Key recycling in authentication

Christopher Portmann*[1,2]

[1]Institute for Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland.
[2]Group of Applied Physics, University of Geneva, 1211 Geneva, Switzerland.

May 31, 2012

Abstract

In their seminal work on authentication, Wegman and Carter propose that to authenticate multiple messages, it is sufficient to reuse the same hash function as long as each tag is encrypted with a one-time pad. They argue that because the one-time pad is perfectly hiding, the hash function used remains completely unknown to the adversary.

Since their proof is not composable, we revisit it using a universally composable framework. It turns out that the above argument is insufficient: information about the hash function is in fact leaked in every round to the adversary, and after a bounded finite amount of rounds it is completely known. We show however that this leak is very small, and Wegman and Carter's protocol is still ε-secure, if ε-almost strongly universal$_2$ hash functions are used. This implies that the secret key corresponding to the choice of hash function can be recycled for any task without any additional error than this ε.

We illustrate this by applying it to quantum key distribution (QKD): if the same hash function is recycled to authenticate the classical communication in every round of a QKD protocol, and used ℓ times per round, the total error after r rounds is upper bounded by $r(\ell\varepsilon + \varepsilon')$, where ε' is the error of one round of QKD given an authentic channel.

1 Introduction

If a player, say, Bob, receives a message x that claims to come from Alice, he might wish to know if this is true, or if the message was generated or modified by some adversary. This task is called *authentication*, and in their seminal work [1], Wegman and Carter showed that it can be achieved with information-theoretic security by appending a tag t to the message (often called a message authentication code or MAC), where $t = h_k(x)$, $\{h_k\}_{k\in\mathcal{K}}$

*chportmann@itp.phys.ethz.ch

is a family of almost strongly universal$_2$ (ASU$_2$) hash functions[1], and k is a secret key shared by Alice and Bob.

There exist ASU$_2$ families $\mathcal{H}$ that require $\log|\mathcal{H}| \approx 2\log\log|\mathcal{X}| + 3\log|\mathcal{T}|$ bits of shared secret key [3], where $\mathcal{X}$ is the message alphabet and $\mathcal{T}$ the tag alphabet. Wegman and Carter [1] propose a scheme to use even less bits of key when multiple messages are to be authenticated: each tag should be encrypted with a fresh one-time pad (OTP), but the same hash function can be used each time. Alice thus appends the tag $t_i = h_{k_1}(x_i) \oplus k_2^i$ to her i^{th} message x_i, where k_1 is used for all messages and k_2^i is a fresh key used only in this round. Asymptotically this scheme consumes only $\log|\mathcal{T}|$ bits of key per round.

To prove the security of this scheme, Wegman and Carter show that given any amount of message-tag pairs $(x_1, t_1), (x_2, t_2), \dots$, the secret key k_1 is still perfectly uniform. They then argue that the probability of an adversary successfully falsifying any new message is the same in every round, and guaranteed to be small by the properties of the ASU$_2$ hash functions.

However, proving that a protocol is secure in a stand-alone model does not necessarily guarantee that it is still secure when combined with other protocols, not even when combined with itself like Wegman and Carter's scheme. A lot of research has gone into composability of cryptographic tasks in recent years. A general framework for proving composable security was developed by Canetti [4, 5], and dubbed *Universally Composable (UC) security*. Independently, Backes, Pfitzmann and Waidner [6, 7] introduced the equivalent notion of *Reactive Simulatability*. These security notions have been extended to the quantum setting by Ben-Or and Mayers [8] and Unruh [9, 10].

An essential application of information-theoretic authentication is in quantum key distribution (QKD) protocols.[2] Every (classical) message exchanged between the two parties generating the key needs to be authenticated with information-theoretic security in order to guarantee the overall unconditional security of the protocol. Recycling the hash function is a practical way to save a large part of the secret key consumed in each round. And as Wegman and Carter's security proof does not fit in any composable security framework, it raises the question of whether this QKD application is still secure.

1.1 Related work

Many works, e.g., [12–15], reuse Wegman and Carter's authentication scheme with key recycling, and they all sketch the security using the same non-com-

[1] ASU$_2$ hashing was only formally defined later by Stinson [2]. A family of functions is said to be ASU$_2$ if any two different messages are almost uniformly mapped to all pairs of tags. An exact definition is given in Definition 3.1 on page 7.

[2] We refer to textbooks such as [11] for a general overview of QKD.

posable argument as Wegman and Carter. Composable security for key recycling in authentication has been studied in the case of quantum messages by Hayden, Leung and Mayers [16], but to the best of our knowledge has not been treated when the messages are classical.[3] Computationally secure variants of Wegman and Carter's scheme have been proposed [17,18], but not analysed in a composable framework either.[4]

Some works [20,21] have pointed out that information-theoretic authentication might not be composable with QKD. They attempt to study this problem by analyzing the security of authentication when the secret keys used are not perfect. In particular, Abidin and Larsson [21] suggest that when QKD and authentication with key recycling are combined recursively, and the (imperfect) secret key resulting from QKD is fed back into the next round of authentication and QKD, the total error could increase exponentially in the number of rounds.

Ben-Or et al. [22] and Müller-Quade and Renner [23] provide a detailed treatment of the composability of QKD. In particular, the latter show that if the authentication and QKD schemes are proven to be composable, and if a short initial key is available, a continuous key stream can be generated with arbitrarily small error.

1.2 New results

We therefore study Wegman and Carter's authentication scheme with key recycling [1] using the UC framework from [5]. Even though we ultimately wish to show that this protocol is composable in a quantum world, it is sufficient to consider classical UC security, since Unruh's lifting theorem [10] proves that classical UC security of a classical scheme implies quantum UC security.

We show that the recycled hash function is gradually leaked to the adversary, even when the key used for the OTP is perfect. This leakage is however very small: we prove that this scheme is indeed ε-UC-secure if the hash functions used are ε-ASU$_2$. Since for any good ASU$_2$ hash function construction ε decreases exponentially fast in the size of the family, $\log|\mathcal{H}|$, it can easily be made arbitrarily small. As a consequence, the doubts of [20,21] are unfounded for all ranges of the parameter ε.

In fact, the hash functions used are slightly weaker than ASU$_2$ hashing, namely almost XOR universal$_2$ (AXU$_2$) hash functions.[5] We show that the

[3]Standard information-theoretic authentication — in which a different hash function is used for every new message — has not been explicitly proven to be composable either, but as we note in Remark 3.2 on page 8, it reduces to stand-alone security, and therefore is immediate from Stinson's work [2].

[4]Though the composability of public-key authentication constructed from digital signatures has been extensively studied [19].

[5]A family of functions is said to be AXU$_2$ if the bitwise XOR of the hash of any two different messages is almost uniform over the choice of hash function. See Definition 4.1

recycled key is close to perfectly uniform and independent from all other random variables produced throughout the protocol. This means that the recycled key can be reused for any task, not only for subsequent rounds of authentication.

An immediate implication of this and the composition theorem [5] is[6] that if this authentication scheme is used ℓ times in each round of an ε'-UC-secure QKD protocol, which is run r times, recycling the same hash function throughout, the final key has distance at most $r(\ell\varepsilon + \varepsilon')$ from uniform.

1.3 Structure of this paper

In Section 2 we introduce the elements from the UC framework that we need in this work. We briefly define the security notion and state the universal composition theorem. In Section 3 we model the UC security for standard authentication without recycling. In Section 4 we then model authentication with key recycling, and prove that using ε-AXU$_2$ hash functions and a OTP results in a scheme which is ε-UC-secure. In Section 5 we take a closer look at the secret key which is leaked to the environment, and show that an optimal attack over ℓ rounds of authentication which takes advantage of this key leakage produces an error of size exactly $\ell\varepsilon$. And finally in Section 6 we illustrate the composition theorem by applying it to the case of many rounds of authentication with key recycling and QKD.

In Appendix A we give a proof of security of standard authentication, as defined in Section 3. And in Appendix B we provide some more details on impersonation attacks.

2 UC security

The UC framework is a very general method allowing arbitrary multipartite cryptographic protocols to be represented and analyzed. Here we focus only on the elements needed for our analysis of information-theoretic authentication, e.g., we do not need to model corruption or consider the running time of the adversary or environment. In particular, it is sufficient to consider classical UC security, even though we need the scheme to be secure in a quantum world and composable with QKD. This is due to Unruh's lifting theorem [10, Theorem 15], which shows that any classical protocol which is classically UC-secure is also quantum UC-secure. For a complete treatment of UC security we refer to [5] and [10] for the classical and quantum settings respectively.

The essence of UC security is to compare the real situation — involving players following the given protocol and an active adversary — to some ideal

on page 10 for an exact definition.

[6] Technically we also need Unruh's lifting theorem [10] for this statement to be absolutely correct.

process. If the two cannot be distinguished by the environment — in particular, if the adversary cannot achieve something which is impossible within the ideal process — then one can be substituted for the other in any setting. For example, if a key distribution protocol is indistinguishable from the ideal setting in which the parties receive a perfect key from a trusted source, then any encryption protocol that is secure with a perfect key is also secure when this key distribution protocol is used instead, i.e., the two protocols can be composed. This gives rise to the universal composition theorem: any two protocols which are UC secure can be concurrently composed and remain secure.

More precisely, for every task considered we need to define some ideal functionality $\mathcal{F}$, which takes all the inputs from the parties and performs the desired task in an ideal way. For example, in the case of authentication analyzed in Section 3, it is always possible for an adversary to cut or jumble the line, making sure the original message is not received. The ideal functionality can thus at best guarantee that the receiver gets either the original message or an error. It receives a message x from the sender and either a `block` or a `let through` command from the adversary, and then delivers x to the receiver or produces an error message depending on the adversary's choice.

The environment $\mathcal{Z}$ is allowed to choose the inputs given to every party, receives all outputs and can communicate freely throughout the protocol with the adversary $\mathcal{A}$. Since the communication between the adversary and the ideal functionality $\mathcal{F}$ is different from when he interacts with the real players, he could immediately alert the environment $\mathcal{Z}$ of this. In the ideal process we therefore replace $\mathcal{A}$ by a simulator $\mathcal{S}$, which can be seen as a buffer between the environment and the ideal functionality. $\mathcal{S}$ often internally simulates $\mathcal{A}$, from which it gets its name.

Definition 2.1 (UC security [5])**.** A protocol π ε-UC-realizes the ideal functionality $\mathcal{F}$, or, more succinctly, is ε-UC-secure, if for all adversaries $\mathcal{A}$ there exists a simulator $\mathcal{S}$ for which no environment $\mathcal{Z}$ can distinguish with probability more than ε if it is interacting with $\mathcal{A}$ and players running π or $\mathcal{S}$ and players using the ideal functionality $\mathcal{F}$.

We illustrate this for the case of authentication in Figure 1.

The above definition can be simplified [5]: it is sufficient to consider a dummy adversary that forwards all messages to the environment and lets it decide what responses to send. Security against all other adversaries holds if it holds for the dummy adversary. In the next sections we therefore restrict our proofs to the dummy adversary.

The main composability theorem can now be stated:

Theorem 2.2 (Universal composition [5])**.** *Let π and ρ be two protocols such that ρ ε_1-UC-realizes $\mathcal{F}$ and $\pi^{\mathcal{F}}$ ε_2-UC-realizes $\mathcal{G}$ when using $\mathcal{F}$ as a subroutine. Then π^{ρ} $(\varepsilon_1 + \varepsilon_2)$-UC-realizes $\mathcal{G}$ when using ρ as a subroutine.*

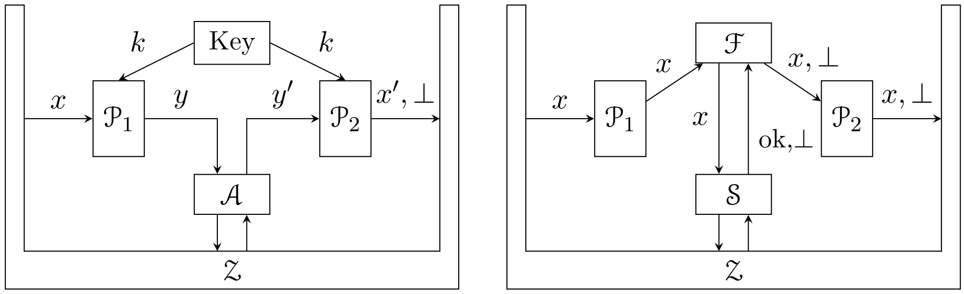

Figure 1 – On the left: the real situation. Players $\mathcal{P}_1$ and $\mathcal{P}_2$ run the authentication protocol using their shared key k. The output of $\mathcal{P}_2$ is either an error $\perp$ if he detected some cheating, or the message x' which might or might not be equal to the input x. On the right: the ideal situation. Some ideal functionality $\mathcal{F}$ either gives $\mathcal{P}_2$ the original message x or an error $\perp$ depending on the decision of the simulator $\mathcal{S}$.

3 Standard authentication

Information-theoretic authentication is usually considered in a setting where two players $\mathcal{P}_1$ and $\mathcal{P}_2$ share a secret key $k \in \mathcal{K}$ and are connected by a channel under the control of an adversary. They wish to guarantee that a message received by $\mathcal{P}_2$ claiming to come from player $\mathcal{P}_1$ was neither generated nor altered by the adversary. These two types of attacks are often called *impersonation* and *substitution*.

For any protocol which authenticates a message by appending a tag, i.e., sends $y = (x, t)$ when the message is x, security against impersonation attacks follows from security against substitution attacks. Since this is the only kind of protocol that we are concerned with, we only consider security against substitution attacks in the body of this paper, and refer to Appendix B for the proof of this reduction.

In Section 3.1 we define UC security for authentication, and in Section 3.2 we describe a secure authentication protocol, the proof of which is given in in Appendix A.

3.1 Security

To send a message x, $\mathcal{P}_1$ uses the key k to generate a new message y containing some redundancy, e.g., $y = (x, h_k(x))$ where $\{h_k\}_{k \in \mathcal{K}}$ is a family of hash functions. Upon receiving y', $\mathcal{P}_2$ checks whether it is valid given the key k, and if so, outputs the corresponding[7] x'. In the previous example with $y' = (x', t')$, $\mathcal{P}_2$ checks that $t' = h_k(x')$ and accepts x' if this is the case or produces an error $\perp$ otherwise. This protocol is depicted on the left in Figure 1.

[7]The function $y = f_k(x)$ has to be injective to guarantee the uniqueness of x'.

Since the channel is completely under the control of the adversary, he can always cut it or completely jumble the message. Hence in the ideal case it is not possible to guarantee that the original message is received, only that $\mathcal{P}_2$ is not tricked into accepting a falsified message. The ideal functionality $\mathcal{F}$ for authentication can be seen as a perfect channel with a switch controlled by the adversary: he can either switch it on and let the message through, or switch it off and let it produce an error. This is depicted on the right in Figure 1.

An authentication scheme is then ε-UC-secure if the environment $\mathcal{Z}$ cannot distinguish between these two situations. To get a more concrete security criterion, we need to define the simulator's actions in the case of the dummy adversary $\mathcal{A}$, who simply forwards y to $\mathcal{Z}$ and forwards the response y' to $\mathcal{P}_2$.

After receiving the message x from the ideal functionality $\mathcal{F}$, the simulator $\mathcal{S}$ must send some y to $\mathcal{Z}$. To do so, it picks a key $k \in \mathcal{K}$ uniformly at random and runs the same protocol as $\mathcal{P}_1$ to generate y. When it gets y' from $\mathcal{Z}$, it checks whether $y = y'$ and sends either ok or $\perp$ to $\mathcal{F}$. Note that this simulator always accepts if the message was not modified, i.e., the ideal case has perfect robustness.

Let X be the random variable describing the initial message $x \in \mathcal{X}$, Y the corresponding encoding generated by $\mathcal{P}_1$ in the real case and $\mathcal{S}$ in the ideal case, and Y' the response from $\mathcal{Z}$. Let $\tilde{X}$ and $\hat{X}$ be random variables over the alphabet $\mathcal{X} \cup \{\perp\}$ describing the outputs of $\mathcal{P}_2$ in the real and ideal cases respectively. In the real case $\mathcal{Z}$ has access to the joint random variable $XYY'\tilde{X}$ and in the ideal case it sees $XYY'\hat{X}$. An authentication scheme is then ε-UC-secure if for any X and Y' chosen by $\mathcal{Z}$, the statistical distance between the real and ideal cases satisfies

$$\frac{1}{2} \sum_{x,y,y',\bar{x}} \left| P_{XYY'\tilde{X}}(x,y,y',\bar{x}) - P_{XYY'\hat{X}}(x,y,y',\bar{x}) \right| \leq \varepsilon. \tag{1}$$

Note that after rearranging the LHS of Eq. (1), we obtain the standard security criterion for authentication [1,2], see also Remark 3.2.

3.2 Protocol

To satisfy Eq. (1) it is sufficient to use a family of strongly universal$_2$ hash functions $\{h_k\}_{k\in\mathcal{K}}$ and define $y := (x, h_k(x))$ with the key k distributed uniformly over $\mathcal{K}$. Then, as described above, $\mathcal{P}_2$ checks that $t' = h_k(x')$ for the $y' = (x', t')$ he receives, and accepts x' if this is the case or produces an error $\perp$ otherwise.

Definition 3.1 (strongly universal$_2$ hash function [2][8]**).** A family of hash

[8] The more common definition of strongly universal$_2$ hashing [2,3,15,24] has an extra condition, namely that for all $x \in \mathcal{X}$ and $t \in \mathcal{T}$, $\Pr[h_k(x) = t] = \frac{1}{|\mathcal{T}|}$. This is however not a necessary condition to prove the security of authentication, so we omit it.

functions $\{h_k : \mathcal{X} \to \mathcal{T}\}_{k\in\mathcal{K}}$ is said to be ε-almost strongly universal$_2$ (ε-ASU$_2$) if for k chosen uniformly at random and all $x_1, x_2 \in \mathcal{X}$ with $x_1 \neq x_2$ and all $t_1, t_2 \in \mathcal{T}$,

$$\Pr\left[h_k(x_1) = t_1 \text{ and } h_k(x_2) = t_2\right] \leq \frac{\varepsilon}{|\mathcal{T}|}. \tag{2}$$

Remark 3.2. If we replace the authenticated message y by its form $(x, h_k(x))$ in Eq. (1), we recover the security condition used by Wegman and Carter [1] and Stinson [2]. It is therefore immediate that previous work on information-theoretic authentication (without key recycling) is composable. However, since we use a simplification of the more common ε-ASU$_2$ hashing definition[8], we cannot directly apply Stinson's proof [2] here. We therefore provide a new proof in Appendix A that standard authentication with ε-ASU$_2$ hashing results in an ε-UC-secure scheme.

4 Authentication with key recycling

If we wish to authenticate many messages and we use the protocol from Section 3, a new hash function and therefore a (completely) new key must be used in every round. This is however not necessary: as we show in Section 4.2, part of the key used to choose the hash function is ε-close to uniform from the point of view of the environment, and can therefore be recycled for further use. Before proving this, we first model this new protocol and its ideal functionality in Section 4.1.

Like for standard authentication analyzed in Section 3, we consider only substitution attacks here, in which the adversary modifies a valid message and tag. For impersonation attacks we refer to Appendix B.

4.1 Security

To model the key recycling, we must view this recycled key as an extra output of the protocol. An authentication scheme with key recycling can be seen as a combination of a key distribution protocol — which only has one output, a secret key — and an authentication scheme — which only has one output, a message. For simplicity we also split the ideal secret key shared by the two players in two parts, $k = (k_1, k_2)$, one which is recycled, k_1, and one which is consumed, k_2. The rest of the model is the same as for standard authentication described in Section 3.1: $\mathcal{P}_1$ uses the shared key (k_1, k_2) to generate a new message y containing some redundancy, $\mathcal{P}_2$ checks that y' is a valid message given (k_1, k_2) and accepts the corresponding x' if that is the case. This is depicted on the left in Figure 2.

In the ideal case, the ideal functionality $\mathcal{F}$ generates a new secret key k_1, which is therefore perfectly uniform and independent from the environment. The rest is identical to standard authentication. The ideal functionality

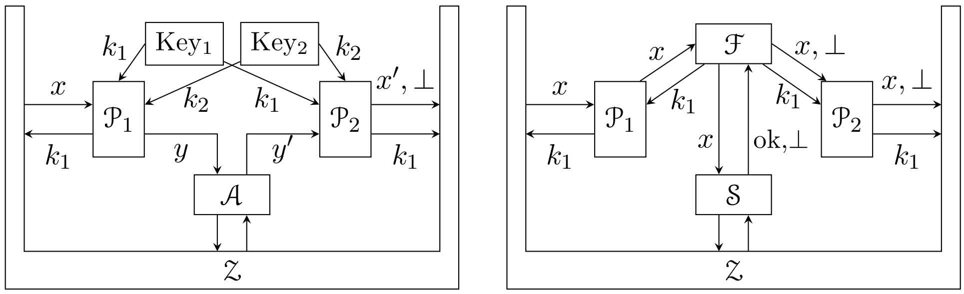

Figure 2 – On the left: the real situation. Players $\mathcal{P}_1$ and $\mathcal{P}_2$ run the authentication protocol using their shared keys k_1, k_2. They both output k_1 for recycling and $\mathcal{P}_2$ additionally produces either an error $\perp$ if he detected some cheating, or the message x' which might or might not be equal to the input x. On the right: the ideal situation. Some ideal functionality $\mathcal{F}$ generates a perfect key k_1 and additionally either gives $\mathcal{P}_2$ the original message x or an error $\perp$ depending on the decision of the simulator $\mathcal{S}$.

also sends either the original message x or an error $\perp$ to $\mathcal{P}_2$ depending on the decision of the simulator $\mathcal{S}$. The simulator $\mathcal{S}$ for the dummy adversary generates its own local keys k_1 and k_2 and runs the same protocol as $\mathcal{P}_1$ to generate y. Upon receiving y' from the environment it checks whether $y' = y$ and sends either `ok` or $\perp$ to $\mathcal{F}$. Here too, the simulator always accepts if the message was not modified, i.e., the ideal case has perfect robustness. This is depicted on the right in Figure 2.

Let X be the random variable describing the initial message $x \in \mathcal{X}$, Y the corresponding encoding generated by $\mathcal{P}_1$ in the real case and $\mathcal{S}$ in the ideal case, and Y' the response from $\mathcal{Z}$. Let $\tilde{X}$ and $\hat{X}$ be random variables over the alphabet $\mathcal{X} \cup \{\perp\}$ describing the outputs of $\mathcal{P}_2$ in the real and ideal cases respectively. And finally let $\tilde{K}$ and K be the random variables for the distribution of k_1 in the real and ideal cases respectively. Thus, in the real case $\mathcal{Z}$ has access to the joint random variable $XYY'\tilde{X}\tilde{K}$ and in the ideal case it sees $XYY'\hat{X}K$. An authentication scheme is then ε-UC-secure if for any X and Y' chosen by $\mathcal{Z}$, the statistical distance between the real and ideal cases satisfies

$$\frac{1}{2} \sum_{x,y,y',\bar{x},k_1} \left| P_{XYY'\tilde{X}\tilde{K}}(x,y,y',\bar{x},k_1) - P_{XYY'\hat{X}K}(x,y,y',\bar{x},k_1) \right| \leq \varepsilon. \quad (3)$$

4.2 Protocol

The protocols we wish to analyze in this setting encode a message x as $y = (x, h_{k_1}(x) \oplus k_2)$, where $\{h_{k_1} : \mathcal{X} \to \mathcal{T}\}_{k_1 \in \mathcal{K}}$ is a family of hash functions that map the message to some bit string $t \in \mathcal{T} = \{0,1\}^m$, $k_2 \in \mathcal{T}$ and $\oplus$ is the bitwise XOR. Then, as described above, $\mathcal{P}_2$ checks that $t' = h_{k_1}(x') \oplus k_2$ for the $y' = (x', t')$ he receives, and accepts x' if this is the case or produces an error $\perp$ otherwise.

These hash functions do not need to be ε-ASU$_2$, it is sufficient for $g_{k_1,k_2}(x) = h_{k_1}(x) \oplus k_2$ to have this property. The property needed for $\{h_{k_1}\}_{k_1 \in \mathcal{K}}$ has been dubbed *ε-almost XOR universal$_2$* by Rogaway [14], *ε-otp secure* by Krawczyk [12, 13], and *ε-Δ universal* by Stinson [24].[9]

Definition 4.1 (XOR universal$_2$ hash function [14])**.** A family of hash functions $\{h_k : \mathcal{X} \to \mathcal{T}\}_{k \in \mathcal{K}}$ for $\mathcal{T} = \{0,1\}^m$ is said to be ε-almost XOR universal$_2$ (ε-AXU$_2$) if for k chosen uniformly at random and all $x_1, x_2 \in \mathcal{X}$ with $x_1 \neq x_2$ and all $t \in \mathcal{T}$,

$$\Pr\left[h_k(x_1) \oplus h_k(x_2) = t\right] \leq \varepsilon. \tag{4}$$

It is immediate from this definition that the hash function $g_{k_1,k_2}(x) := h_{k_1}(x) \oplus k_2$ is ε-ASU$_2$, i.e., for all $x_1, x_2 \in \mathcal{X}$ with $x_1 \neq x_2$ and all $t_1, t_2 \in \mathcal{T}$,

$$\Pr\left[g_{k_1,k_2}(x_1) = t_1 \text{ and } g_{k_1,k_2}(x_2) = t_2\right] \leq \frac{\varepsilon}{|\mathcal{T}|}.$$

Since XORing a uniform string k_2 to any value yields a uniform string we also have

$$\Pr\left[g_{k_1,k_2}(x_1) = t_1\right] = \Pr\left[g_{k_1,k_2}(x_1) = t_1 \middle| \tilde{K} = k_1\right] = \frac{1}{|\mathcal{T}|}, \tag{5}$$

where $\tilde{K}$ is the random variable for the recycled part of the key. Combining the two equations above gives

$$\Pr\left[g_{k_1,k_2}(x_2) = t_2 | g_{k_1,k_2}(x_1) = t_1\right] \leq \varepsilon. \tag{6}$$

We now have all the ingredients needed to prove the security.

Theorem 4.2. *Let π be an authentication scheme that encodes a message x as $y = (x, h_{k_1}(x) \oplus k_2)$ and recycles k_1, where $\{h_{k_1} : \mathcal{X} \to \mathcal{T}\}_{k_1 \in \mathcal{K}}$ is a family of ε-almost XOR universal$_2$ hash functions, and (k_1, k_2) are chosen uniformly at random from $\mathcal{K} \times \mathcal{T}$. Then π is ε-UC-secure.*

Proof. We need to show that Eq. (3) is satisfied for any distributions P_X and $P_{Y'}$. First, for $y = (x,t)$, if the environment chooses $y' = (x,t')$, then the real and ideal protocols both either accept x if $t' = t$ or reject it if $t' \neq t$. This means that

$$\begin{aligned} P_{\tilde{X}|XYY'}(x'|x,(x,t),(x,t')) &= P_{\hat{X}|XYY'}(x'|x,(x,t),(x,t')) \\ &= \begin{cases} 0 & \text{if } x' = \perp \text{ XOR } t' = t, \\ 1 & \text{otherwise,} \end{cases} \end{aligned}$$

[9] Stinson [24] generalizes this notion to any additive abelian group $\mathcal{T}$ instead of only bit strings.

hence $\tilde{X}$ and $\hat{X}$ are completely determined by XYY' and can be dropped. The LHS of Eq. (3) thus reduces to

$$\frac{1}{2}\sum_{x,t,t',k_1} P_{XYY'}(x,(x,t),(x,t'))\left|P_{\tilde{K}|XYY'}(k_1|x,(x,t),(x,t')) - \frac{1}{|\mathcal{K}|}\right|.$$

Furthermore, from Eq. (5) we know that $\tilde{K}$ is independent from XY, and therefore also from XYY', hence $P_{\tilde{K}|XYY'}(k_1|x,(x,t),(x,t')) = \frac{1}{|\mathcal{K}|}$. We can thus assume w.l.o.g. that the adversary chooses $x' \neq x$.

For $y' = (x',t')$ and $x' \neq x$, the random variable $\tilde{X}$ can take two values, $\perp$ if cheating was detected or x' if the players were fooled. $\hat{X}$ however always produces an error $\perp$. Separating the summation in the LHS of Eq. (3) over these two values gives

$$\frac{1}{2}\sum_{x,t,x',t',k_1} P_{XYY'}(x,(x,t),(x',t'))\left|P_{\tilde{X}\tilde{K}|XYY'}(\perp,k_1|x,(x,t),(x,t')) - \frac{1}{|\mathcal{K}|}\right| \\ + \frac{1}{2}\sum_{x,t,x',t',k_1} P_{XYY'\tilde{X}\tilde{K}}(x,(x,t),(x',t'),x',k_1). \quad (7)$$

We show in the following that

$$P_{\tilde{X}\tilde{K}|XYY'}(\perp,k_1|x,(x,t),(x',t')) \leq \frac{1}{|\mathcal{K}|} \quad (8)$$

for all values of x,t,x',t',k_1. This implies that Eq. (7) sums up to twice the value of the second term, which can be bounded in the following way:

$$\begin{aligned}
&\sum_{x,t,x',t',k_1} P_{XYY'\tilde{X}\tilde{K}}(x,(x,t),(x',t'),x',k_1) \\
&\quad = \sum_{x,t,x',t'} P_{XYY'}(x,(x,t),(x',t'))\sum_{k_1} P_{\tilde{X}\tilde{K}|XYY'}(x',k_1|x,(x,t),(x',t')) \\
&\quad = \sum_{x,t,x',t'} P_{XYY'}(x,(x,t),(x',t'))P_{\tilde{X}|XYY'}(x'|x,(x,t),(x',t')) \\
&\quad \leq \sum_{x,t,x',t'} P_{XYY'}(x,(x,t),(x',t'))\varepsilon = \varepsilon,
\end{aligned}$$

where to reach the last line we used

$$P_{\tilde{X}|XYY'}(x'|x,(x,t),(x',t')) = \Pr\left[h_{k_1}(x') \oplus k_2 = t' \middle| h_{k_1}(x) \oplus k_2 = t\right]$$

and Eq. (6).

Finally, we show that Eq. (8) holds. The LHS can be decomposed as

$$\begin{aligned}
&P_{\tilde{X}\tilde{K}|XYY'}(\perp,k_1|x,(x,t),(x',t')) \\
&\quad = P_{\tilde{K}|XYY'}(k_1|x,(x,t),(x',t'))P_{\tilde{X}|XYY'\tilde{K}}(\perp|x,(x,t),(x',t'),k_1). \quad (9)
\end{aligned}$$

Because (x', t') are chosen by the environment when holding (x, t), they do not influence the distribution of $\tilde{K}$ given XY, so

$$P_{\tilde{K}|XYY'}(k_1|x, (x,t), (x',t')) = P_{\tilde{K}|XY}(k_1|x, (x,t)).$$

And as argued above in the case where $x' = x$, from Eq. (5) we know that $\tilde{K}$ is independent from XY, so $P_{\tilde{K}|XY}(k_1|x, (x,t)) = \frac{1}{|\mathcal{K}|}$. Combining this with $P_{\tilde{X}|XYY'\tilde{K}}(\perp|x, (x,t), (x',t'), k_1) \leq 1$ and Eq. (9) proves Eq. (8). $\square$

5 Secret key leakage

An immediate application of the UC security of authentication with key recycling is to reuse the same hash function to authenticate multiple messages, only renewing the part of the key XORed to the tag. The universal composition theorem (Theorem 2.2) says that if we do this ℓ times and each individual protocol is ε-UC-secure, then the composed protocol has error at most $\ell\varepsilon$.[10]

In this section we show that this composition theorem is tight for all protocols with $\varepsilon = 1/|\mathcal{T}|$, where $\mathcal{T}$ is the alphabet for the tag, i.e., there exists an attack such that after ℓ rounds the adversary has probability at least $\ell\varepsilon$ of having successfully forged a message, for any $\ell \leq 1/\varepsilon$.

Let us define $\{F_\ell\}_\ell$ to be a sequence of random variables taking the value 1 if the adversary successfully falsifies a message in any of the first ℓ rounds, and 0 otherwise. Then a quick calculation shows us that for any $0 \leq \ell \leq 1/\varepsilon - 1$,

$$P_{F_{\ell+1}|F_\ell}(1|0) = \frac{P_{F_{\ell+1}}(1) - P_{F_{\ell+1}F_\ell}(1,1)}{P_{F_\ell}(0)} = \frac{(\ell+1)\varepsilon - \ell\varepsilon}{1 - \ell\varepsilon} = \frac{\varepsilon}{1 - \ell\varepsilon}.$$

This means that in every successive round, the adversary's probability of successfully forging a message increases. This happens because — as we show in Theorem 5.1 here below — some information about the hash function is leaked in every round, even if the key used for the OTP is perfectly uniform. The entropy of the hash function gradually decreases, until the adversary has enough information to successfully falsify a new message with probability 1.

This result contrasts strongly with the non-composable analysis found in [1]. There, the adversary simply collects the pairs of messages and tags $(x_1, t_1), (x_2, t_2), \ldots$, and attempts to falsify a message in each round, independently from the attempts in previous rounds. In this case, due to the hiding property of the OTP, the distribution of the hash function always remains perfectly uniform given these message-tag pairs.

Theorem 5.1. *Let π be an authentication scheme that encodes a message x as $y = (x, h_{k_1}(x) \oplus k_2)$ and recycles k_1, where $\{h_{k_1} : \mathcal{X} \to \mathcal{T}\}_{k_1 \in \mathcal{K}}$ is a*

[10] We illustrate this application of the composition theorem in Section 6.

family of $\frac{1}{|\mathcal{T}|}$-almost XOR universal$_2$ hash functions. For any $1 \leq \ell \leq |\mathcal{T}|$, let this protocol be used ℓ times with the same key $k_1 \in \mathcal{K}$ initially chosen uniformly at random, and a new uniformly random $k_2 \in \mathcal{T}$ in each round. Then, there exists an attack that allows the adversary to successfully falsify one of the first ℓ messages with probability at least $\ell/|\mathcal{T}|$. Furthermore, after ℓ rounds, the entropy of the recycled key K_1 is bounded by

$$H(K_1|Z_\ell) \leq \log \frac{|\mathcal{K}|}{|\mathcal{T}|} + \left(1 - \frac{\ell}{|\mathcal{T}|}\right) \log\left(|\mathcal{T}| - \ell\right),$$

where Z_ℓ consists of all the inputs and outputs of the protocol (except for K_1) and the communication with the dummy adversary from these ℓ rounds.

Proof. Since the environment can choose the distribution of the messages to be authenticated in the ℓ rounds, we take them to always be the same message x. The environment also always substitutes the same message $x' \neq x$ for x in each round. To be successful, it needs to guess correctly the value $c = h_{k_1}(x) \oplus h_{k_1}(x')$, since $t' = t \oplus c$, where t is the tag that comes with x and t' is the correct tag for x'. The environment therefore makes a list of the $|\mathcal{T}|$ possible values for c, and in each round eliminates one from its list.

In the first round the environment is given (x, t_1) by the dummy adversary. It picks a c_1 from its list and sends $(x', t_1 \oplus c_1)$ back to the dummy adversary. The legitimate player accepts the message received from the adversary only if $c_1 = h_{k_1}(x) \oplus h_{k_1}(x')$, which happens with probability $p_1 = |\mathcal{T}|^{-1}$.

If the environment is unsuccessful at falsifying the message, it can cross c_1 off its list. In the second round it then receives (x, t_2), picks a new $c_2 \neq c_1$, and sends $(x', t_2 \oplus c_2)$. This time its success probability is $p_2 = (|\mathcal{T}| - 1)^{-1}$, since it only has $|\mathcal{T}| - 1$ elements c left on its list.

If we repeat this for each round, the success probability in the ℓ^{th} round given that the previous $\ell - 1$ were unsuccessful is $p_\ell = (|\mathcal{T}| - \ell + 1)^{-1}$. We now prove by induction that the probability of successfully falsifying at least one message with this strategy is exactly $\ell/|\mathcal{T}|$. Let F_ℓ be a random variable taking the value 1 if the adversary successfully falsifies a message in any of the first ℓ rounds, and 0 otherwise. We have $P_{F_1}(1) = p_1 = 1/|\mathcal{T}|$. And if $P_{F_{\ell-1}}(1) = (\ell - 1)/|\mathcal{T}|$, then

$$P_{F_\ell}(1) = P_{F_{\ell-1}}(1) + P_{F_{\ell-1}}(0) p_\ell = \frac{\ell - 1}{|\mathcal{T}|} + \left(1 - \frac{\ell - 1}{|\mathcal{T}|}\right) \frac{1}{|\mathcal{T}| - \ell + 1} = \frac{\ell}{|\mathcal{T}|}.$$

Let z_0 represent any value of Z_ℓ in which the adversary fails to falsify any message, and z_1 be the case where he does trick the players. If he is successful, he immediately learns the correct value c, and thus

$$H(K|Z_\ell = z_1) = \log \frac{|\mathcal{K}|}{|\mathcal{T}|}.$$

If the adversary is not successful, he has still managed to cross ℓ values for c off his list, so

$$H(K|Z_\ell = z_0) = \log \frac{|\mathcal{K}|}{|\mathcal{T}|} \left(|\mathcal{T}| - \ell\right).$$

Combining the two equations above with the corresponding probabilities, we get

$$H(K|Z_\ell) = \frac{\ell}{|\mathcal{T}|} \log \frac{|\mathcal{K}|}{|\mathcal{T}|} + \left(1 - \frac{\ell}{|\mathcal{T}|}\right) \log \frac{|\mathcal{K}|}{|\mathcal{T}|} \left(|\mathcal{T}| - \ell\right). \qquad \square$$

6 Example: Layered QKD and authentication

As illustration of the universal composition theorem (Theorem 2.2) we sketch the security proof for a composition of quantum key distribution (QKD) and authentication with key recycling.

The two players must share an initial key which is long enough to select the hash function and encrypt the tags for every message exchanged during the first round of QKD. The recycled key and new key produced by the QKD protocol are then used for the authentication in subsequent rounds. W.l.o.g. we assume the initial key to be perfect. If the authentication scheme is ε_1-UC-secure, and is needed ℓ times in each round of QKD, and the QKD protocol is ε_2-UC-secure and repeated r times, it is immediate from the composition theorem that the final key — the concatenation of all unused secret key bits produced in each round and the recycled hash function — has distance at most $r(\ell\varepsilon_1 + \varepsilon_2)$ from uniform.

To sketch this, we first consider the composition of the ℓ rounds of authentication with key recycling, which we illustrate in Figure 3. In the ideal setting, the different rounds of the ideal authentication are all independent. So the statistical distance between the environment's ($\mathcal{Z}$) view of ℓ rounds of ideal authentication and $\ell - 1$ rounds of ideal authentication with 1 round of the real protocol is at most ε_1. Likewise, if we compare the second and third lines of Figure 3, the environment $\mathcal{Z}'$ can notice a difference with probability at most ε_1. Since $\mathcal{Z}'$ is simply $\mathcal{Z}$ with an additional internal round of authentication (which $\mathcal{Z}$ can run on its own anyway), $\mathcal{Z}$ does not have an advantage greater than ε_1 either. By repeating this reasoning and using the triangle inequality for the statistical distance, we find that ℓ rounds of authentication with key recycling have error at most $\ell\varepsilon_1$.

Next, we look at the composition of ℓ rounds of authentication and 1 round of QKD. Let $\tilde{K}_1$ be the key recycled by the authentication protocols, $\tilde{K}_2$ the output of the QKD protocol, let K_1 and K_2 be their ideal counterparts, and let $\rho_{\tilde{E}}$ and ρ_E be the quantum states held by the environment in the real and ideal cases, consisting of all the information it gathered, the classical messages, tags, falsified messages, and quantum information gleaned

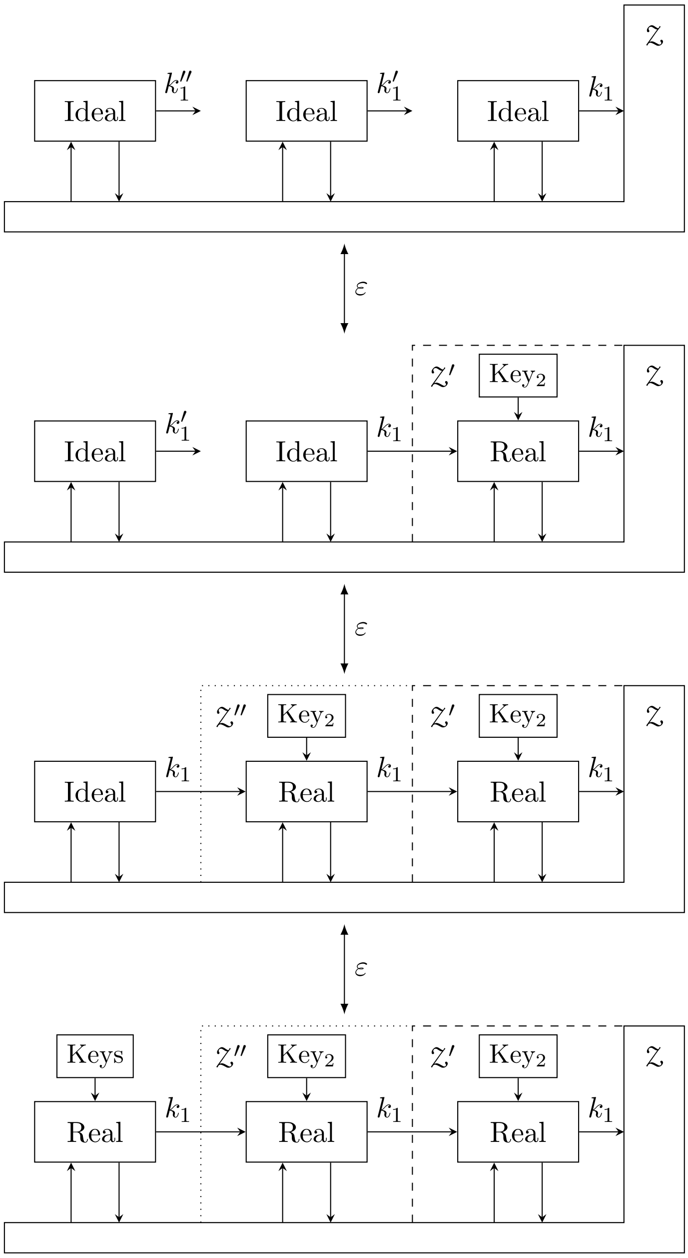

Figure 3 – The bottom line represents the environment's ($\mathcal{Z}$) view of the composition of many rounds of an authentication protocol with key recycling. Each box *Real* contains the two legitimate players and the adversary, depicted separately in Figure 2. The recycled key k_1 is passed from one protocol to the next. The rest of the communication between the environment, players and adversary is stylized by the two arrows beneath the box. The top line represents the ideal case, in which each box *Ideal* contains the players, ideal functionality and simulator. By substituting one real protocol for an ideal one, the distance between the environment's views increases by at most ε.

from the quantum channel. We need to show that the (trace) distance[11] between the real and ideal situations is bounded by $\ell\varepsilon_1 + \varepsilon_2$, i.e.,

$$\frac{1}{2}\left\|\rho_{\tilde{K}_1\tilde{K}_2\tilde{E}} - \rho_{K_1} \otimes \rho_{K_2} \otimes \rho_E\right\|_{\mathrm{tr}} \leq \ell\varepsilon_1 + \varepsilon_2. \tag{10}$$

Since the composition of authentication protocols is close to ideal for all environments, it is in particular secure for an environment that runs a QKD protocol and attempts to distinguish between the real and ideal settings by looking at the output of the QKD protocol. Hence

$$\frac{1}{2}\left\|\rho_{\tilde{K}_1\tilde{K}_2\tilde{E}} - \rho_{K_1} \otimes \rho_{\hat{K}_2 E}\right\|_{\mathrm{tr}} \leq \ell\varepsilon_1,$$

where $\hat{K}_2$ is the output of the QKD protocol run with the ideal authentication. By the security definition of QKD [25], the protocol is ε_2-UC-secure if, when using an ideal authentication protocol, we have

$$\frac{1}{2}\left\|\rho_{\hat{K}_2 E} - \rho_{K_2} \otimes \rho_E\right\|_{\mathrm{tr}} \leq \varepsilon_2.$$

Combining the two equations above and the triangle inequality proves (10).

The final step consists in showing that r rounds of QKD and authentication has error at most $r(\ell\varepsilon_1 + \varepsilon_2)$. The reasoning is however identical to the ℓ sequential compositions of just authentication depicted in Figure 3, so we omit it.

Acknowledgments

The author would like to thank Renato Renner and Christoph Pacher for valuable discussions.

This work has been funded by the Swiss National Science Foundation (via grant No. 200020-135048 and the National Centre of Competence in Research 'Quantum Science and Technology'), the European Research Council – ERC (grant no. 258932) – and by the Vienna Science and Technology Fund (WWTF) through project ICT10-067 (HiPANQ).

Appendices

A Security proof for standard authentication

We prove here the security of standard authentication with ε-ASU$_2$ hashing as defined in Definition 3.1. Note that this proof does not need the extra

[11] The trace distance between two states ρ and σ is the quantum generalization of the statistical distance, and is given by $\frac{1}{2}\|\rho - \sigma\|_{\mathrm{tr}}$, where $\|A\|_{\mathrm{tr}} := \mathrm{tr}\sqrt{A^\dagger A}$.

requirement that $\Pr[h_k(x) = t] = \frac{1}{|\mathcal{T}|}$, which is often part of the ε-ASU$_2$ definition (see Footnote 8 on page 7).

Lemma A.1. *Let π be an authentication scheme that encodes a message x as $y = (x, h_k(x))$, where $\{h_k : \mathcal{X} \to \mathcal{T}\}_{k \in \mathcal{K}}$ is a family of ε-almost strongly universal$_2$ hash functions, and k is chosen uniformly at random from $\mathcal{K}$. Then π is ε-UC-secure.*

Proof. We need to show that Eq. (1) is satisfied for any distributions P_X and $P_{Y'}$. Let $y = (x, t)$ and $y' = (x', t')$. If the environment chooses $x' = x$, both the real protocol and ideal functionally behave identically and are indistinguishable — they both accept x if $t' = t$ and produce an error otherwise. We can therefore assume w.l.o.g. that the environment always chooses $x' \neq x$. In this case, the simulator in the ideal situation always outputs an error $\perp$, i.e., $P_{\hat{X}}(\perp) = 1$. The security criterion (1) therefore reduces to

$$\sum_{x,t,x',t'} P_{XY'Y'\tilde{X}}(x,(x,t),(x',t'),x') \leq \varepsilon.$$

Splitting the random variable $Y = XT$ in its two parts, and combining the following equations,

$$\begin{aligned}
P_{XYY'}(x,(x,t),(x',t')) &= P_{XTY'}(x,t,(x',t')) \\
&= P_X(x)P_{T|X}(t|x)P_{Y'|XT}(x',t'|x,t), \\
P_{T|X}(t|x) &= \Pr[h_k(x) = t], \\
P_{\tilde{X}|XYY'}(x'|x,(x,t),(x',t')) &= \Pr[h_k(x') = t'|h_k(x) = t],
\end{aligned}$$

we get

$$\begin{aligned}
&\sum_{x,t,x',t'} P_{XYY'\tilde{X}}(x,(x,t),(x',t'),x') \\
&\quad = \sum_{x,t,x',t'} P_X(x)P_{Y'|XT}(x',t'|x,t)\Pr[h_k(x) = t \text{ and } h_k(x') = t'] \\
&\quad \leq \sum_{x,t,x',t'} P_X(x)P_{Y'|XT}(x',t'|x,t)\frac{\varepsilon}{|\mathcal{T}|} \\
&\quad = \sum_t \frac{\varepsilon}{|\mathcal{T}|} = \varepsilon.
\end{aligned}$$

□

B Impersonation attacks

B.1 Security

In an impersonation attack, the adversary (or environment in case of a dummy adversary) does not wait for the legitimate parties to authenticate a message, instead he generates his own y' before receiving any y. In the ideal case, the simulator then always sends an error $\perp$ to the ideal functionality who transmits it to $\mathcal{P}_2$.

Since no input x and corresponding y are present, the security criterion for standard authentication (Eq. (1)) then reduces to

$$\frac{1}{2}\sum_{y,x}\left|P_{Y'\tilde{X}}(y,x) - P_{Y'\hat{X}}(y,x)\right| \leq \varepsilon, \tag{11}$$

and we say that an authentication protocol is ε-UC-secure against impersonation attacks if Eq. (11) holds.

In the case of key recycling, the decision of $\mathcal{P}_2$ to accept or reject the message might be correlated to the key k_1, i.e., the random variables $\tilde{X}$ and $\tilde{K}$ can be correlated. It is therefore important that in this setting too, the ideal functionality produces a new key K which is perfectly uniform and independent from $Y'\hat{X}$. The corresponding security criterion (Eq. (3)) then reduces to

$$\frac{1}{2}\sum_{y,x,k_1}\left|P_{Y'\tilde{X}\tilde{K}}(y,x,k_1) - P_{Y'\hat{X}K}(y,x,k_1)\right| \leq \varepsilon, \tag{12}$$

and we say that an authentication protocol with key recycling is ε-UC-secure against impersonation attacks if Eq. (12) holds.

Although these might, at first look, seem like a simplification of their substitution-attack counterparts, it is in fact possible to construct (artificial) protocols that have an impersonation error roughly twice as large as the substitution error.[12] However, in the special case where the encoding of the message x is of the form $y = (x,t)$, we show in the following section that security against impersonation attacks follows from security against substitution attacks.

B.2 Reduction to substitution attacks

Lemma B.1. *Let π be an authentication scheme (with or without key recycling) that encodes a message x as $y = (x,t)$. If π is ε-UC-secure (against substitution attacks), then it is also ε-UC-secure against impersonation attacks.*

[12] Let $\{h_k : \{0,1\} \to \{0,1\}^m\}_k$ be a set of functions such that for all k, $h_k(0) = 0^m$, and 1 is uniformly mapped (over the choice of k) to all $t \in \{0,1\}^m \setminus \{0^m\}$. Let the authentication protocol encode the message $x \in \{0,1\}$ as $(x \oplus k_1, h_{k_2}(x \oplus k_1))$. If the environment performs an impersonation attack by sending the message $y = (0, 0^m)$, this will be accepted with probability 1. If the environment performs a substitution attack, he first has to choose a message x, then receives the corresponding y from the dummy adversary, and has to choose a new $y' \neq y$. For all x, with probability 1/2 the corresponding encoding is $y = (0, 0^m)$, and so the impersonation attack outlined above works only with probability 1/2.

We prove this statement for a scheme with key recycling. The proof when no recycling is performed is identical except for the omission of the random variables K and $\tilde{K}$.

Proof. We assume that the protocol is not ε-UC-secure against impersonation attacks, and construct a substitution attack. Let

$$\frac{1}{2}\sum_{y,x,k_1}\left|P_{Y'\tilde{X}\tilde{K}}(y,x,k_1)-P_{Y'\hat{X}K}(y,x,k_1)\right|>\varepsilon,$$

then there exists a specific $y'=(x',t')$ for which

$$\frac{1}{2}\sum_{x,k_1}\left|P_{\tilde{X}\tilde{K}|Y'}(x,k_1|y')-P_{\hat{X}K|Y'}(x,k_1|y')\right|>\varepsilon.$$

So w.l.o.g. we can take $P_{Y'}(y')=1$.

For the corresponding substitution attack, let the environment choose any distribution P_X such that $P_X(x')=0$. Then upon receiving y, it sends $y'=(x',t')$. In the ideal case, the simulator and ideal functionality therefore always transmit an error, and in the real case $\mathcal{P}_2$ accepts the message x' with the same probability as for the impersonation attack. So $P_{\hat{X}K|Y'}$ and $P_{\tilde{X}\tilde{K}|Y'}$ have exactly the same distributions in the cases of substitution and impersonation attacks. Hence

$$\begin{aligned}\frac{1}{2}\sum_{x,y,y',\bar{x},k_1}&\left|P_{XYY'\tilde{X}\tilde{K}}(x,y,y',\bar{x},k_1)-P_{XYY'\hat{X}K}(x,y,y',\bar{x},k_1)\right|\\&\geq\frac{1}{2}\sum_{y',\bar{x},k_1}\left|P_{Y'\tilde{X}\tilde{K}}(y',\bar{x},k_1)-P_{Y'\hat{X}K}(y',\bar{x},k_1)\right|>\varepsilon.\end{aligned}$$ □

B.3 Tighter bound

The bound on the impersonation error from Lemma B.1 is not always tight. In particular, in the case of the authentication protocol with key recycling given in Section 4.2 it is possible to get a better result. Due to the tight bound in Eq. (5) we find that this scheme is $\frac{1}{|\mathcal{T}|}$-UC-secure against impersonation attacks.

Lemma B.2. *Let π be an authentication scheme that encodes a message x as $y=(x,h_{k_1}(x)\oplus k_2)$ and recycles k_1, where $\{h_{k_1}:\mathcal{X}\to\mathcal{T}\}_{k_1\in\mathcal{K}}$ is a family of ε-almost XOR universal$_2$ hash functions, and (k_1,k_2) are chosen uniformly at random from $\mathcal{K}\times\mathcal{T}$. Then π is $\frac{1}{|\mathcal{T}|}$-UC-secure against impersonation attacks.*

Proof. We need to show that for all distributions $P_{Y'}$, Eq. (12) holds for $\varepsilon = \frac{1}{|\mathcal{T}|}$. Since in the ideal case $\hat{X} = \perp$ and K is independent from $Y'\hat{X}$, the LHS of Eq. (12) reduces to

$$\frac{1}{2} \sum_{x,t,k_1} P_{Y'}(x,t) \left(P_{\tilde{X}\tilde{K}|Y'}(x,k_1|x,t) + \left| P_{\tilde{X}\tilde{K}|Y'}(\perp,k_1|x,t) - \frac{1}{|\mathcal{K}|} \right| \right). \quad (13)$$

And because from Eq. (5),

$$\begin{aligned} P_{\tilde{X}\tilde{K}|Y'}(x,k_1|x,t) &= \Pr\left[h_{k_1}(x) \oplus k_2 = t \text{ and } \tilde{K} = k_1\right] \\ &= P_{\tilde{K}}(k_1) \Pr\left[h_{k_1}(x) \oplus k_2 = t \middle| \tilde{K} = k_1\right] = \frac{1}{|\mathcal{K}||\mathcal{T}|}, \end{aligned}$$

Eq. (13) is equal to $\frac{1}{|\mathcal{T}|}$. ☐

Note that in the case of standard authentication with ε-ASU$_2$ hashing, if we had made the extra assumption that $\Pr\left[h_k(x) = t\right] = \frac{1}{|\mathcal{T}|}$ (see Footnote 8 on page 7), we would also have found that the corresponding scheme is $\frac{1}{|\mathcal{T}|}$-UC-secure against impersonation attacks. However, from Eq. (2) alone, we can at best get the bound

$$\Pr\left[h_k(x) = t\right] = \sum_{t'} \Pr\left[h_k(x) = t \text{ and } h_k(x') = t'\right] \leq \varepsilon,$$

which only guarantees that the scheme is ε-UC-secure against impersonation attacks.

References

[1] Mark N. Wegman and Larry Carter. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22(3):265–279, 1981.

[2] Douglas R. Stinson. Universal hashing and authentication codes. *Designs, Codes and Cryptography*, 4(3):369–380, 1994. A preliminary version appeared at CRYPTO '91. [doi:10.1007/BF01388651].

[3] Jürgen Bierbrauer, Thomas Johansson, Gregory Kabatianskii, and Ben Smeets. On families of hash functions via geometric codes and concatenation. In *Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '93*, pages 331–342. Springer, 1994. [doi:10.1007/3-540-48329-2_28].

[4] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proceedings of the 42nd Symposium on Foundations of Computer Science, FOCS '01*, page 136. IEEE, 2001.

[5] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. Updated version of [4], 2005. [IACR e-print: 2000/067].

[6] Birgit Pfitzmann and Michael Waidner. Composition and integrity preservation of secure reactive systems. In *Proceedings of the 7th ACM Conference on Computer and Communications Security, CSS '00*, pages 245–254. ACM, 2000. [doi:10.1145/352600.352639].

[7] Michael Backes, Birgit Pfitzmann, and Michael Waidner. A general composition theorem for secure reactive systems. In *Proceedings of the First Theory of Cryptography Conference, TCC '04*, pages 336–354. Springer, 2004. [doi:10.1007/978-3-540-24638-1_19].

[8] Michael Ben-Or and Dominic Mayers. General security definition and composability for quantum & classical protocols. eprint, 2004. [arXiv:quant-ph/0409062].

[9] Dominique Unruh. Simulatable security for quantum protocols. eprint, 2004. [arXiv:quant-ph/0409125].

[10] Dominique Unruh. Universally composable quantum multi-party computation. In *Advances in Cryptology, Proceedings of EUROCRYPT '10*, pages 486–505. Springer, 2010. [doi:10.1007/978-3-642-13190-5_25, arXiv:0910.2912].

[11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[12] Hugo Krawczyk. LFSR-based hashing and authentication. In *Proceedings of the 14th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '94*, pages 129–139. Springer, 1994. [doi:10.1007/3-540-48658-5_15].

[13] Hugo Krawczyk. New hash functions for message authentication. In *Advances in Cryptology, Proceedings of EUROCRYPT '95*, pages 301–310. Springer, 1995. [doi:10.1007/3-540-49264-X_24].

[14] Phillip Rogaway. Bucket hashing and its application to fast message authentication. *Journal of Cryptology*, 12(2):91–115, 1999. A preliminary version appeared at CRYPTO '95. [doi:10.1007/PL00003822].

[15] Mustafa Atici and Douglas R. Stinson. Universal hashing and multiple authentication. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '96*, pages 16–30. Springer, 1996. [doi:10.1007/3-540-68697-5_2].

[16] Patrick Hayden, Debbie Leung, and Dominic Mayers. The universal composable security of quantum message authentication with key recycling. Presented at QCrypt 2011, 2011.

[17] Victor Shoup. On fast and provably secure message authentication based on universal hashing. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '96*, pages 313–328. Springer, 1996. [doi:10.1007/3-540-68697-5_24].

[18] Daniel J. Bernstein. Stronger security bounds for wegman-carter-shoup authenticators. In *Advances in Cryptology, Proceedings of EUROCRYPT '05*, pages 164–180. Springer, 2005. [doi:10.1007/11426639_10].

[19] Ran Canetti. Universally composable signature, certification, and authentication. In *Proceedings of the 17th IEEE Computer Security Foundations Workshop*, page 219. IEEE, 2004. [doi:10.1109/CSFW.2004.24, IACR e-print: 2003/239].

[20] Jörgen Cederlöf and Jan-Åke Larsson. Security aspects of the authentication used in quantum cryptography. *IEEE Transactions on Information Theory*, 54(4):1735–1741, 2008. [doi:10.1109/TIT.2008.917697, arXiv:quant-ph/0611009].

[21] Aysajan Abidin and Jan-Åke Larsson. Security of authentication with a fixed key in quantum key distribution. eprint, 2011. [arXiv:1109.5168].

[22] Michael Ben-Or, Michael Horodecki, Debbie Leung, Dominic Mayers, and Jonathan Oppenheim. The universal composable security of quantum key distribution. In *Proceedings of the Second Theory of Cryptography Conference, TCC '05*, pages 386–406. Springer, 2005. [arXiv:quant-ph/0409078].

[23] Jörn Müller-Quade and Renato Renner. Composability in quantum cryptography. *New Journal of Physics*, 11(8):085006, 2009. [doi:10.1088/1367-2630/11/8/085006, arXiv:1006.2215].

[24] Douglas R. Stinson. On the connections between universal hashing, combinatorial designs and error-correcting codes. *Electronic Colloquium on Computational Complexity (ECCC)*, 2(52), 1995.

[25] Renato Renner. *Security of Quantum Key Distribution.* PhD thesis, Swiss Federal Institute of Technology Zurich, September 2005. [arXiv:quant-ph/0512258].