

QCrypt: Implementing a Next-Generation Quantum Key Distillation Engine in Practice

P. Junod (HEIG-VD)

Joint work with

A. Burg, J. Constantin (EPFL), Ch. Portmann (ETHZ & Uni. of Geneva)

R. Houlmann, Ch. L. Ci Wen, N. Walenta, H. Zbinden (Uni. of Geneva)

N. Kulesza (ID Quantique SA)



Outline

- 1 Context
- 2 Classical Channel
 - Authentication
 - Error Correction
 - Privacy Amplification
- 3 Overall Security
 - Random Numbers
 - Security Parameter

QCrypt in a Nutshell

- 4-year project funded by the SNF Nano-Tera initiative (2009-2013)
- Researchers from Uni. of Geneva, ETHZ, EPFL, HEIG-VD and ID Quantique SA
- Two different goals:
 - 1 Build a next-generation high-speed QKD engine
 - 2 Build a 100 Gbps (classical) encryption engine

GAP - University of Geneva / ID Quantique SA

■ Pioneers in the domain of **practical** quantum cryptography

REVIEWS OF MODERN PHYSICS, VOLUME 74, JANUARY 2002

Quantum cryptography

Nicolas Gisin, Grégoire Ribordy, Wolfgang Tittel, and Hugo Zbinden

Group of Applied Physics, University of Geneva, 1211 Geneva 4, Switzerland

(Published 8 March 2002)

Quantum cryptography could well be the first application of quantum mechanics at the single-quantum level. The rapid progress in both theory and experiment in recent years is reviewed, with emphasis on open questions and technological issues.

[Quantum cryptography](#)

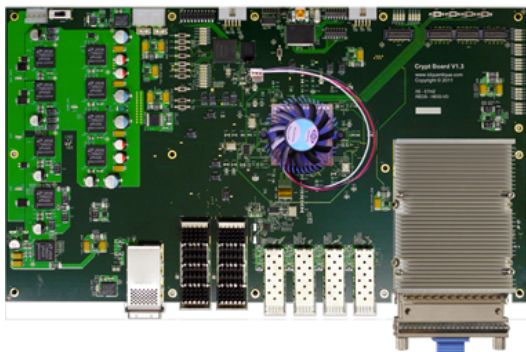
[N Gisin, G Ribordy, W Tittel, H Zbinden - Reviews of modern physics, 2002 - APS](#)

Electrodynamics was discovered and formalized in the 19th century. The 20th century was then profoundly affected by its applications. A similar adventure may be underway for quantum mechanics, discovered and formalized during the last century. Indeed, although ...

[Cited by 3583](#) [Related articles](#) [BL Direct](#) [All 104 versions](#) [Cite](#)

QCrypt - Fast Encryptor

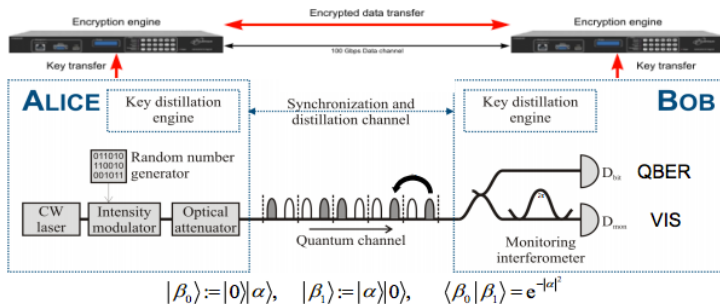
- 10 Ethernet channels of 10 Gbps each
- 100 Gbps layer-2 AES-GCM encryption engine
- 100 Gbps data channel over a single fiber
- (Securely) get keys from the QKD engine





QCrypt - QKD Engine

- Based on the Coherent One-Way (**COW**) Protocol
- Simple data channel with no active elements at Bob
- Interference visibility as measure of Eve's information
- Fast single photon detectors with gate frequencies of up to 2.3 GHz.
- Target throughput for the distilled key: **1 Mbps**



Outline

1 Context

2 Classical Channel

- Authentication
- Error Correction
- Privacy Amplification

3 Overall Security

- Random Numbers
- Security Parameter



Information-Theoretically Secure Authentication

- Like for BB84 and other QKD protocols, one needs to exchange information on a non-confidential, but **authenticated** channel.
- Requirements on the MAC:
 - 1 Information-theoretic security
 - 2 Process blocks of 2^{20} bits
 - 3 Authentication tag of 127 bits

(Strong) Universal Hashing

Definition (Universal Functions)

Let \mathcal{X} and \mathcal{Y} be two finite sets. A family \mathcal{H} of hash functions $h : \mathcal{X} \rightarrow \mathcal{Y}$ is called ε -almost universal if the following condition holds: for any $x \neq x' \in \mathcal{X}$, $\Pr[h(x) = h(x')] \leq \varepsilon$.

Definition (Strongly 2-Universal Functions)

Let \mathcal{X} and \mathcal{Y} be two finite sets. A family \mathcal{H} of hash functions $h : \mathcal{X} \rightarrow \mathcal{Y}$ is called ε -almost strongly 2-universal if the following condition holds: for any $x_1 \neq x_2 \in \mathcal{X}$ and any $y_1, y_2 \in \mathcal{Y}$,

$$\Pr[h(x_1) = y_1, h(x_2) = y_2] \leq \frac{\varepsilon}{|\mathcal{Y}|}$$



(Strong) Universal Hashing

Theorem (Wegman-Carter, 1981 / Stinson, 1991)

Suppose that \mathcal{H} is an ϵ -strongly 2-universal family of hash functions.

Then \mathcal{H} is an information-theoretically secure message

authentication code with $\alpha = \frac{1}{|\mathcal{Y}|}$ and $\beta \leq \epsilon$.

Here, α denotes the **impersonation** probability and β the **substitution** probability.

Towards a Concrete Construction (1)

We consider the two following families of hash functions:

$$\mathcal{H}^{\heartsuit} = \left\{ h_k(\mathbf{x}) = \sum_{i=0}^m x_i k^i : x_i, k \in \text{GF}(2^n) \right\}$$

$$\mathcal{H}^{\spadesuit} = \{ h_{(a,b)}(x) = [ax]_{n-1} + b : a \in \text{GF}(2^n) \text{ and } b \in \text{GF}(2^{n-1}) \}$$

\mathcal{H}^{\heartsuit} is also called **polynomial hashing**.

Theorem (Wegman-Carter, 1979)

\mathcal{H}^{\heartsuit} is a $\frac{m}{2^n}$ -almost universal family of hash functions.

Theorem (Wegman-Carter, 1981)

The set \mathcal{H}^{\spadesuit} is a $\frac{1}{2^{n-1}}$ -almost strongly universal family of hash functions.



Towards a Concrete Construction (2)

Theorem (Stinson, 1994)

Suppose \mathcal{H}_1 is an ε_1 -almost universal family of hash functions mapping \mathcal{X} to \mathcal{Y} and suppose that \mathcal{H}_2 is an ε_2 -almost strongly universal family of hash functions mapping \mathcal{Y} to \mathcal{Z} . Then the composition $\mathcal{H}_2 \circ \mathcal{H}_1$ is an $(\varepsilon_1 + \varepsilon_2)$ -almost strongly universal family of hash functions mapping \mathcal{X} to \mathcal{Z} .

Corollary

Combining the \mathcal{H}^{\heartsuit} and \mathcal{H}^{\spadesuit} families result in a $\frac{m+2}{2^n}$ -almost strongly universal family of hash functions where $\ell = n(m+1)$ is the length in bits of the input message.



Towards a Concrete Construction (3)

- Finite field of size 2^{128}
- Given a message m 128-bit block, one needs $m + 1$ multiplications and $m + 1$ additions in the field
- $3n - 1$ secret key bits are consumed for each block

Implementing Key Reuse

- One can decrease the key bits consumption using the following trick (proposed by Wegman and Carter):
 - Instead of generating a new strongly-universal hash function for each message, generate a single one and keep it secret.
 - Then, encrypt every authentication tag using a one-time pad
- For authenticating t messages n bits each, you need $3n - 1 + t(n - 1)$ bits instead of $t(3n - 1)$.
- Recently shown by Portmann (2012) to be ϵ -**UC-secure**, i.e., the overall authentication error probability will be upper-bounded by $t\epsilon$ for t messages.



Implementing Key Reuse

- Concretely, as we need about $t = 7 \dots 10$ operations of authentications on blocks of 2^{20} bits for distilling 10^5 bits, we get an upper bound on the attack probability in the order of $t \cdot 2^{-114}$ for the authentication part.
- About 2.4% of the distilled key bits will be dedicated to authentication.



Error Correction Engine

- Error correction is comprised of forward error correction followed by a (randomised) integrity verification.
- Implemented through the quasi-cyclic LDPC code defined in IEEE 802.11n.
- Syndrome encoding with a block code length of 1944 bits
- The code rate can be set to $1/2$, $2/3$, $3/4$ or $4/5$ depending on the QBER.



Error Correction Engine

- An integrity check (UHF with collision probability upper bound of 2^{-32}) is required since the error detection capability of the FEC decoding is insufficient to guarantee that all errors will be corrected.
- The integrity check is performed **prior** the privacy amplification (PA) to avoid revealing information to Eve without being able to account it with the PA process.



Privacy Amplification

- The privacy amplification (PA) mechanism is responsible to decrease the information of Eve about the corrected key.
- The PA mechanism uses a fixed compression ratio of 10-to-1.
- It processes input blocks of 10^6 bits and outputs block of 10^5 bits.
- It relies on a universal hash function.

Toeplitz Hashing

- Origin: a construction by Wegman and Carter
- Let M be an $n \times m$ matrix over $\text{GF}(2)$. Then, the mapping $\mathbf{y} = M\mathbf{x}$ is universal.
- However, it would require to transmit $m = 10^{11}$ random bits.
- Mansour et al. (1993) and Krawczyk (1994) showed that restricting the matrix to **Toeplitz** matrices keeps universality, but requires only $n + m - 1$ random bits.

$$T = \begin{pmatrix} t_0 & t_1 & t_2 & \dots & t_{n-2} & t_{n-1} \\ t_{-1} & t_0 & t_1 & \dots & t_{n-3} & t_{n-2} \\ t_{-2} & t_{-1} & t_0 & \dots & t_{n-4} & t_{n-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{-m+2} & t_{-m+3} & t_{-m+4} & \dots & t_{n-m-2} & t_{n-m-1} \\ t_{-m+1} & t_{-m+2} & t_{-m+3} & \dots & t_{n-m-1} & t_{n-m} \end{pmatrix},$$



LFSR Hashing

- Even better: Krawczyk (1994) proposed a construction that requires only $2m$ bits relying on generating the pseudo-random bits using an random LFSR.
- But...
 - This construction is only **almost**-universal, which is not sufficient for PA
 - Generating quickly random irreducible polynomials of degree 10^5 is ... challenging, to say the least!



Back to Toeplitz Hashing

- Eventually, the PA was chosen to be implemented as a Toeplitz matrix-vector multiplication, with help of a shift register.
- It is well known that you can accelerate a Toeplitz matrix-vector multiplication from $O(n^2)$ bits operations down to $O(n \log n)$ using Fast Fourier Transform techniques.
- FFT-like techniques were however abandoned due to **hardware latency** requirements.

Outline

- 1 Context
- 2 Classical Channel
 - Authentication
 - Error Correction
 - Privacy Amplification
- 3 Overall Security
 - Random Numbers
 - Security Parameter



Generation of Random Numbers

- 600 to 800 Mbps of **true** random numbers are required
- Problems:
 - Using several Quantis devices in parallel is too expensive
 - You cannot certify a device according to FIPS 140-2 ... without a deterministic expansion mechanism.
- Chosen solution: couple a Quantis TRNG with an AES-CTR pseudo-random generator, according to NIST SP800-90.
- Costs and business requirements introduce a **computational assumption** in the distillation engine



Target Overall Security Level Definition

- When implementing a QKD protocol in practice, one has to fix security parameters:
 - upper-bound on the remaining information of Eve
 - probability to defeat the authentication mechanism
 - ...
- In a way, one has to define an overall security level, like in classical cryptography, where 100 bits are likely to be secure until 2020-2040, depending on the adversary power.

QKD Overall Security Level

- What is an ε -secure QKD protocol ?
- Asymptotic proof vs. finite-key proof

Tight Finite-Key Analysis for Quantum Cryptography

Marco Tomamichel,^{1,*} Charles Ci Wen Lim,^{2,†} Nicolas Gisin,² and Renato Renner¹

¹*Institute for Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland*

²*Group of Applied Physics, University of Geneva, 1211 Geneva, Switzerland*

Despite enormous progress both in theoretical and experimental quantum cryptography, the security of most current implementations of quantum key distribution is still not established rigorously. One of the main problems is that the security of the final key is highly dependent on the number, M , of signals exchanged between the legitimate parties. While, in any practical implementation, M is limited by the available resources, existing security proofs are often only valid asymptotically for unrealistically large values of M . Here, we demonstrate that this gap between theory and practice can be overcome using a recently developed proof technique based on the uncertainty relation for smooth entropies. Specifically, we consider a family of Bennett-Brassard 1984 quantum key distribution protocols and show that security against general attacks can be guaranteed already for moderate values of M .

- See <http://arxiv.org/abs/1103.4130v1> for the gory quantum details.

QKD Overall Security Level

- Let us denote by \mathbf{S} and $\hat{\mathbf{S}}$ the keys delivered by the QKD protocol on Alice and Bob side, respectively.
- A QKD protocol is called ϵ_{COR} -correct if $\Pr[\mathbf{S} \neq \hat{\mathbf{S}}] \leq \epsilon_{\text{COR}}$.
- A key is called Δ -secret from the eavesdropper Eve if it is Δ -close to a uniformly distributed key that is uncorrelated with the eavesdropper, where

$$\min_{\sigma_E} \frac{1}{2} \|\rho_{SE} - \omega_S \otimes \rho_E\|_1 \leq \Delta$$

where ρ_{SE} denotes the quantum state that describes the correlation between Alice's key and Eve and ω_S is the completely mixed state.

QKD Overall Security Level

- A QKD protocol is ϵ_{sec} -secret if it outputs Δ -secure keys with $(1 - p_{\text{abort}})\Delta \leq \epsilon_{\text{sec}}$, where p_{abort} denotes the probability that the protocol aborts.
- A QKD is called ϵ -secure if it is ϵ_{cor} -correct and ϵ_{sec} -secret with $\epsilon_{\text{cor}} + \epsilon_{\text{sec}} \leq \epsilon$.

Security Specification 1 (Overall QKD Security Level).

The QCrypt QKD engine shall implement an ϵ -secure QKD protocol with $\epsilon \leq \ell \cdot 10^{-11}$ where the QKD protocol outputs an ℓ -bit string.



QKD Overall Security Level

- One can similarly state the average probability of guessing a secret key bit value given the adversary's information as $\frac{1}{2} + \varepsilon/\ell \leq \frac{1}{2} + 10^{-11} \approx \frac{1}{2} + 2^{-36.5}$.



Open Questions

- How do you compare **in practice** this ϵ with the security of a, say, Diffie-Hellman key agreement?
- Does it make sense to compare QKD security and classical security at all?
- What about mixing classical and QKD primitives in the same system?
- Of course, purists will say it's a heresia. But what should practitioners think about this?
- (I let all the aspects of implementation security aside, obviously;-)