An introduction to the theory of unconditionally secure message authentication using the constructive cryptography framework

Dimiter Ostrev*

Abstract

We provide an introduction to certain ideas from the theory of unconditionally secure message authentication. We explain the notions of impersonation and substitution attacks, and explain how protection against these two types of attack implies composable, information theoretic security. We explain the relation of authentication protocols to universal hashing. We give both probabilistic and explicit constructions proving the existence of one way authentication protocols using a short secret key and we prove matching lower bounds on the required secret key size.

Then, we turn attention to interactive authentication protocols. We explain the message size reduction technique used by Gemmell and Naor and later Naor, Segev and Smith, and how it leads to protocols with secret key size independent of the message length. We also prove a matching lower bound on the secret key entropy. We generalize the lower bound proof of Naor, Segev and Smith and remove the assumption that the message is revealed in the first flow of the protocol.

1 Introduction

Message authentication is one of the most important primitives in cryptography. It has direct real world applications, for example in ensuring that the order for a financial transaction comes from somebody authorized to perform it and not a criminal. It is also used as a subroutine in other cryptographic protocols, where it serves to protect against man in the middle attacks.

In defining the security of message authentication protocols, we distinguish between computational and unconditional security. In the first case, the definition and proof of security rests on the assumption that the adversary has limited computational resources, and on the conjecture that a certain problem cannot be solved within the specified resource bound. In the second case, the protocol is guaranteed to remain secure against adversaries with unbounded resources.

*Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, 6, Avenue de la Fonte, L-4364 Esch-sur-Alzette, Luxembourg, e-mail: dimiter.ostrev@uni.lu

Besides the obviously greater assurance, a further advantage of unconditionally secure protocols is that often they are computationally more efficient, i.e. the operations that sender and receiver must perform require less time and/or memory; this was pointed out in [18]. On the other hand, a disadvantage of unconditionally secure protocols is that they consume secret key for each authenticated message.

The goal of this document is to provide a self contained introduction to the theory of unconditionally secure message authentication. The main prerequisites for understanding the exposition are mathematical maturity and knowledge of certain basic concepts in probability theory.

The field of unconditionally secure message authentication is large, and we cannot cover all results in one self-contained tutorial; therefore, we need some criterion to guide the selection of topics. We focus on the question: how much secret key is needed to accomplish unconditionally secure message authentication? With this criterion in mind, we select those concepts, ideas and proof techniques that lead to an answer to this question. We cover two scenarios: one way authentication, in which information flows only from sender to receiver and interactive authentication, in which there is a conversation between sender and receiver over an insecure channel.

We start our exposition in section 2, where we develop the basic ides of Constructive Cryptography. Constructive Cryptography is an approach to defining the security of protocols in such a way that a protocol composes safely with other protocols and remains secure in an arbitrary environment. Then, we consider the scenario of one way authentication in section 3. We cover impersonation and substitution attacks, the relation to universal classes of hash functions, probabilistic and explicit constructions of authentication protocols with small secret key size, and a proof of a matching lower bound on the secret key size. Then, we turn attention to the interactive scenario in section 4. We show how interactive protocols can achieve secret key size that depends only on the desired security level and not on the message length. We also prove a matching lower bound on the secret key size in the interactive case.

2 The Constructive Cryptography Framework

We would like to be able to put together cryptographic protocols into more complex secure systems with the same ease and simplicity with which one can put together Lego bricks to form complex structures. Unfortunately, things are not quite so simple in cryptography; there are known examples of protocols which satisfy intuitively appealing definitions of security in isolation, and yet they become insecure when put together. One such example was given by [12]: it is possible for a QKD protocol to ensure that the adversary has negligible accessible information about the generated key, and yet this key cannot be used for one time pad encryption of a message whose header is known to the adversary.

It turns out that such undesirable examples can be avoided if a carefully

selected security criterion is used for each building block. Frameworks of composable security such as [3, 1, 16] have been developed, and protocols that are secure according to the definition of a given framework compose safely with other protocols within that framework.

In the present paper, we will also need to put together cryptographic building blocks. For our purposes, the Constructive Cryptography framework [14], a special case of [16], will be convenient. Constructive Cryptography can model two honest users, Alice and Bob, and an adversary Eve.

The security definition in Constructive Cryptography, as in other frameworks, follows the real-world ideal-world paradigm: a system is defined as secure if it is indistinguishable from some ideal system. Constructive Cryptography differs from other frameworks in its algebraic approach: it introduces an algebra of cryptographic resources and converters with suitable composition operations and a distance metric on that algebra. This makes it possible to prove statements about cryptographic systems on an abstract level, in a manner analogous to how one can prove theorems in, for example, group theory.

We proceed to give more details about the systems, the operations for putting them together, and the definition of distance in Constructive Cryptography. The first concept that we need is that of resources. A resource is a system with interfaces for Alice, Bob and Eve which allow them to enter inputs and receive outputs. It is meant to capture our most general intuitive idea of what a cryptographic functionality (or any distributed computing functionality) does.

On the set of all resources, we define a parallel composition operation. Given resources $\mathcal{R}, \mathcal{S}$, their parallel composition, denoted $\mathcal{R}\|\mathcal{S}$, is another resource that provides Alice, Bob and Eve with access to the interfaces of both $\mathcal{R}$ and $\mathcal{S}$. This captures our intuition that users may have access to several functionalities.

The next idea that we need is that of a converter: a converter is a system with two interfaces that can receive inputs and produce outputs. We understand intuitively the role of converters when we consider their combination with resources. Given a converter α and a resource $\mathcal{R}$, $\alpha_A\mathcal{R}$ is another resource. Alice's interface to $\alpha_A\mathcal{R}$ is the outside interface of the converter α, while the inside interface of the converter α exchanges inputs and outputs with interface A of the resource $\mathcal{R}$. Bob's and Eve's interfaces to $\alpha_A\mathcal{R}$ are interfaces B, E of $\mathcal{R}$. $\alpha_B\mathcal{R}$ and $\alpha_E\mathcal{R}$ are defined similarly, but with the converter attached to Bob's and Eve's interfaces respectively.

An example may be helpful at this point. Let $\mathcal{N}$ be a noisy channel resource: it allows Alice to input a string, and outputs a corrupted version of the string to Bob. Let E, D be the encoding and decoding algorithms of a suitable error correcting code, which we think of as converters. Then $E_A D_B \mathcal{N}$ is (close to) a noiseless channel resource. In this example, we were not interested in the role of the adversary Eve, so we did not specify her interfaces.

Parallel and sequential composition of converters are defined in a natural way. Given a resource $\mathcal{R}$, an interface i, and two converters α, β, the sequential composition of α and β attached to interface i of $\mathcal{R}$ is defined by

$$(\alpha\beta)_i\mathcal{R} = \alpha_i(\beta_i\mathcal{R})$$

Given two resources $\mathcal{R}, \mathcal{S}$, an interface i and two converters α, β, the parallel composition of α and β attached to interface i of $\mathcal{R}\|\mathcal{S}$ is defined by

$$(\alpha\|\beta)_i(\mathcal{R}\|\mathcal{S}) = (\alpha_i\mathcal{R})\|(\beta_i\mathcal{S})$$

Depending on the role and meaning we attach to a converter, we sometimes refer to it as a protocol, a filter or a simulator. We refer to a converter, or a set of converters, as a protocol, if we think of their role as enabling honest parties to perform some task of interest; in the example above, we can think of the encoding and decoding algorithms E_A, D_B as a protocol that allows Alice and Bob to faithfully transmit messages over the noisy channel. We think of a converter as a filter if we attach to it the role of preventing a malicious user from doing any harm; this is illustrated in a modification to our first example. Suppose that the noisy channel $\mathcal{N}$ also sends a (possibly corrupted) version of Alice's input to Eve, a situation that has been considered in information theoretic cryptography under the names wire-tap channel [26] and its generalization broadcast channel [6]. Then a filter would be a converter $\flat$ that takes the input from the channel and gives no output to Eve. Thus, $E_A D_B \flat_E \mathcal{N}$ is a noiseless channel from Alice to Bob that keeps the message completely confidential from Eve. Finally, we refer to a converter as a simulator if we think of its role as making the interface of one resource appear as the interface of another. The role of simulators will become clear later when we consider the definition of construction.

The next concept is meant to capture our intuition that some functionalities are "close to" each other while others are "far apart." We already used this idea in our first example, when we said that $E_A D_B \mathcal{N}$ is "close to" a noiseless channel, meaning that despite encoding and decoding with a suitable error correcting code, it is still possible that Bob receives the wrong message, but this occurs only with small probability. Formally, we define "close to" and "far apart" for functionalities by the following:

Definition 1. *A distance on a set of resources is a function d taking pairs of resources to $\mathbb{R}_+$ with the properties:*

1. *(identity) $d(\mathcal{R}, \mathcal{R}) = 0$*
2. *(symmetry) $d(\mathcal{R}, \mathcal{S}) = d(\mathcal{S}, \mathcal{R})$*
3. *(triangle inequality) $d(\mathcal{R}, \mathcal{T}) \leq d(\mathcal{R}, \mathcal{S}) + d(\mathcal{S}, \mathcal{T})$*
4. *(non-increasing under a converter) $d(\alpha_i\mathcal{R}, \alpha_i\mathcal{S}) \leq d(\mathcal{R}, \mathcal{S})$*
5. *(non-increasing under a resource in parallel) $d(\mathcal{R}\|\mathcal{T}, \mathcal{S}\|\mathcal{T}) \leq d(\mathcal{R}, \mathcal{S})$*

holding for any three resources $\mathcal{R}, \mathcal{S}, \mathcal{T}$ and any converter α attached to any interface i.

The first three properties in the definition: identity, symmetry, and the triangle inequality, form the mathematical definition of a pseudometric; thus, we should expect any reasonable notion of distance to satisfy them. We do

not require $d(\mathcal{R}, \mathcal{S}) = 0 \Rightarrow \mathcal{R} = \mathcal{S}$, the property that would convert d from a pseudometric to a metric; intuitively, this corresponds to allowing there to be different ways to obtain the same functionality.

The fourth and fifth property, non-increasing under a converter or a resource in parallel, can be intuitively justified in two related ways. First, we anticipate that the concrete way to instantiate a distance d on the set of resources will be to define $d(\mathcal{R}, \mathcal{S})$ to be the supremum of the advantage of any distinguisher in determining whether it is interacting with $\mathcal{R}$ or with $\mathcal{S}$. Such a construction will naturally lead to d satisfying the fourth and fifth property, because a subset of all distinguishers apply the converter α or add the resource $\mathcal{T}$ in parallel. Second, our goal in introducing the distance d is to capture the idea of a real functionality being close to an ideal functionality in an arbitrary context, and the fourth and fifth property of the definition are needed to make that work.

We proceed to make precise the idea that one way to construct a distance on the set of resources is to take the supremum over the choice of distinguisher of the distinguishing advantage.

Definition 2. *A distinguisher $\mathcal{D}$ is a system with four interfaces. Three of the interfaces of $\mathcal{D}$ connect to the Alice, Bob, and Eve interfaces of a resource. The fourth interface outputs 0 or 1.*

For distinguisher $\mathcal{D}$ and resource $\mathcal{R}$, we define $\mathcal{DR}$ to be the random variable that is the output of $\mathcal{D}$ when connected to $\mathcal{R}$.

For a distinguisher $\mathcal{D}$ and two resources $\mathcal{R}, \mathcal{S}$, the advantage of $\mathcal{D}$ in distinguishing between $\mathcal{R}$ and $\mathcal{S}$ is defined as the statistical distance between the random variables $\mathcal{DR}$ and $\mathcal{DS}$; in other words,

$$Adv_{\mathcal{D}}(\mathcal{R}, \mathcal{S}) = |\mathbb{P}(\mathcal{DR} = 1) - \mathbb{P}(\mathcal{DS} = 1)|$$

Finally, we define the function d taking pairs of resources to $\mathbb{R}_+$ by

$$d(\mathcal{R}, \mathcal{S}) = \sup_{\mathcal{D}} Adv_{\mathcal{D}}(\mathcal{R}, \mathcal{S})$$

It can be checked, assuming natural properties of the operations of connecting the interfaces of interacting systems, that d defined this way satisfies the five properties required of a distance on the set of resources.

Proposition 1. *The function d from Definition 2 satisfies the five conditions of Definition 1.*

Proof. $d(\mathcal{R}, \mathcal{R}) = \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{DR} = 1) - \mathbb{P}(\mathcal{DR} = 1)| = 0$, gives the first property.

$$d(\mathcal{R}, \mathcal{S}) = \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{DR} = 1) - \mathbb{P}(\mathcal{DS} = 1)| = \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{DS} = 1) - \mathbb{P}(\mathcal{DR} = 1)| = d(\mathcal{S}, \mathcal{R})$$

gives the second property.

$$\begin{aligned} d(\mathcal{R},\mathcal{T}) &= \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{D}\mathcal{R}=1) - \mathbb{P}(\mathcal{D}\mathcal{T}=1)| \\ &= \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{D}\mathcal{R}=1) - \mathbb{P}(\mathcal{D}\mathcal{S}=1) + \mathbb{P}(\mathcal{D}\mathcal{S}=1) - \mathbb{P}(\mathcal{D}\mathcal{T}=1)| \\ &\leq \sup_{\mathcal{D}} (|\mathbb{P}(\mathcal{D}\mathcal{R}=1) - \mathbb{P}(\mathcal{D}\mathcal{S}=1)| + |\mathbb{P}(\mathcal{D}\mathcal{S}=1) - \mathbb{P}(\mathcal{D}\mathcal{T}=1)|) \\ &\leq \sup_{\mathcal{D}} |\mathbb{P}(\mathcal{D}\mathcal{R}=1) - \mathbb{P}(\mathcal{D}\mathcal{S}=1)| + \sup_{\mathcal{D}'} |\mathbb{P}(\mathcal{D}'\mathcal{S}=1) - \mathbb{P}(\mathcal{D}'\mathcal{T}=1)| \\ &= d(\mathcal{R},\mathcal{S}) + d(\mathcal{S},\mathcal{T}) \end{aligned}$$

gives the third property.

$$\begin{aligned} d(\mathcal{R},\mathcal{S}) = \sup_{\mathcal{D}} Adv_{\mathcal{D}}(\mathcal{R},\mathcal{S}) &\geq \sup_{\mathcal{D}} Adv_{\mathcal{D}\alpha_i}(\mathcal{R},\mathcal{S}) \\ &= \sup_{\mathcal{D}} Adv_{\mathcal{D}}(\alpha_i\mathcal{R},\alpha_i\mathcal{S}) = d(\alpha_i\mathcal{R},\alpha_i\mathcal{S}) \end{aligned}$$

gives the fourth property, where in the first step we used the fact that a subset of all distinguishers applies the converter α to interface i, and in the second step we noticed that we can alternatively think of the converter α as associated to the resources.

$$\begin{aligned} d(\mathcal{R},\mathcal{S}) = \sup_{\mathcal{D}} Adv_{\mathcal{D}}(\mathcal{R},\mathcal{S}) &\geq \sup_{\mathcal{D}} Adv_{\mathcal{D}[\cdot\|\mathcal{T}]}(\mathcal{R},\mathcal{S}) \\ &= \sup_{\mathcal{D}} Adv_{\mathcal{D}}(\mathcal{R}\|\mathcal{T},\mathcal{S}\|\mathcal{T}) = d(\mathcal{R}\|\mathcal{T},\mathcal{S}\|\mathcal{T}) \end{aligned}$$

gives the fifth property, where in the first step we used the fact that a subset of all distinguishers applies the resource $\mathcal{T}$ in parallel (denoted by $[\cdot\|\mathcal{T}]$ in the equation), and in the second step we noticed that we can alternatively think of $[\cdot\|\mathcal{T}]$ as associated to the resources. □

Having introduced resources, converters, and distance, we are ready to define construction. In Constructive Cryptography, the definition of security follows the real world ideal world paradigm: a protocol is secure if it can construct an ideal resource from some real resource. Formally, we have

Definition 3. *A protocol $\pi = (\pi_A, \pi_B)$ consisting of converters for Alice and Bob constructs the resource $\mathcal{S}$ from the resource $\mathcal{R}$ within ϵ, denoted $\mathcal{R} \xrightarrow{\pi,\epsilon} \mathcal{S}$, if the following two conditions hold:*

1. *(Close with Eve blocked) $d(\pi_A\pi_B\sharp_E\mathcal{R}, \flat_E\mathcal{S}) \leq \epsilon$*
2. *(Close with full access for Eve) There exists a simulator σ_E such that*

$$d(\pi_A\pi_B\mathcal{R}, \sigma_E\mathcal{S}) \leq \epsilon$$

When we apply this definition to cryptographic functionalities, we typically have in mind the following interpretation: $\mathcal{S}$ is the goal, the ideal functionality that honest users want to achieve. $\mathcal{R}$ is the real resource that they have available. The protocol π is required to construct $\mathcal{S}$ from $\mathcal{R}$ in two scenarios: without adversary, with the filters applied as in condition 1, and with adversary, as in condition 2.

Since the adversary interface to $\mathcal{S}$ and $\mathcal{R}$ may look different, we allow in condition 2 the choice of a converter σ_E whose job is to make the view from the adversary interface as close as possible between the real and ideal resources. If the ideal functionality $\mathcal{S}$ captures our intuition for "secure against any adversary", then we need not worry about attaching the simulator σ_E to $\mathcal{S}$: a subset of all adversaries also interacts with $\mathcal{S}$ through σ_E. Thus the supremum over all adversaries of the "damage" that can be done to $\sigma_E\mathcal{S}$ is less than or equal to the supremum over all adversaries of the "damage" that can be done to $\mathcal{S}$.

Constructions can compose both in parallel and in sequence [14, Theorem 1]:

Theorem 1. *1. If* $\mathcal{R} \xrightarrow{\pi,\epsilon} \mathcal{S}$ *and* $\mathcal{R}' \xrightarrow{\pi',\epsilon'} \mathcal{S}'$ *then* $\mathcal{R}\|\mathcal{R}' \xrightarrow{\pi\|\pi',\epsilon+\epsilon'} \mathcal{S}\|\mathcal{S}'$.

2. If $\mathcal{R} \xrightarrow{\pi,\epsilon} \mathcal{S}$ *and* $\mathcal{S} \xrightarrow{\tau,\delta} \mathcal{T}$ *then* $\mathcal{R} \xrightarrow{\tau\pi,\epsilon+\delta} \mathcal{T}$

3. For the identity protocol $\mathbf{1} = (\mathbf{1}_a, \mathbf{1}_b)$*, and any resource* $\mathcal{R}$*,* $\mathcal{R} \xrightarrow{\mathbf{1},0} \mathcal{R}$.

Proof. The proof applies the properties of composition operations and distance in a natural way.

First, we consider parallel composition:

$$\begin{aligned} &d(\pi_A\pi_B\sharp_E\mathcal{R}\|\pi'_A\pi'_B\sharp'_E\mathcal{R}', \flat_E\mathcal{S}\|\flat'_E\mathcal{S}') \\ &\leq d(\pi_A\pi_B\sharp_E\mathcal{R}\|\pi'_A\pi'_B\sharp'_E\mathcal{R}', \flat_E\mathcal{S}\|\pi'_A\pi'_B\sharp'_E\mathcal{R}') + d(\flat_E\mathcal{S}\|\pi'_A\pi'_B\sharp'_E\mathcal{R}', \flat_E\mathcal{S}\|\flat'_E\mathcal{S}') \\ &\leq d(\pi_A\pi_B\sharp_E\mathcal{R}, \flat_E\mathcal{S}) + d(\pi'_A\pi'_B\sharp'_E\mathcal{R}', \flat'_E\mathcal{S}') \leq \epsilon + \epsilon' \end{aligned}$$

where we have used the triangle inequality, then monotonicity under a resource in parallel. Similarly,

$$\begin{aligned} &d(\pi_A\pi_B\mathcal{R}\|\pi'_A\pi'_B\mathcal{R}', \sigma_E\mathcal{S}\|\sigma'_E\mathcal{S}') \\ &\quad\leq d(\pi_A\pi_B\mathcal{R}\|\pi'_A\pi'_B\mathcal{R}', \sigma_E\mathcal{S}\|\pi'_A\pi'_B\mathcal{R}') + d(\sigma_E\mathcal{S}\|\pi'_a\pi'_B\mathcal{R}', \sigma_E\mathcal{S}\|\sigma'_E\mathcal{S}') \\ &\quad\leq d(\pi_A\pi_b\mathcal{R}, \sigma_E\mathcal{S}) + d(\pi'_A\pi'_B\mathcal{R}', \sigma'_E\mathcal{S}') \end{aligned}$$

where σ, σ' are the two simulators whose existence is guaranteed by the two construction statements $\mathcal{R} \xrightarrow{\pi,\epsilon} \mathcal{S}$ and $\mathcal{R}' \xrightarrow{\pi',\epsilon'} \mathcal{S}'$. Thus, $\mathcal{R}\|\mathcal{R}' \xrightarrow{\pi\|\pi',\epsilon+\epsilon'} \mathcal{S}\|\mathcal{S}'$, as needed.

Next, we consider sequential composition:

$$\begin{aligned} &d(\tau_A\tau_B\pi_A\pi_B\sharp_E\mathcal{R}, \diamond_E\mathcal{T}) \\ &\qquad\leq d(\tau_A\tau_B\pi_A\pi_B\sharp_E\mathcal{R}, \tau_A\tau_B\flat_E\mathcal{S}) + d(\tau_A\tau_B\flat_E\mathcal{S}, \diamond_E\mathcal{T}) \\ &\qquad\leq d(\pi_A\pi_B\sharp_E\mathcal{R}, \flat_E\mathcal{S}) + d(\tau_A\tau_B\flat_E\mathcal{S}, \diamond_E\mathcal{T}) \leq \epsilon + \delta \end{aligned}$$

where we have applied the triangle inequality, then monotonicity under a converter. Similarly,

$$\begin{aligned} &d(\tau_A\tau_B\pi_A\pi_B\mathcal{R}, \sigma_E\rho_E\mathcal{T}) \\ &\qquad \leq d(\tau_A\tau_B\pi_A\pi_B\mathcal{R}, \tau_A\tau_B\sigma_E\mathcal{S}) + d(\tau_A\tau_B\sigma_E\mathcal{S}, \sigma_E\rho_E\mathcal{T}) \\ &\qquad\qquad \leq d(\pi_A\pi_B\mathcal{R}, \sigma_E\mathcal{S}) + d(\tau_A\tau_B\mathcal{S}, \rho_E\mathcal{T}) \leq \epsilon + \delta \end{aligned}$$

where σ, ρ are the two simulators whose existence is guaranteed by the construction statements $\mathcal{R} \xrightarrow{\pi,\epsilon} \mathcal{S}$ and $\mathcal{S} \xrightarrow{\tau,\delta} \mathcal{T}$. Thus, $\mathcal{R} \xrightarrow{\tau\pi,\epsilon+\delta} \mathcal{T}$, as needed.

The third part of the theorem, concerning the identity protocol, is immediate. □

3 One way authentication

In this section, we present certain results on one way authentication. We use "one way" to refer to protocols that have a single flow from sender to receiver; this is in contrast to the interactive authentication protocols we consider in Section 4 where there is a conversation between sender and receiver on an insecure channel.

We start in subsection 3.1 with a description of the message authentication problem between two parties, a sender and a receiver, who communicate over a channel under the control of an adversary. In subsection 3.2 we present two possible attacks that the adversary may perform: these are the impersonation and substitution attacks. Then, in subsection 3.3 we motivate our focus on these two types of attack by proving that protection against impersonation and substitution attacks for one way authentication implies that the protocol provides composable security in the sense of Abstract Cryptography.

We continue by showing the close connection between authentication protocols and universal classes of hash functions. We start in subsection 3.4 with the motivating example of authentication by a random function, and this leads us to the definition of an almost strongly universal$_2$ class of hash functions in subsection 3.5.

We then devote ourselves to establishing the fundamental resource requirements for one way authentication. In subsection 3.6 we give a probabilistic construction of a good one way authentication protocol, and in subsection 3.7 we give a lower bound on the secret key size required of any good one way authentication protocol. Thus, the fundamental requirement on the secret key size lies between the lower bound of subsection 3.7 and the probabilistic construction of subsection 3.6, and the two match asymptotically up to a constant factor.

We continue by introducing almost XOR universal classes of hash functions in subsection 3.8. Almost XOR universal classes of hash functions can be used to construct almost universal$_2$ classes, and we show in subsection 3.9 that they have a similar probabilistic construction and obey a similar lower bound on the secret key size as do almost universal$_2$ classes of hash functions. In subsection

3.10 we give explicit constructions of almost XOR universal and almost strongly universal$_2$ classes that come close to the secret key size achieved by probabilistic constructions. We will see almost XOR universal classes again in section 4 where they play a role in the message size reduction step of the interactive authentication protocol.

We have gathered references for this section in subsection 3.11.

3.1 Setup

In the message authentication problem, a sender wishes to communicate a message to a receiver over a channel that is under the control of an adversary who can stop, delay or modify messages in transit, and who can insert messages of her own. It is convenient to give the names Alice to the sender, Bob to the receiver and Eve to the adversary. The focus is not on ensuring the secrecy of the message m from Eve, but on ensuring that Bob does not get any messages that were not sent by Alice.

Alice and Bob need to have some advantage over Eve if they are to have a chance. Here, we focus on the case where this advantage is the knowledge of a shared secret key k that is unknown to Eve. Using this secret key, Alice and Bob encode and decode in order to transmit messages. To send message m, Alice sends the codeword $c = f(k, m)$ on the insecure channel; upon receipt of a codeword c', Bob applies a decoding rule to obtain $m' = \phi(k, c')$, where m' is a valid message or an indication of error (i.e. $m' = \perp$) signaling possible interference from Eve.

Since we are focusing on authentication protocols that do not provide secrecy, we can assume without loss of generality that the encoding takes the convenient form

$$f(k, m) = (m, h(k, m))$$

where $t = h(k, m)$ is called an authentication tag. Indeed, any encoding $f(k, m)$ that does not aim to provide secrecy can be transformed to $f'(k, m) = (m, f(k, m))$.

Finally, we make one more simplifying assumption: we consider protocols such that if a message-tag pair (m, t) was generated by Alice and honestly transmitted by Eve, then Bob accepts with certainty. This assumption makes easier certain technical details in the definitions and proofs.

3.2 Impersonation and Substitution Attacks

The adversary Eve, who does not know the secret key k, may try an impersonation attack or a substitution attack on this scheme.

In an impersonation attack, Eve generates a message-tag pair (m', t') and submits it to Bob. The attack succeeds if Bob accepts the received (m', t') as coming from Alice. The probability that an impersonation attack succeeds is at most

$$p_{imp} = \max_{m', t'} \ \mathbb{P}(t' = h(K, m'))$$

where K is a random variable that models Eve's uncertainty about the key; for example, K may be a uniform random variable over the set of l-bit strings $\{0,1\}^l$.

In a substitution attack, Eve waits until she observes a valid message-tag pair (m,t) from Alice, and then substitutes a pair (m',t') with $m' \neq m$. The attack succeeds if Bob accepts (m',t'). Thus, the probability that a substitution attack succeeds is at most

$$p_{sub} = \max_{(m,t),(m',t'),m\neq m'} \mathbb{P}(t' = h(K,m')|t = h(K,m))$$

where again K is a random variable that models Eve's uncertainty about the key.

3.3 From Impersonation and Substitution Attacks to Composable Security

Now suppose that an authentication system provides low p_{imp} and p_{sub}. We will show that such a system constructs an ideal authenticated channel in the sense of Constructive Cryptography, as was observed in [19]. This motivates our focus on the two types of attack, because protection against impersonation and substitution attacks implies composable, information theoretic security.

Before we proceed, we need to make precise the resources and converters we are considering. First, we look at the goal: the ideal authenticated channel that Alice and Bob want to achieve. The resource $\mathcal{A}$ can be defined by the pseudo-code:

0. Wait for an input $b \in \{0,1\}$ from Eve that encodes whether Alice's message is to be transmitted to Bob or blocked.

1. On input m from Alice, if $b = 1$ output m to Bob and Eve, and if $b = 0$ then output m to Eve only, and allow Eve to specify whether Alice or Bob or both get a notification of the error. If b has not yet been specified, then output m to Eve and wait.

Thus, we have a channel that provides to Bob the guarantee: if anything other than $\perp$ is output at his end, then that message must come from Alice. The filter $\flat$ for this channel emulates honest behavior on Eve's interface: it inputs $b = 1$ to allow Alice's message to go through.

Next, we consider the real resources that Alice and Bob have available. The first resource is a secret key resource $\mathcal{K}$:

0. Wait for an input $b \in \{0,1\}$ from Eve that encodes whether Alice and Bob are blocked from obtaining a secret key.

1. If $b = 0$ then allow Eve to specify whether a notification of the error is sent to Alice or Bob or both, or whether nothing appears on their ends. If $b = 1$ then let K be a random variable with the specified distribution (usually uniform over the set $\{0,1\}^l$). Draw $K = k$ and output k to Alice and Bob.

The filter $\diamond$ for this resource always inputs $b = 1$.

The second real resource that Alice and Bob have available is the insecure channel $\mathcal{C}$:

1. On input m from Alice, output m to Eve.
2. On input m' from Eve, output m' to Bob.

Thus, we have a channel that is completely under the control of Eve. The filter $\sharp$ for this channel models honest behavior by forwarding Alice's messages to Bob and by not allowing Eve to input anything.

Next, we consider the converters that Alice and Bob use to construct the ideal resource from the real resource. Both converters have two inside interfaces, connecting to the secret key and the insecure channel resources, and one outside interface, interacting with the user. The converter f for Alice can be described by the pseudo-code:

1. On input k at the first inside interface, and input m at the outside interface, output $f(k, m) = (m, h(k, m))$ at the second inside interface.

Similarly, the converter ϕ for Bob can be described by the pseudo-code:

1. On input k at the first inside interface and (m', t') at the second inside interface, if $t' = h(k, m')$ output m' on the outside interface, else output $\perp$ on the outside interface.

Now, we are ready to formally state and prove that if an authentication system (f, ϕ) provides low probabilities of impersonation and substitution attacks, then it constructs the ideal authenticated channel from the real insecure channel and a secret key.

Theorem 2. *Suppose that the authentication system f, ϕ provides maximum probabilities of impersonation and substitution attacks p_{imp} and p_{sub} respectively. Then*

$$d(f_A \phi_B(\diamond_E \mathcal{K} \| \sharp_E \mathcal{C}), \flat_E \mathcal{A}) = 0$$

and

$$\exists \sigma \; d(f_A \phi_B(\mathcal{K} \| \mathcal{C}), \sigma_E \mathcal{A}) = \max(p_{imp}, p_{sub})$$

Corollary 1. *If $p_{imp}, p_{sub} < \epsilon$, then $\mathcal{K} \| \mathcal{C} \xrightarrow{(f,\phi),\epsilon} \mathcal{A}$.*

Proof. Before we prove this theorem, it is helpful to consider the general problem of determining the distance between two resources. There is a general principle in cryptography: if two interacting systems behave identically unless some event E occurs, then the distinguishing advantage between them is at most $\mathbb{P}(E)$. This idea has been formalized for example in [13, 2, 20, 15].

Now we can return to the proof of the theorem. First, observe that in the case with Eve blocked, both the real resource $f_A \phi_B(\diamond_E \mathcal{K} \| \sharp_E \mathcal{C})$ and the ideal

resource $\flat_E \mathcal{A}$ have Eve's interface blocked and transmit a message from Alice to Bob. Therefore,

$$d(f_A \phi_B(\diamond_E \mathcal{K} \| \sharp_E \mathcal{C}), \flat_E \mathcal{A}) = 0$$

Now, we consider the case with full access for Eve. First, we have to find a suitable simulator σ.

The real resource $f_A \phi_B(\mathcal{K} \| \mathcal{C})$ provides Eve with a control that can block the secret key (and therefore the whole resource). Thus, the simulator sigma has to also provide such a control to Eve, and if Eve sets it to 0, the simulator sets the blocking control of the authenticated channel to 0.

The real resource $f_A \phi_B(\mathcal{K} \| \mathcal{C})$ outputs a codeword $f(k, m)$ on Eve's interface, while the ideal resource $\mathcal{A}$ outputs Alice's message m directly. Thus, we want the simulator σ to convert a message m to a codeword; the simulator can internally draw a secret key k from the appropriate distribution and then compute $f(k, m)$.

The real resource $f_A \phi_B(\mathcal{K} \| \mathcal{C})$ allows Eve to enter messages that go to Bob. Thus, the simulator σ has to also accept messages from Eve. If Eve inputs a codeword $f(k, m)$ that the simulator previously output to her, then the simulator allows Alice's message to go to Bob on the authenticated channel. If Eve inputs a different codeword, then the simulator triggers an error for Bob on the authenticated channel. In case Eve inputs a codeword before any input from Alice, the simulator triggers an error for Bob on the authenticated channel, and, on a subsequent input from Alice, draws the secret key from the conditional distribution on keys given the event that Eve's codeword is not valid.

With the simulator σ just described, we consider $d(f_A \phi_B(\mathcal{K} \| \mathcal{C}), \sigma_E \mathcal{A})$. A distinguisher has access to all interfaces of one resource or the other, and is trying to tell them apart. By inspection, we see that the distinguisher can observe a difference in behavior only if he tries an impersonation or a substitution attack and succeeds, in which case the distinguisher can be sure that he was interacting with the real system. Then, we obtain,

$$d(f_A \phi_B(\mathcal{K} \| \mathcal{C}), \sigma_E \mathcal{A}) = \max(p_{sub}, p_{imp})$$

as needed. □

3.4 Authentication by a Random Function

We have shown that if an authentication system provides low probabilities of impersonation and substitution attacks, then it provides composable, information theoretic security. Now we ask: what kinds of encoding rules can be used to achieve these low probabilities of impersonation and substitution attacks? We begin with the motivating example of an authentication tag from a random function.

Suppose Alice and Bob know a random function $G = g$ from $\mathbb{M}$ to $\mathbb{T}$. Then, Alice and Bob can use g to authenticate messages in $\mathbb{M}$ using tags in $\mathbb{T}$. To send message m, Alice computes $t = g(m)$ and sends (m, t). To verify a received message (m, t), Bob checks whether $t = g(m)$.

The adversary Eve may try an impersonation attack or a substitution attack on this scheme. The probability that an impersonation attack (m', t') succeeds is

$$\mathbb{P}(G(m') = t') = \frac{1}{|\mathbb{T}|}$$

The probability that a substitution attack from (m, t) to (m', t') succeeds is

$$\mathbb{P}(G(m') = t' | G(m) = t) = \frac{1}{|\mathbb{T}|}$$

Thus, random functions provide an authentication scheme that gives the smallest possible probabilities for impersonation and substitution attacks given the size of the tag space $\mathbb{T}$. The problem with this scheme is that $|\mathbb{M}| \log |\mathbb{T}|$ bits of secret key are required to specify a random function from $\mathbb{M}$ to $\mathbb{T}$, and this amount is too large to be practical.

3.5 Almost Strongly Universal$_2$ Classes of Hash Functions

To overcome the problem with authentication by random functions, δ-almost strongly universal$_2$ classes of functions are used, which mimic the probabilities of successful impersonation and substitution attacks given by random functions but require much less bits of secret key.

Definition 4. *Let $\mathbb{M}, \mathbb{T}$ be finite sets, and let $\mathbb{G}$ be a class of functions from $\mathbb{M}$ to $\mathbb{T}$. $\mathbb{G}$ is called δ-almost strongly universal$_2$ if for G a uniform random variable taking values in $\mathbb{G}$,*

1. *For all $m \in \mathbb{M}$ and for all $t \in \mathbb{T}$,*

$$\mathbb{P}(G(m) = t) = \frac{1}{|\mathbb{T}|}$$

2. *For all $m_1 \neq m_2 \in \mathbb{M}$, and for all $t_1, t_2 \in \mathbb{T}$,*

$$\mathbb{P}(G(m_2) = t_2 | G(m_1) = t_1) \leq \delta$$

It is convenient in the context of authentication to think of the class of functions $\mathbb{G}$ as being indexed by a key k that takes values in some finite set $\mathbb{K}$. Thus, there is a function

$$h : \mathbb{K} \times \mathbb{M} \to \mathbb{T}$$

and the class $\mathbb{G}$ is given by

$$\mathbb{G} = \{h(k, \cdot) : k \in \mathbb{K}\}$$

Similarly, the random variable G in the definition takes the form

$$G = h(K, \cdot)$$

where K is a uniform random variable on the set of keys $\mathbb{K}$.

We see from the definitions that there is a bijective correspondence between almost strongly universal$_2$ classes of hash functions and authentication protocols with encoding rule of the form $f(k,m) = (m, h(k,m))$ and with probability of impersonation attack bounded by $1/|\mathbb{T}|$ and probability of substitution attack bounded by δ. For each almost strongly universal$_2$ class, the function $h(k,m)$ can be used to compute the tag in an authentication protocol, and, conversely, for each good authentication protocol, the function $h(k,m)$ that is used to compute the tag defines an almost strongly universal$_2$ class.

3.6 A Probabilistic Construction of a Good One Way Authentication Protocol

The promise of δ-almost strongly universal$_2$ classes is that much less than $|\mathbb{M}| \log |\mathbb{T}|$ bits of key are required to specify a function in the class. Here, we show using a probabilistic construction that there are good one way authentication protocols which require only $O(\log\log|\mathbb{M}| + \log|\mathbb{T}|)$ bits of secret key. The technique was presented in [9] where it was attributed to R. Roth.[1]

Theorem 3. *Let $\{0,1\}^n$ be the desired space of messages, and let 2^{-r} be the desired security level. Then, there exists an authentication scheme for n bit messages with $r+1$ bit tags and*

$$l = \lceil \log(n+r+2) + 2r + \log(96\ln(2)) \rceil$$

bit secret key such that the probabilities of impersonation and substitution attacks are bounded by 2^{-r}.

Proof. Let $h = h(k,m)$ be the function that assigns an authentication tag to a given secret key k and message m. We think of h as a matrix. The rows are indexed by secret keys; each row contains the authentication tags for all messages given that key. The columns are indexed by messages; each column contains the authentication tags for the given message under all secret keys.

Now, we choose a random authentication tag matrix H; each entry of H is chosen independently from the uniform distribution on $\{0,1\}^{r+1}$. We introduce random variables that count the number of keys such that certain messages get certain authentication tags:

- For $m \in \{0,1\}^n$ and $t \in \{0,1\}^{r+1}$, let

$$N_{mt} = N_{mt}(H) = |\{k : t = H(k,m)\}|$$

- For $m_1 \neq m_2 \in \{0,1\}^n$ and $t_1, t_2 \in \{0,1\}^{r+1}$, let

$$N_{m_1 t_1 m_2 t_2} = N_{m_1 t_1 m_2 t_2}(H) = |\{k : t_1 = H(k,m_1) \wedge t_2 = H(k,m_2)\}|$$

[1] The proof only appears in the full version of [9].

Then, N_{mt} has a $Binomial(2^l, 2^{-r-1})$ distribution: each of the 2^l entries in the m-th column of H is an independent trial that succeeds with probability 2^{-r-1}. Similarly, $N_{m_1t_1m_2t_2}$ has a $Binomial(2^l, 2^{-2r-2})$ distribution.

The proof that a random H makes a good authentication code relies on the fact that $N_{mt}, N_{m_1t_1m_2t_2}$ are tightly concentrated around their mean, which we formalize using the Chernoff Bound [11]:

Theorem 4 (Chernoff Bound)**.** *Let $X_1, \ldots X_n$ be independent Bernoulli random variables with parameters $p_1, \ldots, p_n$ respectively; let $X = \sum_i X_i$ and $\mu = \mathbb{E}X = \sum_i p_i$. Then*

1. *(Upper Tail) $\forall \delta > 0, \mathbb{P}(X \geq (1+\delta)\mu) \leq e^{-\frac{\delta^2}{2+\delta}\mu}$.*
2. *(Lower Tail) $\forall \delta \in (0,1), \mathbb{P}(X \leq (1-\delta)\mu) \leq e^{-\mu\delta^2/2}$.*
3. *(Two-sided) $\forall \delta \in (0,1), \mathbb{P}(|X-\mu| \geq \delta\mu) \leq 2e^{-\mu\delta^2/3}$.*

We obtain:

$$\mathbb{P}(|N_{mt} - 2^{l-r-1}| \geq 2^{l-r-3}) \leq 2e^{-2^{l-r-1}/48}$$
$$\mathbb{P}(|N_{m_1t_1m_2t_2} - 2^{l-2r-2}| \geq 2^{l-2r-3}) \leq 2e^{-2^{l-2r-2}/12}$$

Then, we use the union bound to show that the probability that *any* of the random variables deviates significantly from its mean is less than one:

$$\mathbb{P}\Big((\bigcup_{mt}\{|N_{mt}-2^{l-r-1}| \geq 2^{l-r-3}\}) \cup (\bigcup_{m_1t_1m_2t_2}\{|N_{m_1t_1m_2t_2}-2^{l-2r-2}| \geq 2^{l-2r-3}\})\Big)$$
$$\leq 2^n 2^{r+1} 2e^{-2^{l-r-1}/48} + \frac{2^n(2^n-1)}{2} 2^{2r+2} 2e^{-2^{l-2r-2}/12} < 1$$

for the given choice of secret key size $l = \lceil \log(n+r+2) + 2r + \log(96\ln(2)) \rceil$.

Then, there exists a particular choice $H = h$ of the matrix of authentication tags such that

$$\forall m \forall t, |N_{mt}(h) - 2^{l-r-1}| < 2^{l-r-3}$$
$$\forall (m_1 \neq m_2) \forall t_1 \forall t_2, |N_{m_1t_1m_2t_2}(h) - 2^{l-2r-2}| < 2^{l-2r-3}$$

It remains to show that this authentication tag matrix gives probabilities of impersonation and substitution attacks bounded by 2^{-r}. For a secret key chosen from the uniform distribution, the maximum probability of impersonation attack is

$$\max_{m,t} \frac{N_{mt}}{2^l} < \frac{2^{l-r-1} + 2^{l-r-3}}{2^l} < 2^{-r}$$

and the maximum probability of a substitution attack is

$$\max_{m_1t_1m_2t_2} \frac{N_{m_1t_1m_2t_2}}{N_{m_1t_1}} < \frac{2^{l-2r-2} + 2^{l-2r-3}}{2^{l-r-1} - 2^{l-r-3}} = 2^{-r}$$

as needed. □

Remark: The probabilistic construction above does not, strictly speaking, give a 2^{-r}-almost strongly universal$_2$ class; this is because the probability of impersonation attack is only guaranteed to be at most 2^{-r}, rather than 2^{-r-1} as would be required according to the standard definition (Definition 4). This need not worry us. First, the probabilistic construction above captures the essential intuition behind almost strongly universal$_2$ classes, even if it allows a slight deviation in the probability of impersonation attack. Second, we will see in subsection 3.9 that using a similar argument but going through almost XOR universal functions instead of directly, we can get an almost strongly universal$_2$ class that satisfies Definition 4 and has a slightly lower secret key size

$$l = \lceil \log(n + \frac{r}{2}) + 2r + \log(24\ln(2)) \rceil$$

3.7 A Lower Bound on the Secret Key Size

In order to establish the fundamental resource requirements for one way authentication protocols, we need to show that the secret key size achieved by the probabilistic construction in subsection 3.6 is essentially optimal. Now, we present the technique for proving a lower bound that matches the upper bound asymptotically within a constant factor. This technique also comes from [9].

Theorem 5. *If there exists a one way authentication protocol with message space $\{0,1\}^n$, upper bound 2^{-r} on the probabilities of impersonation and substitution attacks and secret key size l bits then*

$$l + \log(l) \geq \log(n + r) + r - 2$$
$$l \geq 2r$$

Proof. The second inequality, $l \geq 2r$, holds even for interactive protocols; we will prove a much more general result in Theorem 12 in Section 4, from which $l \geq 2r$ in the present setting will follow as a special case. Now, we focus on proving the first inequality.

First, we give the high level idea of the proof. We look at the posterior distribution on secret keys conditional on observing a given message-tag pair. The posterior distributions for all message-tag pairs live in the probability simplex on 2^l elements. The requirement that impersonation and substitution attacks are difficult implies that the different posterior distributions are far apart. The ability to pack many points that are far apart in the probability simplex on 2^l elements implies a lower bound on l.

Now, we proceed with the details. First, we introduce notation for the posterior distributions and their support. Given a message-tag pair m, t that occurs with positive probability, let

$$p_{mt}(k) = \mathbb{P}(K = k | M = m, T = t)$$

where K, M, T are random variables denoting the secret key, and an honestly generated message and tag. Also, let

$$S_{mt} = \{k : h(k, m) = t\}$$

be the support of the distribution p_{mt}.

Next, we formalize the statement that the distributions p_{mt} are far apart. First, consider a given m and two distinct tags t, t'. Then, $S_{mt} \cap S_{mt'} = \emptyset$, so

$$\frac{1}{2}\|p_{mt} - p_{mt'}\|_1 = 1$$

Second, consider two distinct messages m, m' and two (not necessarily distinct) tags t, t'. Then,

$$\mathbb{P}(K \in S_{m't'} | M = m, T = t)$$

is at most the probability that a substitution attack that observes m, t and substitutes m', t' is successful.[2] Thus, we obtain

$$\sum_{k \in S_{m't'}} p_{mt}(k) = p_{mt}(S_{m't'}) = \mathbb{P}(K \in S_{m't'} | M = m, T = t) \leq 2^{-r}$$

and from here we get

$$\begin{aligned}\frac{1}{2}\|p_{mt} - p_{m't'}\|_1 = \sup_{S \subset \{0,1\}^l} &(p_{m't'}(S) - p_{mt}(S)) \\ &\geq p_{m't'}(S_{m't'}) - p_{mt}(S_{m't'}) \geq 1 - 2^{-r}\end{aligned}$$

It remains to show that the ability to pack many different p_{mt} that are far apart inside the probability simplex on 2^l elements implies a lower bound on l. First, we count the number of different distributions that we have. There are 2^n messages, and, for each message, there must be at least 2^r possible authentication tags, because the probability of impersonation attack is assumed to be bounded by 2^{-r}. Thus, we have at least 2^{n+r} distributions.

Next, we outline the remaining steps of the argument. First, we round the distributions p_{mt} to distributions $\tilde{p}_{mt}$ that have all entries of the form $i2^{-l}$, with i an integer; we also show that this operation does not change the distances by much. Next, we convert the distributions $\tilde{p}_{mt}$ into codewords c_{mt} of an error correcting code, by interpreting the $\tilde{p}_{mt}$ as run-length encodings. Since the distributions $\tilde{p}_{mt}$ are far apart, the codewords c_{mt} are also far apart. Finally, we apply the Singleton bound for error-correcting codes to get the lower bound on l.

We proceed with the rounding step. Given p_{mt}, define

$$a_{mt}(k) = \lfloor 2^{l+1} p_{mt}(k) \rfloor$$

Then,

$$\sum_k a_{mt}(k) > \sum_k (2^{l+1} p_{mt}(k) - 1) = 2^{l+1} - 2^l = 2^l$$

[2] Here we use the assumption: if the secret key is such that m gets tag t, then the recipient accepts m, t with certainty. Thus, the operational interpretation of S_{mt} is the set of keys such that the recipient accepts m, t.

Then, we can decrease some of the positive $a_{mt}(k)$ by integer steps and obtain $\tilde{a}_{mt}(k)$ such that

$$\sum_k \tilde{a}_{mt}(k) = 2^l$$

Finally, we define

$$\tilde{p}_{mt}(k) = \tilde{a}_{mt}(k)2^{-l}$$

Note that we have the guarantee $\tilde{p}_{mt}(k) \leq 2p_{mt}(k)$ for our construction. Also, if we define $\tilde{S}_{mt}$ to be the support of $\tilde{p}_{mt}$, we have the guarantee that $\tilde{S}_{mt} \subset S_{mt}$.

To complete the rounding step, we obtain a lower bound on the distance between two distinct distributions $\tilde{p}_{mt}$ and $\tilde{p}_{m't'}$. If $m = m' \wedge t \neq t'$, then the supports $\tilde{S}_{mt}$ and $\tilde{S}_{m't'}$ are disjoint, so

$$\frac{1}{2}\|\tilde{p}_{mt} - \tilde{p}_{m't'}\|_1 = 1$$

If $m \neq m'$ then

$$\begin{aligned}\frac{1}{2}\|\tilde{p}_{mt} - \tilde{p}_{m't'}\|_1 &= \sup_{S\subset\{0,1\}^l}(\tilde{p}_{m't'}(S) - \tilde{p}_{mt}(S)) \\ &\geq \tilde{p}_{m't'}(\tilde{S}_{m't'}) - \tilde{p}_{mt}(\tilde{S}_{m't'}) = 1 - \tilde{p}_{mt}(\tilde{S}_{m't'}) \\ &\geq 1 - \tilde{p}_{mt}(S_{m't'}) \\ &\geq 1 - 2p_{mt}(S_{m't'}) \\ &\geq 1 - 2\cdot 2^{-r}\end{aligned}$$

Next, we perform the conversion of the distributions $\tilde{p}_{mt}$ into codewords c_{mt}. We order all keys lexicographically. We then take the probability vector

$$(\tilde{p}_{mt}(k_1), \ldots \tilde{p}_{mt}(k_{2^l})) = \frac{1}{2^l}(\tilde{a}_{mt}(k_1), \ldots \tilde{a}_{mt}(k_{2^l}))$$

and convert it to the codeword

$$c_{mt} = \underbrace{\alpha_1 \ldots \alpha_1}_{\tilde{a}_{mt}(k_1)}\underbrace{\alpha_2 \ldots \alpha_2}_{\tilde{a}_{mt}(k_2)} \ldots \underbrace{\alpha_{2^l} \ldots \alpha_{2^l}}_{\tilde{a}_{mt}(k_{2^l})}$$

where $\alpha_1, \ldots \alpha_{2^l}$ are the symbols of an alphabet of size 2^l.

To complete the conversion into codewords, we need to relate the hamming distance of the codewords to the l_1 distance of the probability distributions. Indeed, we have

$$\frac{d_H(c_{mt}, c_{m't'})}{2^l} \geq \frac{1}{2}\|\tilde{p}_{mt} - \tilde{p}_{m't'}\|_1$$

where d_H denotes the Hamming distance. This is because the i-th key contributes

$$|\tilde{p}_{mt}(k_i) - \tilde{p}_{m't'}(k_i)| = \frac{|\tilde{a}_{mt}(k_i) - \tilde{a}_{m't'}(k_i)|}{2^l}$$

to the l_1 distance $\|\tilde{p}_{mt} - \tilde{p}_{m't'}\|_1$ and contributes at least

$$|\tilde{a}_{mt}(k_i) - \tilde{a}_{m't'}(k_i)|$$

to $2d_H(c_{mt}, c_{m't'})$.

The last step of the proof is to apply the Singleton bound for error correcting codes:

Theorem 6 (Singleton Bound)**.** *Let C be a collection of codewords of length s and minimum distance d over alphabet of size q. Then $|C| \leq q^{s-d+1}$.*

Proof of Singleton Bound. Erase the first $d-1$ positions of each codeword. We obtain $|C|$ distinct codewords of length $s - d + 1$. □

Applying the Singleton Bound to the codewords c_{mt} we obtain

$$2^{n+r} \leq (2^l)^{2^l - 2^l(1-2^{-r+1})+1}$$

which can be simplified to

$$l + \log(l) \geq \log(n + r) + r - 2$$

as needed. □

3.8 Almost XOR Universal Classes of Hash Functions

So far, we have focused on one way authentication protocols and almost strongly universal$_2$ classes. Now, we present a closely related notion: almost XOR universal classes of hash functions.

Definition 5. *Let $\epsilon > 0$, and consider the function*

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^r$$

The associated keyed class of functions from $\{0,1\}^n$ to $\{0,1\}^r$

$$\{g(k, \cdot) : k \in \{0,1\}^l\}$$

is called ϵ-almost XOR universal if for all $m \neq m' \in \{0,1\}^n$ and for all $t \in \{0,1\}^r$,

$$\mathbb{P}(g(K, m) + g(K, m') = t) \leq \epsilon$$

where K is a uniform random variable over $\{0,1\}^l$ and where $+$ is bitwise addition mod 2.

Thus, almost XOR universal classes mimic another property of random functions, namely that for a random function F, $\mathbb{P}(F(m) + F(m') = t) = 2^{-r}$.

An almost XOR universal class can be used to build an almost strongly universal$_2$ class as shown in the following proposition.

Proposition 2. *Let*

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^r$$

define an ϵ-almost XOR universal class. Then,

$$h : (\{0,1\}^l \times \{0,1\}^r) \times \{0,1\}^n \to \{0,1\}^r$$

given by

$$h(k, k', m) = g(k, m) + k'$$

defines an ϵ-almost strongly universal$_2$ class.

Proof. First, take $m \in \{0,1\}^n$ and $t \in \{0,1\}^r$. Then,

$$\mathbb{P}(h(K, K', m) = t) = \mathbb{P}(g(K, m) + K' = t) = \frac{1}{2^r}$$

because K' is independent of K and has the uniform distribution over $\{0,1\}^r$. Now, take $m \neq m' \in \{0,1\}^n$ and $t, t' \in \{0,1\}^r$. The event

$$E = \{g(K, m) + K' = t \ \ \wedge \ \ g(K, m') + K' = t'\}$$

is a subset of the event

$$F = \{g(K, m) + g(K, m') = t + t'\}$$

Moreover, for each value $K = k$ such that F occurs, there is exactly one value k' such that E also occurs. Then,

$$\mathbb{P}(E) = \mathbb{P}(F)\mathbb{P}(E|F) \leq \epsilon \frac{1}{2^r}$$

Then,

$$\mathbb{P}(h(K, K', m') = t' | h(K, K', m) = t) = \frac{\mathbb{P}(E)}{\mathbb{P}(h(K, K', m) = t)} \leq \epsilon$$

as needed. □

3.9 Lower and Upper Bounds on the Secret Key Size for Almost XOR Universal Classes

Since almost XOR universal and almost strongly universal$_2$ classes are closely related, similar techniques can be used to prove lower and upper bounds on the key size required for almost XOR universal classes. For the upper bound, we have the following:

Theorem 7. *Let $\{0,1\}^n$ be the space of messages, let 2^{-r} be the security level, and let $\{0,1\}^{r+s}$ be the output space. Let the key size be*

$$l = \lceil \log(n + \frac{r + s - 1}{2}) + r + \log(12 \ln(2)) \rceil$$

Then, there exists a function

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^{r+s}$$

defining a 2^{-r}-almost XOR universal class.

Proof. We choose a random matrix $G(k,m)$, and show that it defines a 2^{-r}-almost XOR universal class with non-zero probability.

We choose each entry of G independently from the uniform distribution on $\{0,1\}^{r+s}$. Given $m \neq m' \in \{0,1\}^n$ and $t \in \{0,1\}^{r+s}$, we define a random variable $N_{mm't} = N_{mm't}(G)$ that counts the number of keys k for which $G(k,m) + G(k,m') = t$ holds. Then, $N_{mm't}$ has a $Binomial(2^l, 2^{-r-s})$ distribution.

From the Chernoff Bound (Theorem 4) we get:

$$\mathbb{P}(N_{mm't} \geq 2^{l-r}) \leq e^{-\frac{(2^s-1)^2}{2^s+1}2^{l-r-s}}$$

From the union bound, we get

$$\mathbb{P}\Big(\bigcup_{mm't}\{N_{mm't} \geq 2^{l-r}\}\Big) \leq \binom{2^n}{2} 2^{r+s} e^{-\frac{(2^s-1)^2}{2^s+1}2^{l-r-s}} < 1$$

for the given choice of l. Then, there exists a particular choice $G = g$ for which all counts $N_{mm't}(g)$ are less than 2^{l-r}. This g defines a 2^{-r}-almost XOR universal class, as needed. □

From Theorem 7 and Proposition 2 we obtain the following Theorem, which improves slightly on the earlier construction of Theorem 3.

Theorem 8. *Let $\{0,1\}^n$ be the space of messages, let 2^{-r} be the security level, and let $\{0,1\}^{r+s}$ be the space of authentication tags. Let*

$$l = \lceil \log(n + \frac{r+s-1}{2}) + 2r + s + \log(12\ln(2)) \rceil$$

be the secret key size. Then, there exists a function

$$h : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^{r+s}$$

that defines a 2^{-r}-almost strongly universal$_2$ class.

Proof. Let

$$l' = \lceil \log(n + \frac{r+s-1}{2}) + r + \log(12\ln(2)) \rceil$$

and let

$$g : \{0,1\}^{l'} \times \{0,1\}^n \to \{0,1\}^{r+1}$$

be the function given by Theorem 7. Take

$$h : (\{0,1\}^{l'} \times \{0,1\}^{r+s}) \times \{0,1\}^n \to \{0,1\}^{r+s}$$

given by

$$h(k, k', m) = g(k, m) + k'$$

and apply Proposition 2. Then, h defines a 2^{-r}-almost strongly universal$_2$ class with key size $l' + r + s = l$. □

Next, we give a lower bound on the key size for an almost XOR universal class. The lower bound follows from the lower bound on key size for one way authentication protocols (Theorem 5), and the observation that an almost XOR universal class can be used to construct a one way authentication protocol.

Theorem 9. *Let*

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^{r+s}$$

define a 2^{-r}-almost XOR universal class. Then,

$$l + r + s + \log(l + r + s) \geq \log(n + r) + r - 2$$
$$l + r + s \geq 2r$$

Proof. From g, construct

$$h : \{0,1\}^{l+r+s} \times \{0,1\}^n \to \{0,1\}^{r+s}$$

as in Proposition 2. Then, h defines a 2^{-r} almost strongly universal$_2$ class and therefore also a one way authentication protocol with probability of impersonation attack bounded by 2^{-r-s} and probability of substitution attack bounded by 2^{-r}. Apply Theorem 5 to get the two inequalities for the key size $l + r + s$ of this class. □

3.10 Explicit Constructions of Almost XOR Universal and Almost Universal$_2$ Classes

So far, we have presented probabilistic proofs that almost XOR universal and almost strongly universal$_2$ classes with given parameters exist. Now we give an explicit construction of such classes.

We present a construction from [18] that is based on polynomials over finite fields. Let $GF(2^l)$ be the field with 2^l elements. Given a message $m \in \{0,1\}^n$, break up m into blocks of l bits and interpret it as the coefficients $m_1, \ldots m_d$ of a polynomial of degree $d = \lceil n/l \rceil$ over $GF(2^l)$. Then, the associated keyed family of hash functions is

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^l$$
$$g(k, m) = \sum_{i=1}^{d} k^i m_i$$

Given $m_1 \neq m_2$ and t, we have

$$\mathbb{P}(g(K, m_1) + g(K, m_2) = t) = \mathbb{P}(\sum_{i=1}^{d} (m_{1i} + m_{2i}) K^i - t = 0) \leq \frac{d}{2^l}$$

because a polynomial of degree at most d can have at most d roots. Therefore, g is a $\lceil n/l \rceil 2^{-l}$-almost XOR universal keyed hash function family. We can see that this construction has parameters comparable to the probabilistic construction of Theorem 7. Indeed, if we set

$$2^{-r} = \lceil \frac{n}{l} \rceil 2^{-l}$$

as the security level obtained by this construction, we obtain

$$\log(n) + r \leq l + \log(l) \leq \log(n + l) + r$$

As in Proposition 2, we can convert an almost XOR universal to an almost strongly universal$_2$ class. We define

$$h : \{0,1\}^{2l} \times \{0,1\}^n \to \{0,1\}^l$$

by

$$h(k, k', m) = \sum_{i=1}^{\lceil n/l \rceil} m_i k^i + k'$$

When $l \approx \log(n) + r$, this almost strongly universal$_2$ class has secret key size comparable to the lower bound given by Theorem 5.

3.11 References for One Way Authentication

The problem of message authentication was introduced in [10]. The related topic of universal hashing was introduced in [4], and the connection between universal hashing and message authentication was made in [25], where the authors also introduce the notion of almost strongly universal$_2$ class of hash functions.

Various constructions and bounds for the one way message authentication problem appear in [10, 25, 21, 22, 23, 9, 24, 17]. We have chosen to focus the exposition on the techniques presented in [9] because the construction and the lower bound there match asymptotically within a constant factor, and thus give asymptotically the right answer to the question of how many bits of secret key are needed for one way authentication.

The composable security of authentication by universal hashing was shown in [19].

4 Interactive Authentication

We begin this section with a brief motivation for the study of interactive authentication. We saw in Theorem 5 that there are two lower bounds on the secret key length l for a one way authentication protocol. The first lower bound,

$$l + \log(l) \geq \log(n + r) + r - 2$$

depends on both the message length n and the security level 2^{-r}. It comes from what can be called a "sphere packing" proof due to [9], and we have shown that it is asymptotically tight. The second bound (which we did not prove there but will prove in this section) states that

$$l \geq 2r$$

and depends only on the security level. It comes from an "information theoretic" proof that has gradually evolved over the years; see for example [10, Theorem 1], [22, Equation 28 and subsequent discussion], [17, Theorem 7 for the case $n = 1$].

The bound $l \geq 2r$ cannot be tight for one way protocols as it does not depend on the message length. On the other hand, the information theoretic proof looks entirely natural and reasonable. So why does it not give the right lower bound? The short answer is that the information theoretic proof can be extended to apply to interactive protocols as well, and there exist interactive authentication protocols that achieve $l \approx 2r$. Thus, the information theoretic proof gives the right lower bound, but for the larger class of interactive protocols.

Now, we give an outline of the rest of this section. We begin by constructing interactive protocols that can use a key of size $l \approx 2r$ to authenticate a message of any length with security level 2^{-r}. In subsection 4.1 we show how to construct an authenticated channel for a large message from an authenticated channel for a smaller message and a bidirectional insecure channel. This step can be repeated a number of times, leading eventually to a protocol that authenticates a message of any length n using a key of size $l \approx 2r$; we formalize this in subsection 4.2.

Having constructed protocols that achieve $l \approx 2r$ for any message size, we turn our attention to proving a matching lower bound. First, we introduce some notation for interactive protocols in subsection 4.3. Next, we consider possible attacks that the adversary Eve can perform. At first sight, it may appear that the more interaction a protocol has, the more options Eve has for manipulating the order of messages in time. However, it turns out that to prove our lower bound, we will need to consider only two natural options: either Eve ignores Alice completely and has a conversation only with Bob, or Eve has an entire conversation with Alice, then uses the information obtained to guess the secret key, and then has a conversation with Bob. These two options are natural generalizations of impersonation and substitution attacks, and we introduce them in detail in subsection 4.4.

The intuition for the lower bound proof is that if a protocol protects against impersonation and substitution attacks, then the joint distribution of the protocol transcript and the secret key in an honest execution and in an attack must be far apart. We formalize this intuition using the relative entropy, an information theoretic quantity that measures how far apart two distributions are. We recall the definition of the relative entropy and an important property called monotonicity in subsection 4.5. Finally, we prove in subsection 4.6 that any protocol with security level ϵ against impersonation and substitution attacks must use secret key of entropy $\approx 2\log(1/\epsilon)$.

4.1 Message size reduction

In this section, we present an idea for constructing an authenticated channel for a larger message from an authenticated channel for a smaller message and a bidirectional insecure channel by using an interactive protocol. The idea first appeared in [9], however, [7] showed that the initial proposal was insecure. Fixes of the original idea later appeared in [8, 18]. Here, we follow the construction of [18].

Theorem 10. *Let $\mathcal{A}^n_{A\to B}$ be an authenticated channel that allows the transmission of an n-bit message from Alice to Bob. Let $\mathcal{C}_{A\leftrightarrow B}$ be a bidirectional insecure channel. Let*

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^r$$

be a keyed ϵ-almost XOR universal hash function family. Then,

$$\mathcal{C}_{A\leftrightarrow B} \| \mathcal{A}^{l+r}_{A\to B} \xrightarrow{\pi,\epsilon} \mathcal{A}^n_{A\to B}$$

where $\pi = (\pi_A, \pi_B)$ is the protocol given by:
$\pi_A =$ "On input m:

1. *Draw random $a \in \{0,1\}^r$ and send (m, a) on the insecure channel.*
2. *Receive b' on the insecure channel.*
3. *Send $(b', g(b', m) + a)$ on the channel $\mathcal{A}^{l+r}_{A\to B}$."*

$\pi_B =$ "

1. *Receive m', a' on the insecure channel.*
2. *Draw random $b \in \{0,1\}^l$ and send it on the insecure channel.*
3. *Receive $(b', g(b', m) + a)$ on the channel $\mathcal{A}^{l+r}_{A\to B}$.*
4. *If $(b', g(b', m) + a) = (b, g(b, m') + a')$ accept m', otherwise reject."*

Proof. First, observe that with filters attached and the adversary blocked, both resources transmit an n-bit message from Alice to Bob. Thus,

$$d(\pi_A\pi_B(\sharp_E\mathcal{C}_{A\leftrightarrow B} \| \flat_E\mathcal{A}^{l+r}_{A\to B}), \flat_E\mathcal{A}^n_{A\to B}) = 0$$

Now, consider the case with full access for Eve. First, we have to define a suitable simulator. The simulator σ contains a copy of the real system $\pi_A\pi_B(\mathcal{C}_{A\leftrightarrow B} \| \mathcal{A}^{l+r}_{A\to B})$. Alice's input message to the simulated real system comes from the authenticated channel $\mathcal{A}^n_{A\to B}$. Eve is allowed to interact with the simulated real system, and σ observes this interaction. If during the interaction Eve only forwards messages honestly, then the simulator releases Alice's message on $\mathcal{A}^n_{A\to B}$. If there is any deviation from honest behavior, then the simulator blocks Alice's message on $\mathcal{A}^n_{A\to B}$; if in addition an error is triggered on either

side of the simulated real system, then the simulator orders $\mathcal{A}^n_{A\to B}$ to output the corresponding error.

Next, we consider the task of distinguishing $\pi_A\pi_B(\mathcal{C}_{A\leftrightarrow B}\|\mathcal{A}^{l+r}_{A\to B})$ and $\sigma_E\mathcal{A}^n_{A\to B}$. By inspection, we see that the distinguisher can tell the real and the ideal resource apart only in the case when the real system accepts a message m' on Bob's side that was not previously input on Alice's side, whereas the ideal system would never output such a message to Bob. Thus, the distance between the real and ideal system is the maximum probability that the real system accepts an incorrect message on Bob's side.

Now, we evaluate this maximum probability. A distinguisher can choose the message m on Alice's interface, the (m', a') received by Bob, and the b' received by Alice. In addition, the distinguisher can choose the order of events in time. Thus, we have to consider cases based on the order of events in time.

Let $T(m, a)$ be the time π_A sends out (m, a), let $T(m', a')$ be the time π_B receives m', a'; immediately afterwards π_B generates the random b and sends it out. Let $T(b')$ be the time that π_A receives b'; immediately afterwards π_A generates $(b', g(b', m) + a)$ and sends it out on the channel $\mathcal{A}^{l+r}_{A\to B}$. Finally, let $T(b', g(b', m) + a)$ be the time that $(b', g(b', m) + a)$ is delivered to π_B on the channel $\mathcal{A}^{l+r}_{A\to B}$.

We assume that π_A, π_B generate an error if events in their view happen out of the expected order. Moreover, from the definition of $\mathcal{A}^{l+r}_{A\to B}$, we know that Alice must first send $(b', g(b', m) + a)$, before it can be delivered to Bob. Thus, we have

$$\begin{aligned} T(m, a) < T(b') < T(b', g(b', m) + a) \\ T(m', a') < T(b', g(b', m) + a) \end{aligned}$$

where the first line is the order imposed by π_A and the second line is the order imposed by π_B.

We see that there are three cases for the order of events: $T(m', a') < T(m, a)$, $T(m, a) < T(m', a') < T(b')$, and $T(b') < T(m', a') < T(b', g(b', m) + a)$. We consider the probability that π_B accepts an incorrect message for each of the cases in turn.

Case 1: $T(m', a') < T(m, a)$ In this case, a is generated independently after π_B determines the pair $(b, g(b, m')+a')$ against which it checks the final incoming message. Thus, the probability that π_B accepts is at most 2^{-r}. This is at most ϵ because g is ϵ-almost XOR universal.[3]

Note that this bound on the probability holds even if $m = m'$. This is as it should be, because $T(m', a') < T(m, a)$ describes a situation in which Eve delivered a message to Bob before Alice intended to send anything, and thus Eve's message should be rejected.

[3] $\forall(m \neq m')\exists t, \mathbb{P}(g(K, m) + g(K, m') = t) \geq 2^{-r}$, and this must be at most ϵ.

Case 2: $T(m,a) < T(m',a') < T(b')$ **and** $m \neq m'$**.** This describes a situation in which the normal time order of events in the protocol is preserved, but the message is modified in transit.

b' is generated after b is known. Moreover, the only way that π_B accepts is if $b' = b$. Thus, we can assume that $b' = b$.

Since $B = b$ is generated after (m,a) and (m',a') are fixed, we have

$$\mathbb{P}(g(B,m) + a = g(B,m') + a') \leq \epsilon$$

because g is ϵ-almost XOR universal.

Case 3: $T(b') < T(m',a') < T(b', g(b',m) + a)$ In this case, the message $(b', g(b',m) + a)$ that Alice sends to Bob on $\mathcal{A}^{l+r}_{A\to B}$ is fixed before b is independently drawn. Thus, the probability that b matches the already fixed b' is 2^{-l}. This is at most ϵ because g is ϵ-almost XOR universal.[4]

We see that in all three cases, the probability that the real system accepts an incorrect message for Bob is bounded by ϵ. Thus,

$$d(\pi_A\pi_B(\mathcal{C}_{A\leftrightarrow B} \| \mathcal{A}^{l+r}_{A\to B}), \sigma_E \mathcal{A}^n_{A\to B}) \leq \epsilon$$

as needed. □

4.2 Interactive authentication with key size independent of the message length

The general composition theorem allows us to repeat the construction step of Theorem 10 several times. We obtain the following corollary:

Corollary 2. *Suppose that for $i = 1, \ldots k$ there is an ϵ_i-almost XOR universal keyed hash function family*

$$g_i : \{0,1\}^{l_i} \times \{0,1\}^{n_i} \to \{0,1\}^{r_i}$$

such that $l_i + r_i = n_{i-1}$ for $i = 2, \ldots k$. Let π_i be the corresponding protocol from Theorem 10. Then,

$$\mathcal{A}^{l_1+r_1} \| \mathcal{C} \xrightarrow{\pi_k \ldots \pi_1, \epsilon_1 + \cdots + \epsilon_k} \mathcal{A}^{n_k}$$

[18] proposes to use the almost XOR universal functions we saw in subsection 3.10 for each of the message size reduction steps. Recall that $GF(2^l)$ denotes the field with 2^l elements, and, given a message $m \in \{0,1\}^n$, we break up m into blocks of l bits and interpret it as the coefficients $m_1, \ldots m_d$ of a polynomial

[4] Let k be a fixed key. Then $\forall (m \neq m'), \mathbb{P}(g(K,m) + g(K,m') = g(k,m) + g(k,m')) \geq 2^{-l}$, and this must be at most ϵ.

of degree $d = \lceil n/l \rceil$ over $GF(2^l)$. Then, the associated keyed family of hash functions is

$$g : \{0,1\}^l \times \{0,1\}^n \to \{0,1\}^l$$
$$g(b,m) = \sum_{i=1}^{d} b^i m_i$$

and g is a $\lceil n/l \rceil 2^{-l}$-almost XOR universal keyed hash function family.

In particular, given a target ϵ, we can use the construction of almost XOR universal families from polynomials over finite fields with the parameters

$$r_1 = l_1 = 3 + \lceil \log \frac{1}{\epsilon} \rceil, \;\; d_1 = 4, \;\; n_1 = d_1 l_1, \;\; \epsilon_1 = d_1 2^{-l_1} \leq \frac{\epsilon}{2}$$

and for $i > 1$ we define recursively

$$r_i = l_i = \frac{n_{i-1}}{2}, \;\; d_i = 2^{l_i - l_{i-1} - 1} d_{i-1}, \;\; n_i = d_i l_i, \;\; \epsilon_i = d_i 2^{-l_i} = d_{i-1} 2^{-l_{i-1}-1} = \frac{\epsilon_{i-1}}{2}$$

We obtain the following:

Corollary 3. *Given n, ϵ, there exists a protocol π such that*

$$\mathcal{C} \| \mathcal{A}_{A \to B}^{2\lceil \log(1/\epsilon) \rceil + 6} \xrightarrow{\pi, \epsilon} \mathcal{A}_{A \to B}^{n}$$

4.3 Notation for Interactive Protocols

Now, we turn attention to proving a matching lower bound on the secret key size needed for message authentication by an interactive protocol.

First, we introduce some notation. Let M be a random variable that models the choice of message. Let $C_1, \ldots C_r$ be random variables that denote the flows of the protocol; thus, in an honest execution, Alice sends $C_1, C_3, \ldots$ and receives $C_2, C_4, \ldots$, and conversely for Bob.[5] Let B be a random variable that denotes Bob's decision to accept or reject; thus $B = 1$ means Bob accepts and $B = 0$ means Bob rejects.

Now we consider the joint distribution of the random variables

$$B, C_1, \ldots C_r, K, M$$

Given the distribution of K, and given a distribution of M, which we are free to specify as long as it is independent from K, an honest execution of the protocol determines the conditional distribution of the other random variables. Let p be the joint probability mass function of

$$B, C_1, \ldots C_r, K, M$$

[5] This model is sufficiently general to capture also protocols that have a variable number of flows: just let r be the maximum number of flows and allow the last few C_i to be empty for some executions. The model also captures protocols in which Alice and/or Bob sends several consecutive flows without response from the other party: these can always be merged into a single C_i.

that is fixed by an honest execution of the protocol. Note that using Bayes' rule, we have

$$p(b, c_1, \ldots c_r, k, m) = p(k)p(m)\Big(\prod_{i=1}^{r} p(c_i|c_1, \ldots c_{i-1}, k, m)\Big)p(b|c_1, \ldots c_r, k, m)$$

Note also that Bob's decisions do not depend directly on m, but only on the key k and the transcript of the protocol so far, i.e. for even i

$$p(c_i|c_1, \ldots c_{i-1}, k, m) = p(c_i|c_1, \ldots c_{i-1}, k)$$

and additionally

$$p(b|c_1, \ldots c_r, k, m) = p(b|c_1, \ldots c_r, k)$$

4.4 Impersonation and Substitution Attacks for Interactive Protocols

Now, we introduce two more joint probability mass functions that are induced by two natural strategies for Eve. First, we consider an impersonation attack, in which Eve has a conversation with Bob:

1. The key k is selected according to the distribution $p(k)$.
2. Eve selects a message m according to the distribution $p(m)$.
3. Eve has a conversation with Bob. For odd i, Eve chooses C_i according to the distribution $p(c_i|c_1, \ldots c_{i-1}, m)$. For even i, Bob chooses C_i according to the distribution $p(c_i|c_1, \ldots c_{i-1}, k)$. We assume that if Bob notices at an early stage that Eve's messages are not consistent with the secret key k, i.e. if
$$p(c_1, \ldots c_{i-1}, k) = 0$$
then he rejects, sets $B = 0$ and stops.
4. Bob computes b according to the distribution $p(b|c_1, \ldots c_r, k)$ and stops.

Let p' denote the joint probability mass function determined by this execution. Then, whenever $p(c_1, \ldots c_r, k) > 0$, we have

$$p'(b, c_1, \ldots c_r, k, m) = p(k)p(m) \prod_{odd\ i} p(c_i|c_1, \ldots c_{i-1}, m) \prod_{even\ i} p(c_i|c_1, \ldots c_{i-1}, k) * p(b|c_1, \ldots c_r, k)$$

Second, we consider a substitution attack, in which Eve has a conversation with Alice and then tries to guess the secret key.

1. The key k is selected according to the distribution $p(k)$.
2. Eve selects the message m for Alice according to $p(m)$.

3. Eve has a conversation with Alice. For i odd, Alice computes c_i according to the distribution $p(c_i|c_1, \ldots c_{i-1}, k, m)$. For i even, Eve computes c_i according to the distribution $p(c_i|c_1, \ldots c_{i-1}, m)$. We assume that if Alice notices Eve's messages are not consistent with the key k, that is, if

$$p(c_1, \ldots c_{i-1}, k, m) = 0$$

for some odd i, then Alice rejects and stops.

4. Eve selects a key $K' = k'$ according to the distribution

$$\mathbb{P}(K' = k') = p(k'|c_1, \ldots c_r, m)$$

and then runs a conversation with Bob using any message of her choice and the key k'.

Let p'' be the joint probability mass function determined by this execution. Then, whenever $p(c_1, \ldots c_r, k, m) > 0$,

$$p''(c_1, \ldots c_r, k, k', m) = p(k)p(m) \prod_{odd\ i} p(c_i|c_1, \ldots c_{i-1}, k, m) \\ * \prod_{even\ i} p(c_i|c_1, \ldots c_{i-1}, m) * p(k'|c_1, \ldots c_r, m)$$

4.5 Relative Entropy

As mentioned in the introduction, the lower bound proof will exploit the fact that the distributions for an honest execution and for an attack are far apart. To quantify this statement, we use the relative entropy. First, recall the definition of relative entropy [5, section 2.3]

Definition 6. *Let $p(x), q(x)$ be two probability mass functions on the same finite set. Then, the relative entropy of p and q is*

$$D(p\|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

where the conventions $0 \log(0/q) = 0$ and $p \log(p/0) = \infty$ are used.

An important property of the relative entropy is monotonicity. Intuitively, it states that further processing can only bring distributions closer together. Formally, we have:

Theorem 11 (Monotonicity of the Relative Entropy)**.** *Let $p(x), q(x)$ be probability mass functions. Let $r(y|x)$ be a set of transition probabilities (i.e. $\forall x \forall y, r(y|x) \geq 0$ and $\forall x, \sum_y r(y|x) = 1$). Let*

$$p'(y) = \sum_x r(y|x)p(x)$$
$$q'(y) = \sum_x r(y|x)q(x)$$

Then,

$$D(p'\|q') \leq D(p\|q)$$

Proof of monotonicity. We use the log-sum inequality [5, Theorem 2.7.1]:

Lemma 1 (Log sum inequality)**.** *Let $a_1, \ldots a_n, b_1, \ldots b_n$ be non-negative. Then,*

$$\big(\sum_i a_i\big) \log \frac{\sum_i a_i}{\sum_i b_i} \leq \sum_i a_i \log \frac{a_i}{b_i}$$

Proof of log sum inequality. Apply Jensen's inequality

$$f(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i f(x_i)$$

to the convex function $f(x) = x\log(x)$, the numbers $x_i = a_i/b_i$ and the weights $\lambda_i = b_i/(\sum_j b_j)$. □

Now we have:

$$\begin{aligned} D(p'\|q') &= \sum_y p'(y) \log \frac{p'(y)}{q'(y)} \\ &= \sum_y \big(\sum_x r(y|x)p(x)\big) \log \frac{\sum_x r(y|x)p(x)}{\sum_x r(y|x)q(x)} \\ &\leq \sum_y \sum_x r(y|x)p(x) \log \frac{r(y|x)p(x)}{r(y|x)q(x)} \\ &= \sum_x p(x) \log \frac{p(x)}{q(x)} = D(p\|q) \end{aligned}$$

as needed. □

Second proof of monotonicity. Think of random variables X, Y with joint probability mass function $p(x, y) = p(x)r(y|x)$. Now consider the following:

$$\begin{aligned} -D(p,q) = \sum_x p(x) \log \frac{q(x)}{p(x)} &= \sum_y p'(y) \sum_x p(x|y) \log \frac{q(x)}{p(x)} \\ &\leq \sum_y p'(y) \log \left(\sum_x p(x|y) \frac{q(x)}{p(x)} \right) \\ = \sum_y p'(y) \log \left(\sum_x \frac{p(y|x)}{p'(y)} q(x) \right) &= \sum_y p'(y) \log \left(\frac{\sum_x r(y|x)q(x)}{p'(y)} \right) \\ &= \sum_y p'(y) \log \frac{q'(y)}{p'(y)} = -D(p', q') \end{aligned}$$

where we have applied Jensen's inequality, then Bayes' rule. This completes the proof. □

4.6 Lower Bound on the Secret Key Entropy for Interactive Protocols

Now we are ready to state and prove the lower bound on secret key entropy for interactive protocols.

Theorem 12. *Let π be any interactive message authentication protocol such that the probability Bob rejects in an honest execution is at most δ and the probabilities of successful impersonation and substitution attacks are bounded by ϵ. Let K be a random variable that models the secret key used by protocol π. Then*

$$H(K) \geq \delta \log \frac{\delta}{1-\epsilon} + (2-\delta) \log \frac{1-\delta}{\epsilon}$$

where $H(\cdot)$ is the Shannon entropy.

Before we prove this Theorem, we remark that in the special case $\delta = 0$, $\epsilon = 2^{-r}$ and π a protocol with a single flow from Alice to Bob, the Theorem gives $H(K) \geq 2r$, and so the secret key length l must be at least $2r$, as we claimed in the second bound of Theorem 5. Now, we proceed to prove Theorem 12.

Proof. First, we give the high level idea of the proof. Intuitively, the key K must contain two parts; the first part protects against impersonation attack and we will show that it must have entropy at least

$$\delta \log \frac{\delta}{1-\epsilon} + (1-\delta) \log \frac{1-\delta}{\epsilon}$$

The second part of the key protects against substitution attack, and we will show that it must have entropy at least

$$\log \frac{1-\delta}{\epsilon}$$

We proceed with the details. First consider the relative entropy between distributions induced by an honest execution and distributions induced by an impersonation attack. For an honest execution, we have that the marginal distribution p_B of B is obtained from the marginal distribution $p_{C_1,\ldots C_r,K,M}$ of the protocol transcript, the secret key and the message using the transition probabilities

$$p(b|c_1,\ldots c_r,k) = p(b|c_1,\ldots c_r,k,m)$$

For an impersonation attack, the marginal distribution p'_B is obtained from the marginal distribution $p'_{C_1,\ldots C_r,K,M}$ using the same transition probabilities $p(b|c_1,\ldots c_r,k,m)$. We apply monotonicity of the relative entropy (Theorem 11) and obtain

$$D(p_B \| p'_B) \leq D(p_{C_1,\ldots C_r,K,M} \| p'_{C_1,\ldots C_r,K,M})$$

We combine this with the further observation that $\mathbb{P}(B = 1) \geq 1 - \delta$ in an honest execution and $\mathbb{P}(B = 1) \leq \epsilon$ in an impersonation attack. We get

$$D\Big((\delta, 1-\delta)\|(1-\epsilon, \epsilon)\Big) \leq D(p_B \| p'_B) \leq D(p_{C_1,\ldots C_r,K,M} \| p'_{C_1,\ldots C_r,K,M}) \quad (1)$$

Now, we turn our attention to substitution attacks. The overall probability that Bob accepts in a substitution attack is at most ϵ. On the other hand, conditional on Eve guessing the correct key, the probability that Bob accepts is at least $1 - \delta$, because Eve can now simulate to Bob an honest execution of the protocol. Therefore,

$$\epsilon \geq (1-\delta)\mathbb{P}(K' = K)$$

which we transform to

$$\begin{aligned}\frac{\epsilon}{1-\delta} \geq \mathbb{P}(K' = K) &= \sum_{c_1,\ldots c_r,k,m:p(c_1,\ldots c_r,k,m)>0} p''(c_1,\ldots c_r,k,k,m) \\ &= \sum_{c_1,\ldots c_r,k,m:p(c_1,\ldots c_r,k,m)>0} p(c_1,\ldots c_r,k,m)\frac{p''(c_1,\ldots,c_r,k,k,m)}{p(c_1,\ldots c_r,k,m)}\end{aligned}$$

and we further transform to

$$\log\frac{1-\delta}{\epsilon} \leq \sum_{c_1,\ldots c_r,k,m:p(c_1,\ldots c_r,k,m)>0} p(c_1,\ldots c_r,k,m)\log\frac{p(c_1,\ldots c_r,k,m)}{p''(c_1,\ldots,c_r,k,k,m)} \quad (2)$$

where we have taken logarithms of both sides, applied Jensen's inequality and concavity of the logarithm, and then flipped the sign.

Now, we combine (1) and (2) and we get

$$\begin{aligned}\delta\log\frac{\delta}{1-\epsilon} + (1-\delta)\log\frac{1-\delta}{\epsilon} + \log\frac{1-\delta}{\epsilon} & \\ \leq \sum_{c_1,\ldots c_r,k,m:p(c_1,\ldots c_r,k,m)>0} & p(c_1,\ldots c_r,k,m) \\ & * \log\frac{p(c_1,\ldots c_r,k,m)^2}{p'(c_1,\ldots c_r,k,m)p''(c_1,\ldots c_r,k,k,m)} \quad (3)\end{aligned}$$

The final step of the proof is to simplify the expression inside the logarithm. We have:

$$\begin{aligned}&\frac{p(c_1,\ldots c_r,k,m)^2}{p'(c_1,\ldots c_r,k,m)p''(c_1,\ldots c_r,k,k,m)} \\ &\quad = \frac{p(k)p(m)\prod_{i=1}^r p(c_i|c_1,\ldots c_{i-1},k,m)}{p(k)p(m)\big(\prod_{i=1}^r p(c_i|c_1,\ldots c_{i-1},m)\big)p(k|c_1,\ldots c_r,m)} = \frac{1}{p(k)}\end{aligned}$$

Combining this with (3) we get

$$\delta\log\frac{\delta}{1-\epsilon} + (2-\delta)\log\frac{1-\delta}{\epsilon} \leq H(K)$$

as needed. □

4.7 References for interactive authentication

The idea of interactive message authentication protocols was introduced in [9]. Unfortunately, the interactive protocol proposed there was insecure, as shown by [7]. Fixes of the original interactive protocol later appeared in [8, 18]. A proof of the lower bound on the secret key size appears in [18]; however, here we have deviated somewhat from the exposition in that paper.

Acknowledgments

This work was supported by the Luxembourg National Research Fund, under CORE project AToMS (Project ID 8293135).

References

[1] Michael Backes, Birgit Pfitzmann, and Michael Waidner. A general composition theorem for secure reactive systems. In *Theory of Cryptography Conference*, pages 336–354. Springer, 2004.

[2] Mihir Bellare and Phil Rogaway. The game-playing technique. *International Association for Cryptographic Research (IACR) ePrint Archive: Report*, 331:2004, 2004.

[3] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 136–145. IEEE, 2001.

[4] J Lawrence Carter and Mark N Wegman. Universal classes of hash functions. In *Proceedings of the ninth annual ACM symposium on Theory of computing*, pages 106–112. ACM, 1977.

[5] Thomas M Cover and Joy A Thomas. *Elements of information theory 2nd edition*. Wiley-interscience, 2006.

[6] Imre Csiszár and Janos Korner. Broadcast channels with confidential messages. *IEEE transactions on information theory*, 24(3):339–348, 1978.

[7] Christian Gehrmann. Cryptanalysis of the gemmell and naor multiround authentication protocol. In *Annual International Cryptology Conference*, pages 121–128. Springer, 1994.

[8] Christian Gehrmann. Multiround unconditionally secure authentication. *Designs, Codes and Cryptography*, 15(1):67–86, 1998.

[9] Pete Gemmell and Moni Naor. Codes for interactive authentication. In *Annual International Cryptology Conference*, pages 355–367. Springer, 1993.

[10] Edgar N Gilbert, F Jessie MacWilliams, and Neil JA Sloane. Codes which detect deception. *Bell Labs Technical Journal*, 53(3):405–424, 1974.

[11] Michel Goemans. Chernoff bounds, and some applications. `http://math.mit.edu/~goemans/18310S15/chernoff-notes.pdf`, 2015.

[12] Robert König, Renato Renner, Andor Bariska, and Ueli Maurer. Small accessible quantum information does not imply security. *Physical Review Letters*, 98(14):140502, 2007.

[13] Ueli Maurer. Indistinguishability of random systems. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 110–132. Springer, 2002.

[14] Ueli Maurer. Constructive cryptography-a new paradigm for security definitions and proofs. *TOSCA*, 6993:33–56, 2011.

[15] Ueli Maurer. Conditional equivalence of random systems and indistinguishability proofs. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pages 3150–3154. IEEE, 2013.

[16] Ueli Maurer and Renato Renner. Abstract cryptography. In *In Innovations in Computer Science*. Citeseer, 2011.

[17] Ueli M Maurer. Authentication theory and hypothesis testing. *IEEE Transactions on Information Theory*, 46(4):1350–1356, 2000.

[18] Moni Naor, Gil Segev, and Adam Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. *IEEE Transactions on Information Theory*, 54(6):2408–2425, 2008.

[19] Christopher Portmann. Key recycling in authentication. *IEEE Transactions on Information Theory*, 60(7):4383–4396, 2014.

[20] Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. *IACR Cryptology ePrint Archive*, 2004:332, 2004.

[21] Gustavus J Simmons. Authentication theory/coding theory. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 411–431. Springer, 1984.

[22] Gustavus J Simmons. A survey of information authentication. *Proceedings of the IEEE*, 76(5):603–620, 1988.

[23] Douglas R Stinson. Universal hashing and authentication codes. In *Annual International Cryptology Conference*, pages 74–85. Springer, 1991.

[24] Douglas R. Stinson. Combinatorial techniques for universal hashing. *Journal of Computer and System Sciences*, 48(2):337–346, 1994.

[25] Mark N Wegman and J Lawrence Carter. New hash functions and their use in authentication and set equality. *Journal of computer and system sciences*, 22(3):265–279, 1981.

[26] Aaron D Wyner. The wire-tap channel. *Bell Labs Technical Journal*, 54(8):1355–1387, 1975.