Categorical composable cryptography

Martti Karvonen (joint work with Anne Broadbent)

Structure meets Power Workshop
28.7.2021

Composability in cryptography

One would expect that if you wire together "provably secure" protocols you end up with a secure protocol.

- This is false in general! Standard game-based security notions don't necessarily guarantee composability. In fact, many "secure" protocols might not be secure anymore if several copies are run concurrently.

- QKD and 20(ish) years between first security proofs and composable ones.

- Several frameworks for composability and plenty of work within them, but none have convinced the whole community.

Real-world ideal-world paradigm

AKA simulation paradigm.

Usual definition: a real protocol P securely realizes the ideal functionality F from the resource R if for any attack A on $P \circ R$ there is a simulator S on F such that $(A, P) \circ R$ is indistuingishable from $S \circ F$ by any (efficient) environment.

"Any bad thing that could happen during the protocol could also happen in the ideal world."

This is usually made precise by fixing a concrete (and often low-level) formalism for interactive computation, e.g. the Universal-composability framework of Canetti or the Reactive Simulability-framework by Backes, Pfitzmann & Waidne, both using variants of interactive TMs.

Real-world ideal-world paradigm

In contrast, the Abstract Cryptography approach of Maurer and Renner formalizes the simulation paradim in terms an axiomatization of interacting systems.

The related constructive cryptography of Maurer views a cryptographic protocol P as a *secure construction* of a target functionality from given starting functionalities.

Our development follows these in spirit, but not in technical details.

N+1th approach

In our work we formalize the simulation paradigm over an arbitrary category (and a model of attacks). The main result is that protocols secure against a fixed attack model can be composed sequentially and in parallel. The resulting model is flexible:

- simulation-based security definitions are inherently composable, whether the model of computation is synchronous or not, classical or quantum etc. To model multiparty computation, need only a symmetric monoidal category.
- abstract attack models pave way for other kinds of attackers than malicious ones
- different notions of security (computational, finite-key regimen etc) fit in
- CT and the tools and connections it brings

N+1th approach

Moreover, our approach lets one see existing results from a new viewpoint:

- Under some assumptions, monoidal functors preserve security vs. Unruh's lifting theorem
- existence of initial attacks vs. Canetti's "completeness of the dummy adversary"
- purely pictorial derivations of existing no-go results for two and three parties. moreover, the pictures were already there to "illustrate" the proofs

Cryptography as a resource theory

The key idea is that cryptography is a resource theory: the resources are various functionalities (e.g. keys, channels etc) and transformations are given by protocols that build the target resource *securely* from the starting resources. Instead of an discussing the formalization, we will discuss an extended example - the OTP.Our approach is inspired by

'*Constructive Cryptography – A New Paradigm for Security Definitions and Proofs*'
Maurer, U., TOSCA 2011.

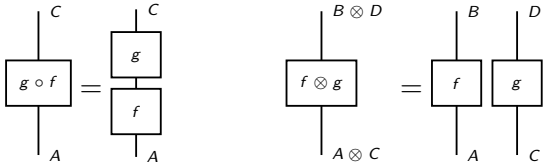
'*Bicategorical Semantics for Nondeterministic Computation*'
Stay, M. & Vicary, J., MFPS 2013.

In this viewpoint, OTP is a protocol: $key \otimes insecure\ channel \to secure\ channel$

Recap on pictures

Let **C** be a symmetric monoidal category — concretely, you can think of (finite) sets and stochastic maps. We will depict a morphism as 

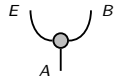
, and composition and monoidal product as



Special morphisms get nicer pictures: identities and symmetries are

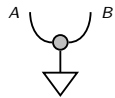
OTP: starting resources

Channel from Alice to Bob that leaks everything to Eve:



(Note: if instead the message goes via Eve (who may tamper with it), the analysis is different)

Shared random key:

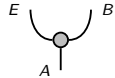


Target resource: a channel

Free building blocks: local (efficient) computation

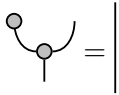
Insecure protocol

One way of transforming



to

is by having Eve delete everything she receives, as



But this is not secure against Eve! No guarantees if Eve disobeys the protocol.

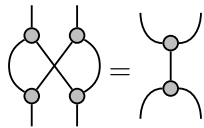
Local ingredients for OTP

A group structure on the message space: a multiplication  with unit  satisfying the following equations.

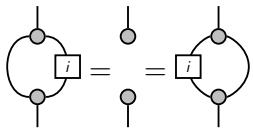
Note that copying and deleting satisfy similar equations

Rest of the group structure

In addition, multiplication and copying interact:

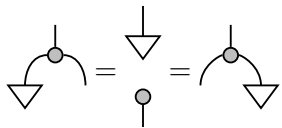


and the map  giving inverses satisfies



Uniform randomness

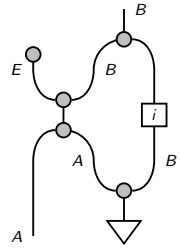
The key being uniformly random is captured by



"Adding uniform noise to a channel gives uniform noise"

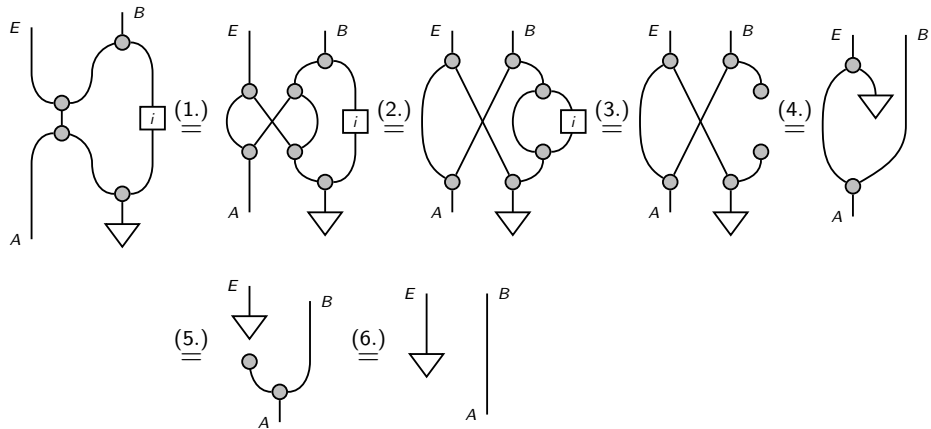
For the experts: a Hopf algebra with an integral in a symmetric monoidal category.

The protocol



Alice adds the key to her message, broadcasts it to Eve and Bob. Eve deletes her part and Bob adds the inverse of the key to recover the message.

Security of OTP



1. Bialgebra. 2. Associativity. 3. Antipode 4. Units 5. Random noise 6. Units.

More on OTP

In other words, anything Eve might learn from the ciphertext she could already compute without it, so this protocol is indeed a secure transformation against Eve.

Reusing keys is not a secure map $key \to key \otimes key$. However, a computationally secure PRNG will give a computationally secure way of constructing a long shared key from a short one. Composing these two results in *the stream cipher*, which is secure automatically as a composite of secure protocols inside our framework.

(Reasoning about computational security amounts to replacing equations with $\approx$ or working with a (pseudo)metric).

Summary

We have a categorical framework where

- composability is guaranteed (also for computational security)
- attack models are general enough to cover various kinds of adversarial behavior (e.g. colluding vs independent attackers)
- string diagrams can be (but don't have to be) used to make existing (or new) pictures into rigorous proofs

Questions...

?

Broadbent A., MK, “Categorical composable cryptography” (2021),arXiv:2105.05949

See also my talk at ACT on July 16th.