Finite-key-size effect in commercial plug-and-play QKD system

Poompong Chaiwongkhot,[1, 2, *] Shihan Sajeed,[1, 3] Lars Lydersen,[4] and Vadim Makarov[1, 2, 3]
[1] *Institute for Quantum Computing, University of Waterloo, Waterloo, ON, N2L 3G1 Canada*
[2] *Department of Physics and Astronomy, University of Waterloo, Waterloo, ON, N2L 3G1 Canada*
[3] *Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2L 3G1 Canada*
[4] *Department of Electronics and Telecommunications,*
Norwegian University of Science and Technology, NO-7491 Trondheim, Norway
(Dated: April 29, 2016)

We demonstrate the ability of an eavesdropper to control the raw-key size in a commercial plug-and-play QKD system Clavis2 from ID Quantique, and its effect on security analysis. Experimentally, we could consistently force the system to generate the key outside of the secure regime. We also test manufacturer's software update that patches this problem.

INTRODUCTION

Quantum key distribution (QK) [1, 2] systems are expected to provide highly secure keys between two parties [3, 4]. To fulfill that expectation, every feature, imperfection, and loophole both in theory and in physical implementation has to be taken into account. One of these features is that, with limited resources and time, a QKD system can exchange a limited length of raw key. In theory, a secret key can be generated if the error rate of the raw key is known [5, 6]. In practice, that error rate is estimated by disclosing a small portion of the raw key. If the raw key sample is finite, estimated variable might deviate from the one of the raw key. Hence, the security of the secret key might be compromised. Finite key size analysis takes those statistical deviation into account and modifies the amount of the secret key generated after privacy amplification. This is done by introducing the security parameter ϵ, which is the probability that non-zero secret key has been generated according to the protocol but the third party may have obtained knowledge [4].

This study is aimed to emphasize the significance of finite-key-size effects on a practical system. The goal is to demonstrate the ability of Eve to force the system to generate a secret key from a raw key size that is smaller than which was predicted in the system design. As a result, the asymptotic limit employed in the system might no longer hold.

EXPERIMENT

The subject of this study is a plug-and-play QKD system Clavis2 produced by ID Quantique. More detail and specifications of the system can be seen in Refs. 7 and 8. The security of this system implemented in the manufacturer's software is based on the security analysis in Ref. 9 which did not considered the finite-key-size effect.

Under normal operation, the system exchanges the quantum signal and saves the raw key until the memory buffers in Alice and Bob of around 2 Mbit are filled. Then, they perform sifting, error correction and privacy amplification [6, 7]. One of the features of Clavis2 is that the system will terminate the raw key exchange process if the photon detection efficiency in the quantum channel drops below a certain value, and perform the post-processing from the raw key already exchanged until then. This feature was implemented to compensate the drift of timing alignment of detector gates [8, 10]. Since the security proof of the system did not take into account the statistical deviation of non-infinite key length, if Eve can force the system to generate secret keys from a shorter raw key length, she can amplify the effect of the statistical deviation and may be able to take advantage of it.

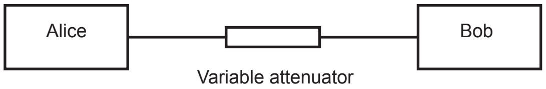

FIG. 1. Scheme of experiment

To demonstrate the ability of Eve to force the system to work with a small key length, we began our experiment by setting the system in a normal operation where the quantum channel of Alice and Bob consisted of a 2 m long optical fiber and a variable attenuator (OZ Optics DD-100-11-1550) simulating transmission loss of a longer line and also giving Eve the ability to control it (see Fig. 1). We ran multiple sessions of key distribution. In each session, the attenuator was set to simulate the normal transmission loss at the beginning of the synchronization phase. We ran three sets of key exchanges with quantum channel transmission loss of 2, 3 and 4 dB. During the raw-key exchange phase, we let the system exchange the raw key for a set period then adjusted the attenuation so that the total loss in the channel suddenly became about 40 dB. This reduced the detection rate in Bob below the threshold and forced the system to terminated the key exchange. After that, the system began the post-processing out of the raw key that had already been exchanged. Then the system reported the distilled

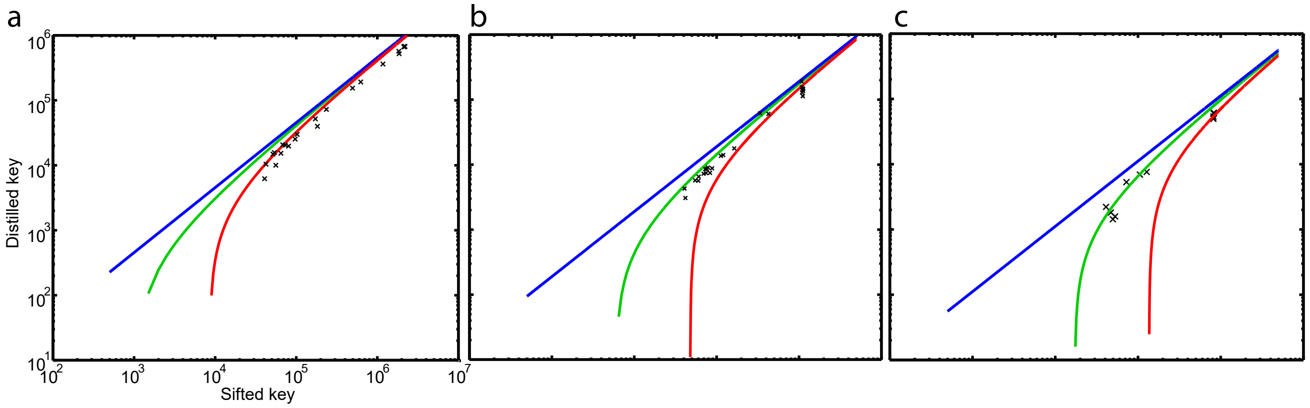

FIG. 2. Secret key rate versus sifted key rate. Black × are experimental results with (a) 2 dB line loss and 3% error rate, (b) 3 dB line loss with 5% error rate, and (c) 4 dB line loss with 6% error rate. Blue (dark grey) line is the infinite key bound. Red (grey) line is finite-key size bound with $\epsilon = 10^{-10}$. Green (light grey) line is finite-key-size bound with $\epsilon = 10^{-1}$. Secure key bounds in each sub-figure were calculated separately according to the error rate and line loss of each experiment.

key length for this session. At the same time, we reset the variable attenuator to the original loss value. The system returned to the synchronization step [8, 10], and began a new session of key exchange. We varied the delay from the start of raw key exchange phase between 10–280 s before applying the 40 dB loss. The delay was picked so that the raw key size before termination was between the system's minimum threshold of 80 kbit and the memory limit of 2 Mbit (leftmost and rightmost data points in each plot in Fig. 2; note that sifted key size plotted is half the raw key size). The amount of the raw key exchanged does not depend solely on the session duration. Some sessions experienced fluctuations in transmission and detection rate, which caused a lower key exchange rate but not below the termination threshold. Some sessions terminated before we induced the loss, if the threshold was crossed before that. In this analysis, we consider the length of distilled key as a function of sifted key instead of session time duration.

For each session with non-zero distilled key, we recorded the length of the sifted key, the number of bits disclosed in the error correction, the error rate, and length of secret key reported by the system. Next step is to verify if this data falls under the theoretical bound.

SECURITY VERIFICATION

We formulated the key rate equation for weak-coherent-state BB84 based on Gottesman-Lo-Lütkenhaus-Preskill (GLLP) security proof [11] which gives the lower bound of secret key rate under asymptotic assumption. For finite key size effect, we used correction terms based on previous analysis on BB84 system [12, 13]. Note that this equation is the secret key length as a function of sifted key and other system-reported-parameters.

$$\begin{aligned} l \quad \leq \quad & nA(1 - H(\frac{E}{A}) - leak_{EC}) \\ & - \frac{1}{2}\sqrt{(\frac{\ln(1/\epsilon_{PE})\ln(n+1)^2)}{n}} \\ & - 7\sqrt{\frac{1}{n}log(\frac{2}{\epsilon'})}) - 2log\frac{1}{\epsilon_{PA}} - log\frac{2}{\epsilon_{EC}}, \end{aligned} \tag{1}$$

where n is sifted-key size, E is error rate per sifted key reported by the system, $leak_{EC} = 1.2h(E)$ is the estimated quantity of bits disclosed in error correction where $h(E)$ is Shannon limit of error correction and factor 1.2 is the efficiency of the error correction protocol [5]. $A = \frac{(p_{det} - p_{multi})}{p_{det}}$ is a correction term where p_{det} is the probability of detection and p_{multi} is the probability of multi-photon pulse generated by Alice. In this analysis, this term is calculated using Poisson distribution and average mean photon number per pulse sent by Alice of 0.02. The last four terms are correction terms due to finite key statistics. The first correction term is the result of statistical deviation in parameter estimation step where ϵ_{PE} is the probability that the key has more errors than what was estimated in parameter estimation step. The second takes account of statistical approximation in privacy amplification step, where ϵ' is the probability of failure. The third term is for the probability, ϵ_{PA}, that the hash function transforms two different key sequences into the same final key. The last term takes account of failure probability, ϵ_{EC}, of error correction where there is non-zero error bit left after the correction. The security parameter $\epsilon = \epsilon_{PE} + \epsilon' + \epsilon_{PA} + \epsilon_{EC}$ [4, 12, 13].

After substituting the parameters from the experiment into Eq. (1), we obtained a lower bound of secure key length as shown in Fig. 2. The blue (dark grey) line was calculated under the asymptotic assumption as used in the system's protocol. The other two lines are the bound

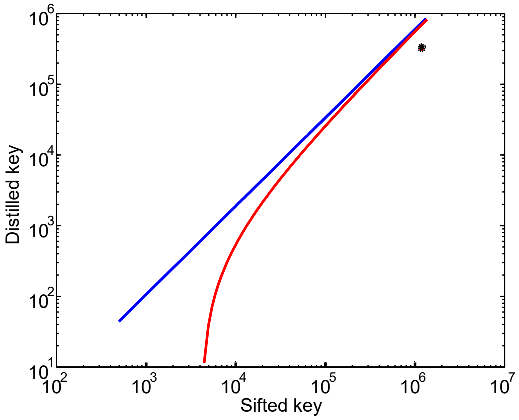

FIG. 3. Experimental result with new software. The line loss was 3 dB and error rate was 1%. Blue (dark grey) line is the infinite key bound. Red (grey) line is finite-key size bound with $\epsilon = 10^{-10}$. A group of × presents results of 8 key distillations. Regardless of our interruptions, the system retained the raw key exchanged before termination of each raw key exchange session, and accumulated it until the size reached about 2 Mbit before proceeding to distillation.

of secret key rate under the finite-key size assumption (the area below each line gives the secure zone corresponding to the security conditions applied to that plot). It can be seen from Fig. 2 that the experimentally distilled key sizes, denoted black ×, satisfied the security criteria for the asymptotic assumption. However, the experimental results fall out of bound of finite-key size analysis with values of ϵ up to 10^{-1}. This means that there is a 10% chance that the information of the key generated under these condition might be leaked to Eve. In practice, the security parameter ϵ can be picked to be of the same order as major natural disasters such as a serious earthquake, volcanic eruption or nuclear power plant meltdown [14]. If such disaster happened, it is most likely that the security of the key would not matter anymore. For example, the probability of a nuclear power plant meltdown is 10^{-4} per year, according to the Nuclear Regulatory Commission. If our QKD machine generates two keys every minute or approximately a million keys a year, one might pick $\epsilon = 10^{-10}$ so that the probability that at least one key leaks to Eve is of the same order as such disasters [4].

In the middle of our study in 2014, ID Quantique released a new software patch for Clavis2. This patch accumulates the key if the key exchange session is terminated and lets the system perform post-processing only when the raw key size exceeds a threshold of around 2 Mbit. We performed our experiment and recalculated our plot using the new parameters acquired from the system. The result showed that the distilled key is within the secure bound of $\epsilon = 10^{-10}$ (see Fig. 3).

CONCLUSION

We have shown that, by dynamically controlling quantum channel loss, Eve can force the commercial QKD system Clavis2 to distill a secret key from a shorter raw key length. We have shown that the key generated from a sufficiently low raw key size was not guaranteed secure by the proofs with finite-key-size analysis. We have also investigated the security update from ID Quantique, and found that the key generated by the new software is secure under finite-key-size analysis. Our study only covers statistical evidence from the system against the theoretical bound. An explicit attack that exploits this effect is still open for future study. Our investigation highlights the significance of finite-key-size analysis and why this effect should be included in the implementations of QKD, especially in commercial systems.

We thank N. Lütkenhaus, R. Renner, and J. Skaar for discussions. We thank ID Quantique for cooperation, technical assistance, and providing us the QKD hardware. This work was supported by Industry Canada, NSERC, CFI and Ontario MRI. P.C. and S.S. acknowledge support from CryptoWorks21. P.C. acknowledges support by Thai DPST scholarship.

* poompong.ch@gmail.com

[1] C. H. Bennett and G. Brassard, in *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing* (IEEE Press, New York, Bangalore, India, 1984) pp. 175–179.

[2] A. K. Ekert, Phys. Rev. Lett. **67**, 661 (1991).

[3] N. Lütkenhaus, Phys. Rev. A **61**, 052304 (2000).

[4] R. Renner, N. Gisin, and B. Kraus, Phys. Rev. A **72**, 012332 (2005).

[5] G. Brassard and L. Salvail, Lect. Notes Comp. Sci. **765**, 410 (1994).

[6] C. H. Bennett, F. Bessette, L. Salvail, G. Brassard, and J. Smolin, J. Cryptology **5**, 3 (1992).

[7] D. Stucki, N. Gisin, O. Guinnard, G. Ribordy, and H. Zbinden, New J. Phys. **4**, 41 (2002).

[8] Clavis2 specification sheet, `http://www.idquantique.com/images/stories/PDF/clavis2-quantum-key-distribution/clavis2-specs.pdf`, visited 25 May 2015.

[9] A. Niederberger, V. Scarani, and N. Gisin, Phys. Rev. A **71**, 042316 (2005).

[10] N. Jain, C. Wittmann, L. Lydersen, C. Wiechers, D. Elser, C. Marquardt, V. Makarov, and G. Leuchs, Phys. Rev. Lett. **107**, 110501 (2011).

[11] D. Gottesman, H.-K. Lo, N. Lütkenhaus, and J. Preskill, Quant. Inf. Comp. **4**, 325 (2004).

[12] R. Y. Q. Cai and V. Scarani, New J. Phys. **11**, 045024 (2009).

[13] M. Tomamichel, C. C. W. Lim, N. Gisin, and R. Renner, Nat. Commun. **3**, 634 (2012).

[14] R. Renner, private communication and lectures (2014).