homomorphic ad hoc threshold encryption for scalable mpc. Cryptology ePrint Archive, Report 2018/997, 2018. `https://eprint.iacr.org/2018/997`.