Editors' Suggestion

Deep-learning-based radio-frequency side-channel attack on quantum key distribution

Adomas Baliuka,[1,2] Markus Stöcker,[1,2] Michael Auer,[1,2,3] Peter Freiwang,[1,2]
Harald Weinfurter,[1,2,4,5,*] and Lukas Knips[1,2,4,†]

[1] *Fakultät für Physik, Ludwig-Maximilians-Universität, 80799 Munich, Germany*
[2] *Munich Center for Quantum Science and Technology, 80799 Munich, Germany*
[3] *Universität der Bundeswehr, 85577 Neubiberg, Germany*
[4] *Max-Planck-Institut für Quantenoptik, 85748 Garching, Germany*
[5] *Institute of Theoretical Physics and Astrophysics, Faculty of Mathematics, Physics, and Informatics, University of Gdańsk, 80-308 Gdańsk, Poland*

(Received 2 August 2023; accepted 18 October 2023; published 20 November 2023)

Quantum key distribution (QKD) protocols have been proven to be secure on the basis of fundamental physical laws; however, the proofs consider a well-defined setting and encoding of the sent quantum signals only. Side channels, where the encoded quantum state is correlated with properties of other degrees of freedom of the quantum channel, allow an eavesdropper to obtain information unnoticeably, as demonstrated in a number of hacking attacks on the quantum channel. However, also classical radiation emitted by the devices may be correlated, leaking information on the potential key, especially when combined with novel data-analysis methods. We demonstrate here a side-channel attack using a deep convolutional neural network to analyze the recorded classical, radio-frequency electromagnetic emissions. Even at a distance of a few centimeters from the electronics of a QKD sender containing frequently used electronic components, we are able to recover virtually all information about the secret key. However, as shown here, countermeasures can enable a significant reduction of both the emissions and the amount of secret-key information leaked to the attacker. Our analysis methods are independent of the actual device and thus provide a starting point for assessing the presence of classical side channels in QKD devices.

DOI: 10.1103/PhysRevApplied.20.054040

I. INTRODUCTION

Quantum key distribution (QKD) [1–6] is one of the most-mature quantum technologies. It allows two authenticated parties to use a quantum channel to exchange a cryptographic secret, which they can later use for symmetric cryptography. Using fundamental physical principles, QKD allows one to quantify the amount of information leakage to an eavesdropper and to subsequently eliminate it entirely using appropriate postprocessing. QKD is used for both short-distance and long-distance communication via free-space [7–11] and fiber-based [12–15] links with first or planned implementations in large networks [16,17]. With a plethora of test beds and implementations in multinational industry-oriented consortia, QKD has reached commercial end-user availability.

However, despite its conceptual elegance, the practical security hinges on the quality of the implementation, in particular, strict adherence to the theoretical model used to prove security. The quantum states sent over the quantum channel have to be prepared precisely within the requirements of the QKD protocol. Any correlation with any other degree of freedom, but also with classical properties of the devices used, potentially opens side channels [5,6,18,19]. These will allow an eavesdropper to infer the key by measurements unnoticeable to the users. We refer to side channels exploited by interacting with the quantum channel as "quantum side channels" and all other side channels as "classical side channels."

Electronic devices continually emit electromagnetic radiation and are in turn influenced by it. Thus, the operation of security-critical devices may be influenced by *active attacks* [20] rendering them insecure, such as demonstrated on quantum random number generators [21]. However, if emissions from a device are correlated with sensitive information processed by it, a critical side

[*] h.w@lmu.de
[†] lukas.knips@mpq.mpg.de

channel opens up widely and allows much-simpler *passive attacks*. They do not need any manipulation of device components and are practically impossible to detect. Investigations of information leakage from conventional communication systems via electromagnetic radiation go back to at least the 1940s, later under the U.S.-military code name TEMPEST [22]. TEMPEST attacks now refer to eavesdropping via electromagnetic or acoustic side channels and are widely considered in security specifications and during certification of security-critical systems. Technologies such as software-defined radio, specialized probes [23], and particularly deep learning [24,25] make the exploitation of vulnerabilities much easier and more effective.

In reaction to quantum hacking attacks on QKD devices, countermeasures have been developed to protect against side-channel attacks on the quantum channel [6]. However, as standard electronic components, especially logic units such as field-programmable gate arrays (FPGAs), application-specific integrated circuits, or CPUs, emit electromagnetic radiation, they also open a new, classical side channel for attacks on potentially every QKD system.

Here we demonstrate a deep-learning-based side-channel attack on a QKD device using radio-frequency (rf) emissions at frequencies up to a few gigahertz. Our setup does not require expensive specialized equipment and works with few computational resources. In some scenarios, our attack is able to recover virtually all information about the secret key. In contrast to a recent attack on QKD single-photon-detector electronics [26], our attack targets the control electronics in general. We demonstrate its power and security threat on a QKD sender module. We analyze how the information leakage depends on the distance to the device, as well as on to what extent it can be mitigated with improved design and shielding. The QKD sender electronics inspected here is homebuilt, but since it is made from conventional electronic components also used in other QKD systems, it is representative of other, also commercial systems. In addition, our data evaluation may also be applied to attack via other weak points, for example, power consumption [27] or acoustic side channels [28]. This clearly demonstrates that a detailed examination of classical side channels is important for future QKD devices and networks. Emission security should be considered from the early design stages [29] until the deployment of QKD devices.

II. EXPERIMENTAL SETUP AND DATA COLLECTION

A. Sender module and attacker setup

The sender module is a homebuilt BB84 polarization-encoding decoy-capable QKD device building upon the device presented in Ref. [9]. It features an FPGA, which controls four distinct vertical-cavity-surface-emitting-laser (VCSEL) drivers [30–32]. The drivers are connected to four VCSELs emitting short light pulses with a wavelength of around 850 nm, which are subsequently polarized by differently rotated polarization filters. In this way, the sender device can emit optical pulses with any of the four polarization directions [horizontal (H), vertical (V), diagonal (P), and antidiagonal (M)]. For the measurements presented here, the module sends random streams of *symbols* (H, V, P, M) at a symbol rate f_{clock} of 100 MHz.

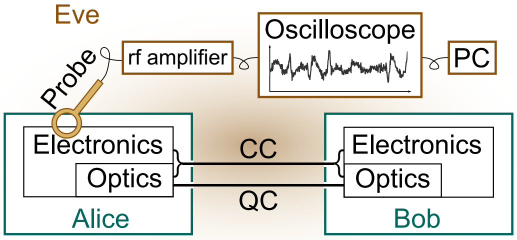

FIG. 1. Sender (Alice) and receiver (Bob) devices, comprising electronics and optics, are connected via a quantum channel (QC) and a classical channel (CC). The eavesdropper (Eve) has access to both channels. Eve measures Alice's emissions using a near-field probe for magnetic fields or a log-periodic antenna (not shown) for far-field measurements, whose rf signal is amplified, captured by an oscilloscope, and evaluated on a personal computer (PC) [30].

For our attack, we record electromagnetic near-field emissions from the printed circuit board of the QKD sender using a magnetic near-field probe, an rf amplifier, and an oscilloscope [30] with a bandwidth of 8 GHz; see Fig. 1. Given the symbol rate of 100 MHz, we sample the emission signal at $f_{\text{samp}} = 10$ GSa/s to obtain 100 voltage samples per symbol. The oscilloscope memory sets the length of the total time trace N_{meas} to 2 MS (Mega Samples, 1 million samples), corresponding to a measurement duration t_{seq} of 200 μs. We hence have sequences of length $N_{\text{seq}} = t_{\text{seq}} \times f_{\text{clock}} = 20\,000$ symbols sent by the sender. Besides near-field emissions, we also record far-field emissions using a commercially available directed wideband log-periodic dipole antenna.

B. Near-field spectrum

The spectrum of the recorded emissions contains a significant signal at the clock frequency f_{clock}; see Fig. 2. Because the measurement occurs in a noisy office environment, emissions in communication bands (e.g., Wi-Fi and Universal Mobile Telecommunications System) contribute to the measured signal. Since the deep-learning methods used in our attack deal well with such noisy signals, we make no attempt to remove the background noise and use no manual filtering in addition to what naturally occurs

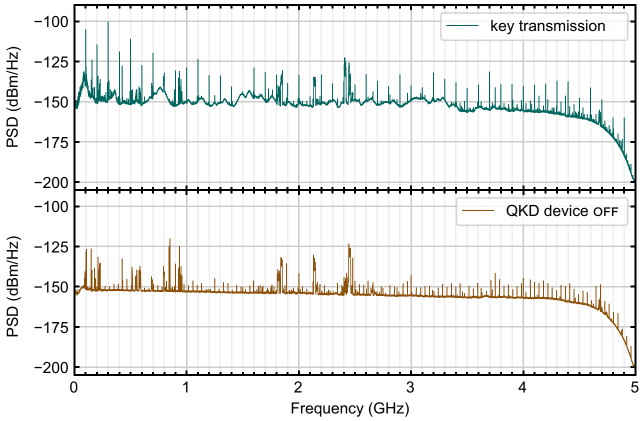

FIG. 2. Near-field spectra [power spectral density (PSD)] of the emissions during a key transmission (green) and during a null measurement (brown) where the QKD sender is not powered. The regular spikes on the fine grid in the upper plot are harmonics of the 100-MHz clock frequency. Frequencies higher than about 5 GHz are suppressed due to the limited bandwidths of the probe, amplifier, and oscilloscope. The spectra are obtained by Barlett's method (segment length 100 000 samples) and averaged over 30 independent measurements. At some frequencies the background exceeds the signal due to the noisy office environment changing in time.

in the measurement devices (probe, amplifier, and oscilloscope). Our method clearly has the advantage that it can cope with standard environments typical, for example, of server rooms. This makes the attack scenario more realistic and enables device evaluation without the need for specialized shielded facilities.

C. Single and averaged time traces

We first examine the data in the time domain. For that, we measure a time trace using the near-field probe while the device is repeating a fixed pseudorandom sequence of 20 000 symbols. Time synchronization between symbols in the key and the measured time trace is achieved with use of the phase of the clock signal, which is digitally extracted from the emissions, as well as a separately recorded trigger signal signifying the time of the first symbol in the key. We verified that the measurement of the trigger signal does not influence the performance of our attack; see Appendix A.

The near-field probe signal is split into snippets with a length corresponding to a few symbols. Since the electronic processes that produce the symbol, especially in the FPGA, take several clock cycles, we thereby make sure we capture all relevant information. This yields a set of snippets of the time trace together with the respective subsequence of the key. Here, for illustration, we choose a snippet length of seven symbols.

To gain more insight, consider, for example, the subsequences matching the pattern "??VXV??," where "?" can be any symbol. We group them according to the center symbol "X" and show their respective snippets in Fig. 3. Precise features of these individual time traces are difficult to identify, and finding the symbol sequence corresponding to a given time trace with the naked eye seems hard. However, a first view of the raw data reveals common features and suggests that changes and specific patterns in the measured magnetic field amplitudes correspond to switching between different adjacent symbols, rather than just the symbols themselves. When the signals corresponding to the same subsequence are averaged (Fig. 3, bottom), the four different averages differ significantly more around the varying center symbol than at the outermost symbols, where they roughly reproduce the clock signal.

III. MACHINE-LEARNING-BASED ATTACK

Conventional methods to extract confidential information from such emission data require both specialized knowledge of signal processing and detailed models of the emissions, limiting the relevance of the resulting attacks to certain domains of application and specific types of devices or electronic components. In contrast, when machine-learning techniques are used [33–35], there is no need to understand how the emissions arise. Rather, an effective statistical model of the phenomena is created from recorded data by a training procedure, making the approach more general and adaptable. Since we are able to collect training data in a known, controlled environment, we apply *supervised learning*, as opposed to, for example, *unsupervised learning* [36] or *reinforcement learning* [37], which may be promising for other types of attack.

The attacker's task is mapping a one-dimensional time series (the recorded emissions) to a sequence of symbols (the raw key). To do this, general sequence-to-sequence methods could be used, such as transformer neural networks [38]. However, the task can be simplified further by assuming that there are no significant long-time correlations between electromagnetic emissions and the symbol sequence, i.e., that the emissions at a given time depend only on the symbols currently being processed but not on all symbols processed at earlier times. Note that the presence of such correlations in the behavior of the electronics could indicate serious security problems related to the QKD device [39,40].

With this assumption, it suffices to be able to map a short snippet of the time trace to the symbol sent during that time. Application of the mapping individually to each snippet then yields the entire key. The attack thus becomes a classification task, i.e., mapping a one-dimensional fixed-length time series to one of four classes (H, V, P, M). To do this, we design and train a convolutional neural network. A snippet length of 500 samples (corresponding to five symbols) as input to the neural network proved sufficient for our attack. For prediction of the center symbol, this length ensures that all relevant information is contained

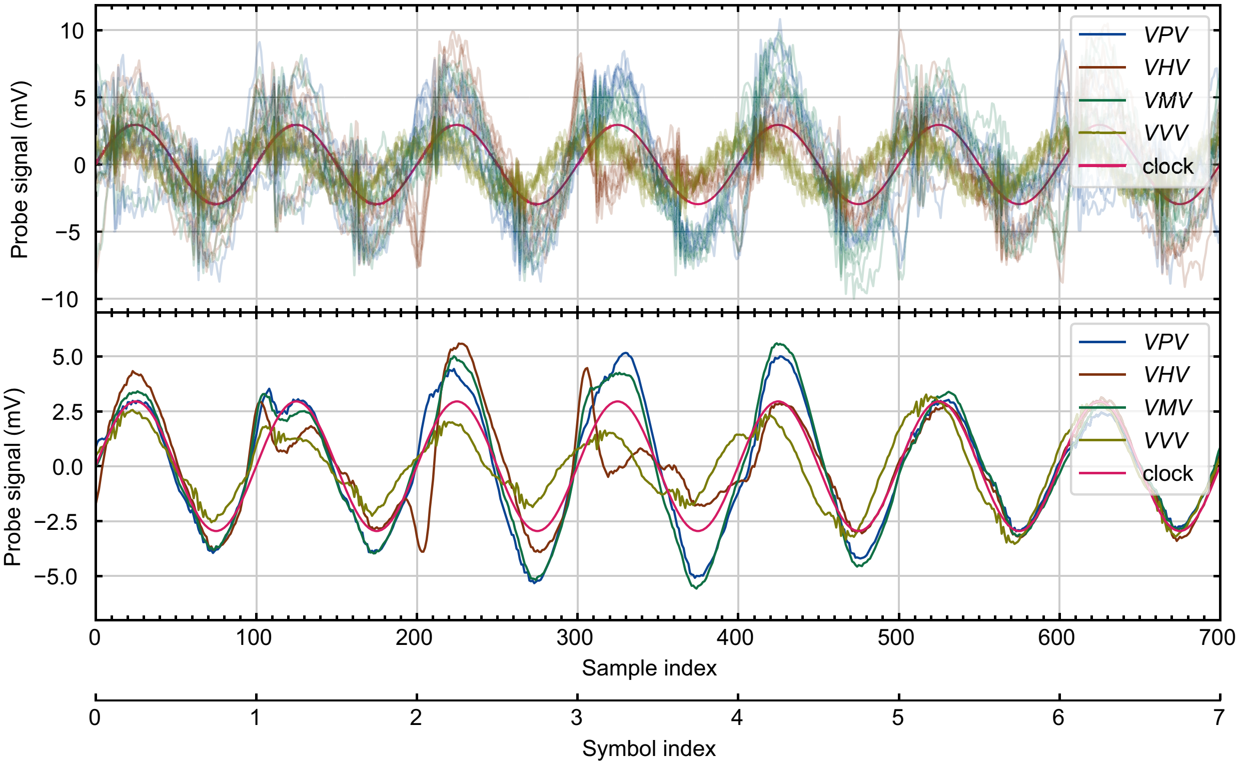

FIG. 3. Top: For each of the three-symbol key excerpts (*VHV*, *VVV*, *VPV*, or *VMV*), nonoverlapping snippets of the recorded time trace (for clarity, only seven per excerpt). The range between sample index 200 and sample index 500 corresponds to the three symbols in the key excerpt. The regions before and after correspond to random symbols that happened to be adjacent to the selected occurrences of the excerpts in the key. For reference, the 100-MHz clock signal is shown, as obtained digitally from the probe signal with use of a band-pass filter. Bottom: Averages of all matching snippets (about 300 each) for each symbol combination across one measurement. In the regions where random symbols contribute to the average (roughly sample ranges 0–200 and 500–700), the differences cancel and the result is close to the clock signal.

in the snippet, while lowering demands on the precision needed in synchronizing the time trace with the symbol sequence. The attack also works with shorter snippet lengths but performs slightly worse.

A. Attacker model

Our experimental design and data evaluation are motivated by so-called *profiled attacks* [41]. For such attacks, the attacker prepares for the actual attack while having full access to a copy of the victim's device. This assumption is in accordance with Kerckhoffs's principle [42] that security of a system should not depend on the secrecy of its design, and is thus appropriate for QKD devices.

A profiled attack consists of two phases. First, in the so-called *profiling*, or *training*, phase, the attacker uses the copy of the target device and records data corresponding to known symbol sequences chosen at will. The data are used to create a model that captures the correlations between secret information and side channels. In our case, this so-called *training dataset* consists of a sequence of key symbols, say, $\left(y_1^{\text{train}}, y_2^{\text{train}}, \ldots, y_{N_{\text{train}}}^{\text{train}}\right)$, and a sequence of time-trace snippets, say, $\left(x_1^{\text{train}}, x_2^{\text{train}}, \ldots, x_{N_{\text{train}}}^{\text{train}}\right)$, where y_i^{train} is the symbol sent during the middle of the snippet x_i^{train}.

The neural network f_θ, given trainable parameters θ, maps any snippet x to the network's *prediction* $f_\theta(x) \in \{H, V, P, M\}$. The training dataset is used to obtain optimized parameters $\tilde{\theta}$ such that the model's predictions $(f_{\tilde{\theta}}(x_1^{\text{train}}), f_{\tilde{\theta}}(x_2^{\text{train}}), \ldots, f_{\tilde{\theta}}(x_{N_{\text{train}}}^{\text{train}}))$ approximate the true key symbols $\left(y_1^{\text{train}}, y_2^{\text{train}}, \ldots, y_{N_{\text{train}}}^{\text{train}}\right)$.

In the second, so-called *attack*, or *test*, phase of the profiled attack, the attacker performs a measurement on the victim's device during its normal operation, i.e., where the attacker has no control or access to any information except the recorded emissions. The attacker records a *test dataset* comprising recorded emissions only, say, $(x_1^{\text{test}}, x_2^{\text{test}}, \ldots, x_{N_{\text{test}}}^{\text{test}})$, and obtains an estimate of the key as the predictions of the previously trained model.

To evaluate the success of the attack, we make use of our access to the true sequence of sent symbols $(y_1^{\text{test}}, y_2^{\text{test}}, \ldots, y_{N_{\text{test}}}^{\text{test}})$ and compare it with the predictions of the fully trained network $(f_{\tilde{\theta}}(x_1^{\text{test}}), f_{\tilde{\theta}}(x_2^{\text{test}}), \ldots, f_{\tilde{\theta}}(x_{N_{\text{test}}}^{\text{test}}))$

by defining the *prediction accuracy* of the test dataset, or simply the *test accuracy*

$$A = \frac{N_{\text{correct}}}{N_{\text{test}}}, \tag{1}$$

where $N_{\text{correct}} = \left|\{i : f_{\tilde{\theta}}(x_i^{\text{test}}) = y_i^{\text{test}}\}\right|$ is the number of symbols correctly predicted by the neural network. Since the four unique symbols are equally likely to occur in the key, random guessing gives a prediction accuracy of 25%. We consider an attack successful (in extracting above-zero information about the key) if the test accuracy exceeds random guessing by more than three standard deviations of the binomial distribution. For a success probability of 25% and 20 000 trials, this implies accuracies should be above 25.92%.

We use prediction accuracy because it is intuitive and allows us to evaluate the attack on the sender module alone without having to discuss sifting or basis choice. A high prediction accuracy implies a successful attack. However, prediction accuracy does not directly correspond to the amount of secret information gained by the attacker. Even a small but above-random prediction accuracy may still allow a critical attack (see Appendix B). How the raw-key-symbol prediction accuracy relates to information leakage is further discussed in Appendix C.

Note that for a QKD device normal operation implies that the secret key is used only once. This means that the attacker has access to only a single time trace of the emissions to extract information about the key. While this makes the attacker's task more difficult, we adhere to this restriction, resulting in a *single-trace attack*.

B. Neural-network architecture and training

Our neural-network architecture consists of fully connected layers, one-dimensional convolutions, max pooling, and batch normalization to process traces of the emissions in the time domain. To make better use of the data, we use data augmentation. For more details on the architecture of the network and the data augmentation, see Appendix D.

Training state-of-the-art neural networks can require very large datasets and is typically performed on graphics processing units (GPUs) or tensor processing units due to the large amount of computational resources required. Since our model is rather small and operates on one-dimensional data, training on a standard laptop with GPU support takes only a few minutes. As moving the probe to a new location and performing the measurement takes only a few seconds, our method allows one to identify vulnerable components almost in real time.

IV. RESULTS

A. Near-field measurements

We perform the measurement procedure described in Sec. II A for various locations of the magnetic near-field probe and collect independent datasets for each location as described in Sec. III A. Two raw keys (each of length 20 000) are created by a pseudorandom number generator with different seeds, such that all four symbols are equally likely. One of them is used for the training, and the other one is used for the test measurements. This is crucial to avoid overfitting [43] and misinterpretation of results.

Although a single time trace, i.e., the emissions recorded while 20 000 symbols are sent, is sufficient to demonstrate information leakage, we combine several measurements to increase the amount of training data and thus improve attack performance. By our trading off longer measurement time for better attack performance (see Appendix D), each training dataset contains snippets obtained from seven combined time traces (equivalent to a symbol sequence of 140 000 symbols), unless indicated otherwise. To monitor how much the results depend on unrelated classical communication and background fluctuations in the noisy office environment, we also record each test dataset three times. The test datasets are not combined but instead are evaluated separately and independently, thus meeting the requirements for a single-trace attack.

As the first step, we investigate which components and areas of the circuit board contribute to rf emissions or leak information about the key. We measure at locations given by a two-dimensional grid spaced at 10 mm in the x and y directions while keeping the magnetic near-field probe at a fixed distance of 10 mm from the board. As shown in Fig. 5(a), especially measurements close to the FPGA allow us to retrieve the key with high accuracy. Other components, such as the voltage regulator, also produce significant rf emissions, which, however, are not correlated with the symbols and effectively reduce the attacker's signal-to-noise ratio, leading to lower test accuracy at those locations. On the other hand, we obtain high test accuracies also in regions with small amplitudes of recorded emissions [Fig. 5(b)]. This shows that the test accuracy cannot be inferred from the amplitude of recorded emissions.

As the second step, we investigate how the distance of the probe from the circuit board affects the accuracy of our attack. We select a location (see Fig. 5) close to the FPGA that promises a successful attack. Positioning the probe above this location at various distances from the board, we observe a decrease of the test accuracy with increased distance as shown in Fig. 4. However, the accuracy is above the random-guessing value up to a distance of 8 cm.

B. Far-field measurements

At distances greater than a few centimeters, the near-field probe is no longer effective. To investigate whether emissions can still be detected at very long distances, we use a log-periodic dipole antenna [30]. Because of a high level of background noise in the environment and non-ideal antenna characteristics, we are not able to extract

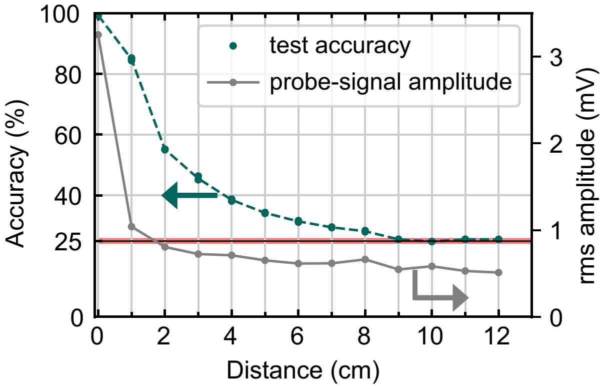

FIG. 4. Effect of varying the distance from the circuit board at a location above the FPGA, which promises high accuracy as indicated in Fig. 5(a). The test accuracy is shown as the average (dotted green line) of three independent attacks (green dots) at each distance. For short distances, the test accuracy is remarkably high (about 99%). The baseline is 25%, corresponding to randomly guessing one of the four symbols. The red area indicates three standard deviations around random guessing, assuming 20 000 trials of a 25% Bernoulli distribution. The rms amplitude of the recorded emissions is shown for reference.

key symbols using the neural network. Thus, we pursue the more-modest goal of investigating if any emissions are present and whether they contain nonzero information about the operation of the QKD device. To demonstrate nonzero information, it is sufficient to use emissions to reliably and consistently distinguish two modes of operation of the device. To show this, we record about 500 emission spectra of our unshielded sender device at a distance of about 2.5 m for two different modes of operation of the QKD sender: sending a random key (“key”), or being turned on but idle (“no key”). To exclude the influence of background variations in time, the two modes are alternated many times during data collection within datasets. By studying the spectra of a training dataset (396 spectra), we manually identify a spectral region with a peak that seems highly correlated with the mode of operation (see Fig. 6). With use of the selected frequency interval around 1.7 GHz, different machine-learning approaches (support-vector classification, K-neighbors classification, or linear discriminant analysis) can clearly distinguish those modes of operation (test accuracy of 100% for a test dataset of 94 spectra).

Although we are not able to reconstruct the key with this equipment and analysis, this result indicates the possibility of information leakage also over longer distances [44].

V. COUNTERMEASURES

There are numerous design and shielding techniques for reducing emissions and preventing information leakage via rf emissions [29,45,46]. The countermeasures described in the following paragraphs significantly reduced emissions

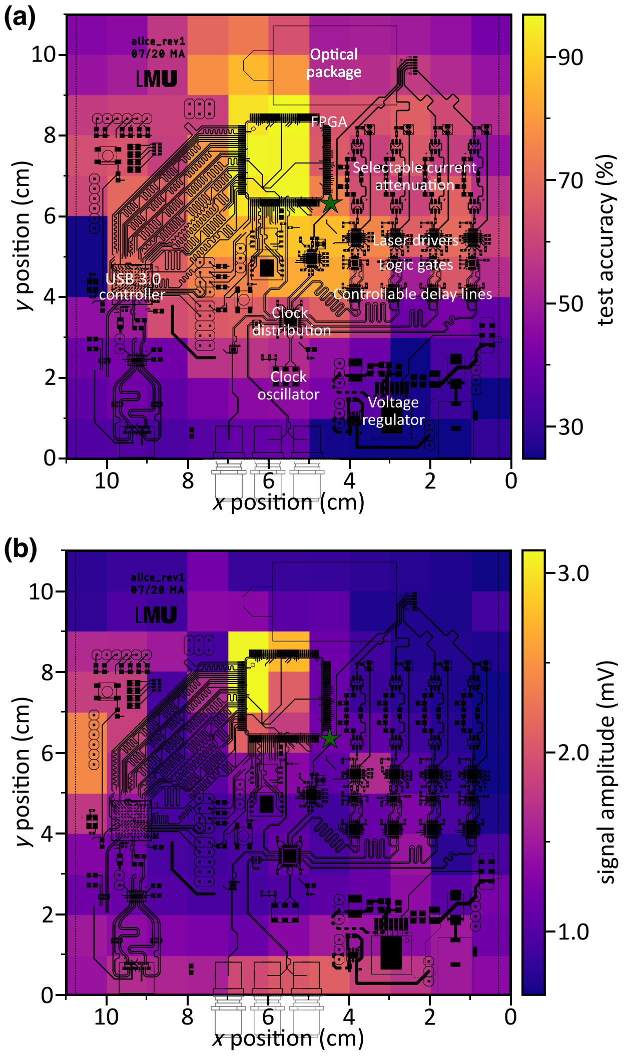

FIG. 5. (a) Test accuracy of our neural network when trained and tested at respective positions and (b) rms amplitude of recorded emissions. Note that both power and accuracy also depend on the angular orientation of the near-field probe, i.e., its rotation about the axis perpendicular to the board, which has been kept fixed. A green star indicates the location used for the distance measurement (Fig. 4).

from a revised version of our electronics, thus making the attack much less effective (Fig. 7), resulting in the attack being no longer successful for distances greater than 5 cm (compared with 9 cm for the former revision as shown in Fig. 4).

An FPGA in a ball-grid array footprint was chosen with proper care of differential signal routing, grounding, and placing of decoupling capacitors. Critical signals were routed in layers shielded by a ground and a supply-voltage plane. Optimization of the FPGA design has not been done, but could further lower the emissions.

With the addition of metallic shielding with a thickness of a few millimeters, our attack could no longer perform

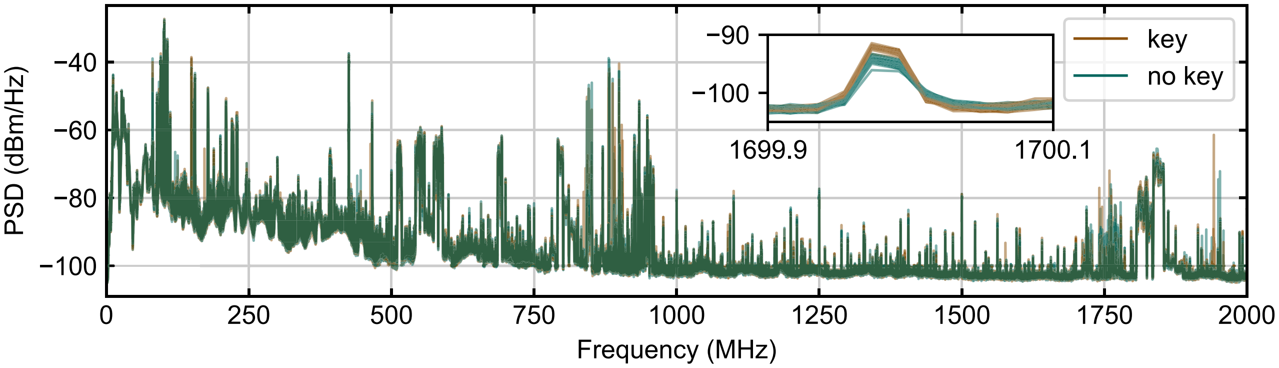

FIG. 6. Measurements with an antenna at a distance of about 2.5 m. The spectra of 30 measurement runs when Alice is sending a random key (brown) and not sending a key (green) are clearly distinguishable as shown in the selected signal range around 1.7 GHz (inset). This is even despite various strong noise contributions from Wi-Fi, General Packet Radio Service, Universal Mobile Telecommunications System, Bluetooth, etc., signals. PSD, power spectral density.

better than random guessing. However, for a QKD device, an optical channel design with a large puncture of the shielding could significantly reduce the shielding effectiveness [47]. We were able to detect a small amount of emissions in front of a hole in the shielding (about 2 × 2 cm^2), which allowed us to predict key symbols with a test accuracy of more than 27% (highly significant for a key length of 20 000 symbols). Also, metallic shielding does not help against low-frequency magnetic fields [48].

VI. CONCLUSION AND OUTLOOK

We have demonstrated that an eavesdropping attack analyzing the electromagnetic emissions from the QKD sender using machine learning can retrieve all information about the key. Although we focused our analysis on rf emissions, our method and machine-learning techniques can also be used for studying information leakage via other potential side channels. As shown, countermeasures can reduce the success rate of the attack; however, they may be difficult to implement, especially if standard electronic components turn out to be the strongest source of information leakage. Even small changes in device design or operation environment can have a large effect. Since countermeasures are much easier to plan and implement in early stages of the design of devices, preliminary testing of emissions can be very valuable.

We emphasize the need to test QKD devices and examine information leakage not only via attacks on the quantum channel but also via classical side channels (e.g., electromagnetic emissions, acoustic vibrations, classical message timing, and power consumption). The method introduced here may serve as a starting point for precompliance testing and for preparation for security certification.

The source code used for data evaluation is available from Ref. [49] (MIT license). It includes the measurement pipeline (remote controlling of the oscilloscope), the neural network, the hyperparameter-optimization routine, the training pipeline, and graphing of results. The measured data as recorded by the oscilloscope are available from Ref. [50]. The data and software provided allow one to reproduce all reported results and may enable further work towards enhanced attacks via, for example, improvements of the neural network.

ACKNOWLEDGMENTS

We are grateful to Wenjamin Rosenfeld and Margarida Pereira for helpful discussions. This work was supported by the DFG under Germany’s Excellence Strategy EXC-2111 390814868, the European project OpenQKD, the BMBF projects DE-QOR and QUBE-II, and the QuNET+ projects SKALE and MiQuE. A.B. acknowledges support from Elitenetzwerk Bayern in the Ph.D. program ExQM. The funders played no role in study design, data collection, analysis and interpretation of data, or writing of the manuscript.

APPENDIX A: SYNCHRONIZATION OF TIME TRACES

To predict the secret key from recorded emissions, it is necessary to synchronize the symbol sequence with the recorded emissions. We achieve this in two steps. First, we digitally obtain the clock signal from the recorded emissions via a narrow-band-pass filter, allowing us to synchronize the phase; i.e., we determine at which points (spaced 100 samples apart) in the recorded time trace a new symbol begins. Second, we synchronize the absolute time, i.e., find the point in the recorded time trace corresponding to the first symbol of the key. This is achieved with use of a separately recorded dedicated trigger signal from the device (see Fig. 8), which is always zero except for a short time indicating the start of the key.

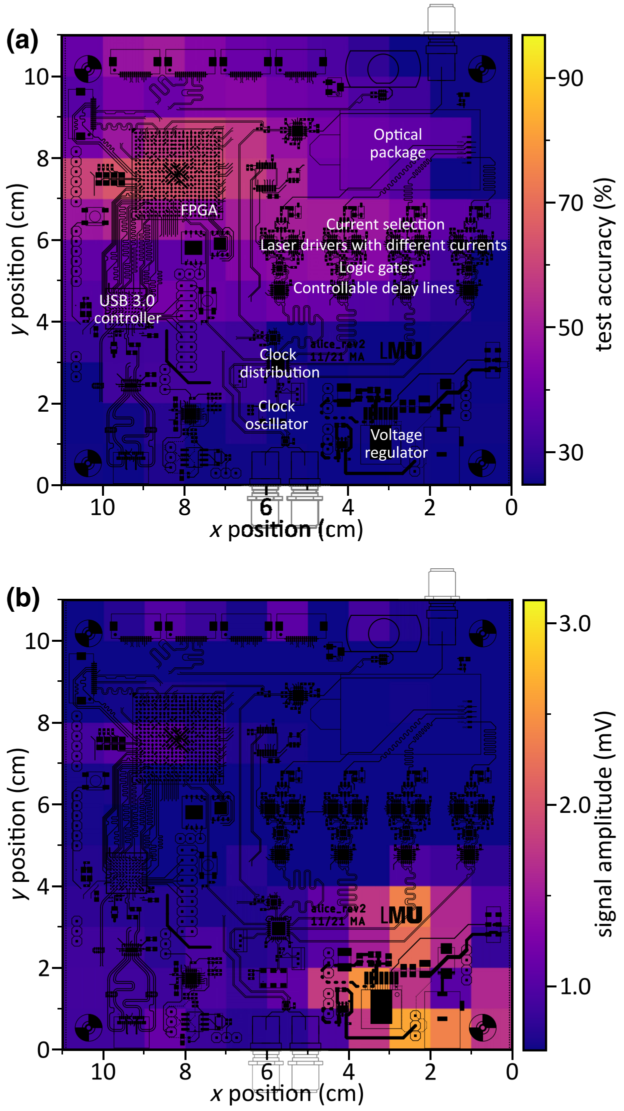

FIG. 7. Test accuracy (a) and amplitude of the probe signal (b) of our revised electronics. Both accuracy and amplitude are significantly reduced compared with the case for the original electronics as shown in Fig. 5. The strongest emissions are observed from the voltage regulator (bottom right on the board), which, however, do not carry information about the key. Nevertheless, despite the much-lower amplitude of emissions from the FPGA, it leaks a significant amount of information about the key.

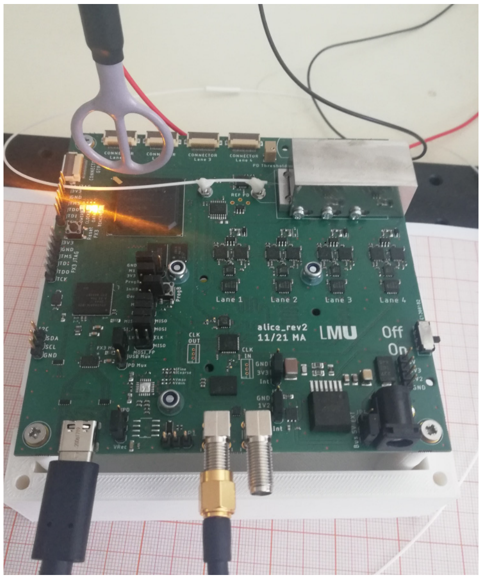

FIG. 8. Photograph of the measurement setup for the revised electronics. The USB-C connection (bottom left) acts as a power supply and the SMA connection (bottom center) supplies the trigger signal for time synchronization. The fiber-connected optics module (top right) was switched off during our measurements so as to test the electronics. When active, its dedicated metal casing effectively shields rf emissions.

We ensure that recording the trigger signal does not affect our results by the following reference measurement. With an otherwise-identical setup, we record two sets of measurements, each set consisting of three datasets, namely, training, validation (see Appendix D), and test datasets. The first set of measurements is performed while we are recording the trigger signal for all datasets, which is used to synchronize and perform the attack as described in the main text. For the second set of measurements, we also record the trigger signal for training and validation datasets and use it for synchronization. However, we do not record the trigger signal for the test dataset. We observe that the training and validation accuracies between the two sets of measurements agree (within the usual variation).

For evaluation of the test dataset of the second set of measurements, which does not contain a trigger signal, we obtain only the clock phase from the filtered probe signal, which leaves the absolute time of the first symbol to be determined. Even though we lack this information, we use the trained neural network to predict the key from the test dataset starting at an arbitrary symbol position. This results in a predicted sequence of symbols, which, in the case of a perfect attack, reproduces the correct key symbols up to a circular shift. To evaluate the test accuracy, we correlate the true key with the predicted key, thus obtaining the optimal circular shift. In all cases, the optimal shift is unique and very easily distinguished from all other shifts. Using this optimal displacement, we find the test accuracy is consistent with that obtained for the first set of measurements, which include the trigger signal also in the test dataset. This shows that recording the trigger signal does not affect our results, justifying its use.

For devices where no trigger signal is available, a different approach is needed to synchronize the training and validation datasets, since the method described above works only for the test dataset. If the rf emissions are sufficiently strong, a header may be used within the key (e.g., a key section with a single repeated symbol, which leads to lower emissions during that time). The position of this header can be located within the measured data, thus providing synchronization of absolute time. For the training and validation datasets, this approach is consistent with the attacker model in Sec. III A. For evaluation of the test dataset, use of a header in the key would violate the attacker model and the synchronization must be achieved by other means (e.g., by finding the optimal shift as described above).

APPENDIX B: SELECTIVE-CHANNEL-BLOCKING ATTACK

A standard assumption in QKD is that the attacker has access to a lossless quantum channel. The levels of optical loss typical in QKD allow the attacker to very strongly preselect the optical pulses where the attack gives the most information and block all others. For example, if the attacker knows 3% of the random key symbols with certainty and guesses the rest at random, this achieves a raw-key prediction accuracy of only about 27.3%. However, at about 15 dB of optical loss, the attacker can block the 97% of unknown pulses, thus gaining complete knowledge of the sifted key.

To select which pulses to block, the attacker may use a model. In our case, the neural network outputs probabilities for each symbol (see Appendix C), which can be used directly. Assume that the attacker can afford to let through 1% of pulses, which amounts to replacing optical loss of 20 dB by a lossless channel. To not influence the proportions of symbols sent, the attacker should let through about the same number of pulses for each symbol.

In simulating this scenario, we re-evaluate the data for the distance measurement. The training procedure remains unchanged. For each test dataset consisting of 20 000 snippets, we artificially restrict ourselves to 1% of snippets. We select 50 snippets for each symbol, keeping only those where that symbol is predicted with the highest probability. While for our device and neural network this only slightly improves the results (Fig. 9), the possibility of selective channel blocking should be kept in mind when one is evaluating performance. This illustrates why it is not sufficient to examine only averaged performance metrics such as prediction accuracy.

APPENDIX C: INFLUENCE OF BASIS CHOICE AND PREDICTION OF SECRET-KEY BITS

In the main text we evaluated the effectiveness of the attacks using prediction accuracy, i.e., the fraction of correctly recovered raw-key symbols. There are several ways to evaluate information leakage and to put this quantity into perspective, for example, for comparison with other side-channel attacks. The neural network does not merely predict each symbol but rather assigns to each snippet a probability distribution over all possible symbols (H, V, P, M), of which the most likely is taken as the prediction. These probabilities are used to calculate categorical cross-entropy, which is a measure of information leakage to the attacker and is minimized when the neural network is trained.

Note that no single averaged metric is sufficient to rule out successful attacks. This is because an attacker can strongly preselect the set of pulses to be those leaking the most information and block the remaining optical pulses; see Appendix B.

Here we wish to relate the raw-key prediction accuracy to the sifted-key prediction accuracy. A standard assumption is that the attacker obtains access to the basis choices during the postprocessing phase. In this case, there are two ways to evaluate the fraction of correctly recovered bits in the sifted key.

First, the predicted symbols of the neural network can be represented as a *confusion matrix*, which shows which symbols are more easily distinguishable than others; see Table I for an example. By associating bit values with the symbols, one can compute the prediction accuracy for sifted-key bits. A bit prediction is still correct when the network confuses two symbols that represent the same bit value in the sifted key, which leads to higher accuracies for bit prediction than symbol prediction. Assuming that the symbols H and P represent bit value 0, while V and M represent bit value 1, the bit prediction accuracy is 89.0% for the example in Table I. However, since the optics design is largely independent of the electronics design, one can also remap which laser-driver line corresponds to which symbol. If the laser drivers originally used for the symbols H and V are rewired to represent bit value 0 and the laser drivers originally used for the symbols P and M are rewired to represent bit value 1, the bit prediction accuracy is 87.1%. If H and M represent 0 and V and P represent 1, the bit prediction accuracy is 92.5%.

A second way to evaluate the sifted-key bit prediction accuracy is to train a neural network for binary classification. In our case, the results agree very well

TABLE I. Confusion matrix of symbol predictions (test dataset) measured on the electronics without countermeasures. The data refer to the measurement shown in Fig. 4 at a distance of 1 cm. The test accuracy of symbol prediction is 84.3%.

		Predicted symbol			
		H	V	P	M
True symbol	H	4917	4	5	31
	V	7	3781	813	500
	P	6	781	3860	327
	M	9	427	220	4299

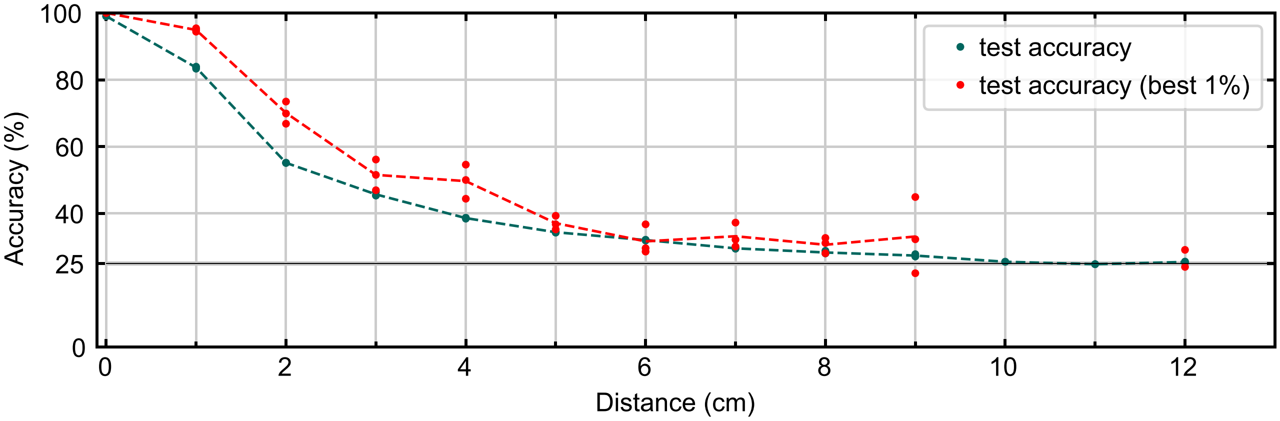

FIG. 9. Same as Fig. 4 but also showing accuracy on a subset (1%) of the test dataset. This subset is defined by our selecting the 50-most-confident (i.e., highest predicted probability) predictions for each symbol. The performance on the restricted test dataset is slightly better. For some locations (10 and 11 cm) the neural network never predicts a certain symbol. In that case, the selection procedure does not apply and test accuracy data on the restricted dataset are omitted. The high statistical variation across the test datasets is due to the much-smaller size of the restricted dataset.

with those obtained from the confusion-matrix approach. The confusion-matrix approach not only gives the attacker information about Alice's basis choices, which is useful for additional attacks using optical measurements, but it also does not require retraining for a new driver or symbol mapping as opposed to the binary classification network.

APPENDIX D: NEURAL NETWORK AND TRAINING

The neural network accepts as input a time trace of 500 samples (corresponding to five symbols with 100 samples each). We normalize the input data to have mean 0 and standard deviation 1 across the entire dataset. To track progress during training and examine generalization of the neural network, we use an additional *validation dataset*, which is obtained by an independent measurement of a single trace (20 000 symbols) with use of the same raw key as for the training datasets. We call the prediction accuracy evaluated on the validation dataset the "validation accuracy."

The neural-network architecture is shown in Table II. One-dimensional convolutional filters identify specific patterns corresponding to switching between symbol combinations, while max-pooling and batch-normalization layers reduce complexity and increase the speed and

TABLE II. Architecture of the neural network. GeLU, Gaussian error linear unit.

Layer type	Output shape	Parameters
Input	(500)	0
Gaussian noise: $\sigma = 0.1$	(500)	0
Convolution (1D): GeLU (dilation rate 1, kernel size 3)	(500, 13)	52
Max pooling (1D): size 2	(250, 13)	0
Spatial dropout (1D): 25%	(250, 13)	0
Batch normalization	(250, 13)	52
Convolution (1D): GeLU (dilation rate 2, kernel size 15)	(250, 118)	23 128
Max pooling (1D): size 1	(250, 118)	0
Spatial dropout (1D): 25%	(250, 118)	0
Batch normalization	(250, 118)	472
Convolution (1D): GeLU (dilation rate 4, kernel size 5)	(250, 100)	59 100
Max pooling (1D): size 4	(62, 100)	0
Batch normalization	(62, 100)	400
Spatial dropout (1D): 25%	(62, 100)	0
Flatten	(6200)	0
Dense: GeLU	(224)	1 389 024
Dense: softmax	(4)	900
		Total: 1 473 128

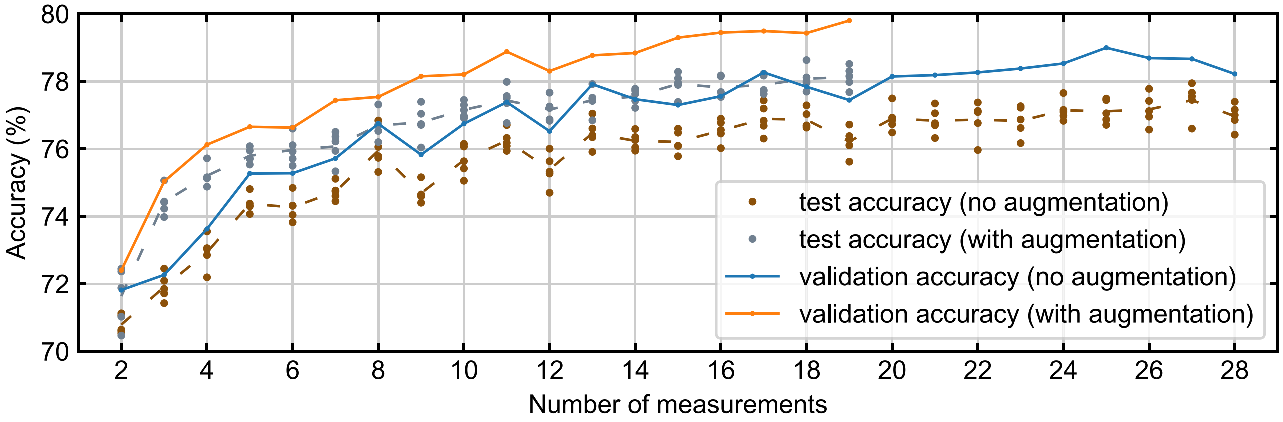

FIG. 10. Dependence of validation and test accuracies on the amount of training and validation data. A single measurement (time trace) is always used for validation, and the remaining measurements are used for training. The network is trained independently for each number of measurements. Each trained neural network is evaluated on five independently measured test datasets. The test accuracies on each are shown separately (dots), as well as on average (dashed lines). This makes the analysis more robust with regard to, for example, fluctuations in the noisy experimental environment. All data shown here are recorded above the FPGA at a distance of 2.5 cm from the circuit board.

robustness of the network, respectively. Dropout layers help avoid overfitting, and the flattening layer reshapes the tensor dimensions. Finally, two dense layers are used to classify the signal into one of four classes corresponding to the symbols H, V, P, and M. For more details on the layers, see, for example, Refs. [34,35]. We implement the neural network using the TensorFlow computing framework [51] and train it by minimizing categorical cross-entropy [52] using the Adam (adaptive moment estimation) optimizer [53]. With this setup, training on seven measurements of 20 000 symbols each takes around 5 min on an Nvidia A40 GPU.

While the network architecture is chosen manually, the hyperparameters (e.g., convolution kernel sizes and noise levels) are found by hyperparameter optimization with use of Hyperband-based methods [54]. This optimization also effectively removes the second max-pooling layer by setting its size to 1. The hyperparameter optimization is performed on a dedicated, larger set of 30 measurements and is repeated for both versions of the electronics (with and without countermeasures). There is no meaningful performance difference between the two hyperparameter sets on measurements from either device. Therefore, to simplify the evaluation, we use the same neural network (obtained by hyperparameter optimization on data measured from the electronics without countermeasures) for all reported findings

We choose how many training data to collect at all locations using the larger dataset, which is also used for hyperparameter optimization. The improvement of validation and test accuracy with increasing amount of measured training data and data augmentation is shown in Fig. 10. For data augmentation, we duplicate the traces and shift them by between one and three samples in either direction. The analysis suggests that seven measurements are representative of the attack's performance, while keeping the data collection manageable. The test accuracies are evaluated on separate measurements of the same raw key, which implies a single-trace attack, while also showing how reliable it is.

Note that the discrepancy between validation and test accuracies is small. This indicates good generalization of the neural network and validates our approach of predicting the center symbol of a small snippet. In the revised electronics, we observed a slightly higher discrepancy between validation and test accuracies.

[1] C. H. Bennett and G. Brassard, in *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing* (Bangalore, 1984), p. 175.

[2] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, Quantum cryptography, Rev. Mod. Phys. **74**, 145 (2002).

[3] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, The security of practical quantum key distribution, Rev. Mod. Phys. **81**, 1301 (2009).

[4] E. Diamanti, H.-K. Lo, B. Qi, and Z. Yuan, Practical challenges in quantum key distribution, npj Quantum Inf. **2**, 16025 (2016).

[5] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani, J. L. Pereira, M. Razavi, J. Shamsul Shaari, M. Tomamichel, V. C. Usenko, G. Vallone, P. Villoresi, and P. Wallden, Advances in quantum cryptography, Adv. Opt. Photonics **12**, 1012 (2020).

[6] F. Xu, X. Ma, Q. Zhang, H.-K. Lo, and J.-W. Pan, Secure quantum key distribution with realistic devices, Rev. Mod. Phys. **92**, 025002 (2020).

[7] T. Schmitt-Manderbach, H. Weier, M. Fürst, R. Ursin, F. Tiefenbacher, T. Scheidl, J. Perdigues, Z. Sodnik, C. Kurtsiefer, J. G. Rarity, A. Zeilinger, and H. Weinfurter, Experimental demonstration of free-space decoy-state quantum key distribution over 144 km, Phys. Rev. Lett. **98**, 010504 (2007).

[8] S. Nauerth, F. Moll, M. Rau, C. Fuchs, J. Horwath, S. Frick, and H. Weinfurter, Air-to-ground quantum communication, Nat. Photonics **7**, 382 (2013).

[9] G. Vest, P. Freiwang, J. Luhn, T. Vogl, M. Rau, L. Knips, W. Rosenfeld, and H. Weinfurter, Quantum key distribution with a hand-held sender unit, Phys. Rev. Appl. **18**, 024067 (2022).

[10] C.-Y. Lu, Y. Cao, C.-Z. Peng, and J.-W. Pan, Micius quantum experiments in space, Rev. Mod. Phys. **94**, 035001 (2022).

[11] L. Knips, M. Auer, A. Baliuka, Ömer Bayraktar, P.Freiwang, M. Grünefeld, R. Haber, N. Lemke, C. Marquardt, F. Moll, J. Pudelko, B. Rödiger, K. Schilling, C. Schmidt, and H. Weinfurter, in *Quantum 2.0 Conference and Exhibition* (Optica Publishing Group, Boston, MA, 2022), p. QTh3A.6, https://opg.optica.org/abstract.cfm?URI=QUANTUM-2022-QTh3A.6.

[12] P. Jouguet, S. Kunz-Jacques, A. Leverrier, P. Grangier, and E. Diamanti, Experimental demonstration of long-distance continuous-variable quantum key distribution, Nat. Photonics **7**, 378 (2013).

[13] H.-L. Yin, T.-Y. Chen, Z.-W. Yu, H. Liu, L.-X. You, Y.-H. Zhou, S.-J. Chen, Y. Mao, M.-Q. Huang, W.-J. Zhang, H. Chen, M. J. Li, D. Nolan, F. Zhou, X. Jiang, Z. Wang, Q. Zhang, X.-B. Wang, and J.-W. Pan, Measurement-device-independent quantum key distribution over a 404 km optical fiber, Phys. Rev. Lett. **117**, 190501 (2016).

[14] A. Boaron, G. Boso, D. Rusca, C. Vulliez, C. Autebert, M. Caloz, M. Perrenoud, G. Gras, F. Bussières, M.-J. Li, D. Nolan, A. Martin, and H. Zbinden, Secure quantum key distribution over 421 km of optical fiber, Phys. Rev. Lett. **121**, 190502 (2018).

[15] Q. Zhang, F. Xu, Y.-A. Chen, C.-Z. Peng, and J.-W. Pan, Large scale quantum key distribution: Challenges and solutions [Invited], Opt. Express **26**, 24260 (2018).

[16] Y.-A. Chen, *et al.*, An integrated space-to-ground quantum communication network over 4600 km, Nature **589**, 214 (2021).

[17] A. Lewis and M. Travagnin, European Commission Joint, Research Centre, 2022).

[18] Y. Zhao, C.-H. F. Fung, B. Qi, C. Chen, and H.-K. Lo, Quantum hacking: Experimental demonstration of time-shift attack against practical quantum-key-distribution systems, Phys. Rev. A **78**, 042333 (2008).

[19] L. Lydersen, C. Wiechers, C. Wittmann, D. Elser, J. Skaar, and V. Makarov, Hacking commercial quantum cryptography systems by tailored bright illumination, Nat. Photonics **4**, 686 (2010).

[20] I. Giechaskiel and K. Rasmussen, Taxonomy and challenges of out-of-band signal injection attacks and defenses, IEEE Commun. Surv. Tutor. **22**, 645 (2020).

[21] P. R. Smith, D. G. Marangon, M. Lucamarini, Z. L. Yuan, and A. J. Shields, Out-of-band electromagnetic injection attack on a quantum random number generator, Phys. Rev. Appl. **15**, 044044 (2021).

[22] National Security Agency (NSA): TEMPEST: A signal problem, in *Cryptologic Spectrum* (1972), https://www.nsa.gov/portals/75/documents/news-features/declassified-documents/cryptologic-spectrum/tempest.pdf.

[23] J. Heyszl, S. Mangard, B. Heinz, F. Stumpf, and G. Sigl, in *Topics in Cryptology—CT-RSA 2012*, edited by O. Dunkelman (Springer Berlin Heidelberg, Berlin, Heidelberg, 2012), p. 231.

[24] L. Masure, C. Dumas, and E. Prouff, in *IACR Transactions on Cryptographic Hardware and Embedded Systems* (IACR, 2020)(1), p. 348, https://tches.iacr.org/index.php/TCHES/article/view/8402.

[25] S. Duan, Z. Li, Y. Luo, M. Sun, W. Wang, X. S. Lin, and X. Xu, in *Emerging Topics in Hardware Security*, edited by M. Tehranipoor (Springer International Publishing, Cham, 2021), p. 111.

[26] K. Durak, N. C. Jam, and S. Karamzadeh, Attack to quantum cryptosystems through RF fingerprints from photon detectors, IEEE J. Sel. Top. Quantum Electron. **28**, 1 (2022).

[27] D. Park, G. Kim, D. Heo, S. Kim, H. Kim, and S. Hong, Single trace side-channel attack on key reconciliation in quantum key distribution system and its efficient countermeasures, ICT Express **7**, 36 (2021).

[28] D. Genkin, A. Shamir, and E. Tromer, Acoustic cryptanalysis, J. Cryptol. **30**, 392 (2017).

[29] H. Ott, *Electromagnetic Compatibility Engineering* (Wiley, Hoboken, New Jersey, 2009).

[30] Data bus: Cypress FX3 USB 3.0. FPGA: Xilinx Spartan 6. VCSEL driver: Texas Instruments ONET4291va. rf amplifier: Texas Instruments TRF37C73. Oscilloscope: Teledyne LeCroy WavePro 604HD with up to 20 GS/s (Gigasamples per second, 1 billion oscilloscope readings per second). Commercial probes from Langer: EMV RF-R 400-1, RF-R 50-1, RF-U 5-2, RF-B 3-2, and XF-R 400-1. Ultrahigh-frequency log-periodic antenna: Creative Design Corporation CLP5130-2.

[31] M. Auer, Master's thesis, Ludwig Maximilian University, Munich (2020), https://xqp.physik.uni-muenchen.de/publications/theses˙master/master˙auer.html.

[32] M. Auer, P. Freiwang, A. Baliuka, M. Schattauer, L. Knips, and H. Weinfurter, in *2021 Conference on Lasers and Electro-Optics Europe and European Quantum Electronics Conference* (Optical Society of America, Munich, 2021), https://www.osapublishing.org/abstract.cfm?URI=EQEC-2021-eb˙p˙3.

[33] J. Patterson and A. Gibson, *Deep Learning—A Practitioner's Approach* (O'Reilly Media, Inc., Sebastopol, 2017).

[34] J. Schmidhuber, Deep learning in neural networks: An overview, Neural Netw. **61**, 85 (2015).

[35] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions, J. Big Data **8**, 53 (2021).

[36] M. E. Celebi and K. Aydin, eds., *Unsupervised Learning Algorithms* (Springer International Publishing, Springer Cham Heidelberg New York Dordrecht London, 2016).

[37] R. S. Sutton and A. G. Barto, *Reinforcement Learning—An Introduction* (MIT Press, Cambridge, 2018), 2nd ed.

[38] P. Xu, X. Zhu, and D. A. Clifton, Multimodal learning with transformers: A survey, ArXiv:2206.06488 (2022).

[39] M. Pereira, G. Kato, A. Mizutani, M. Curty, and K. Tamaki, Quantum key distribution with correlated sources, Sci. Adv. **6**, eaaz4487 (2020).

[40] K.-I. Yoshino, M. Fujiwara, K. Nakata, T. Sumiya, T. Sasaki, M. Takeoka, M. Sasaki, A. Tajima, M. Koashi, and A. Tomita, Quantum key distribution with an efficient countermeasure against correlated intensity fluctuations in optical pulses, npj Quantum Inf. **4**, 8 (2018).

[41] J. Kim, S. Picek, A. Heuser, S. Bhasin, and A. Hanjalic, in *IACR Transactions on Cryptographic Hardware and Embedded Systems* (IACR, 2019)(3), p. 148, https://tches.iacr.org/index.php/TCHES/article/view/8292.

[42] A. Kerckhoffs, La cryptographie militaire, J. Sci. Militaires **9**, 5 (1883).

[43] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer Series in Statistics (Springer, New York, 2009).

[44] R. Wang, H. Wang, and E. Dubrova, in *Proceedings of the 4th ACM Workshop on Attacks and Solutions in Hardware Security* (ACM, Virtual Event USA, 2020), p. 35.

[45] C. R. Paul, *Introduction to Electromagnetic Compatibility* (John Wiley & Sons, Inc., Hoboken, New Jersey, 2005).

[46] J. Kuruvilla, W. Runcy, and G. Gejo, *Materials for Potential EMI Shielding Applications* (Elsevier, Amsterdam, 2020).

[47] M. Robinson, T. Benson, C. Christopoulos, J. Dawson, M. Ganley, A. Marvin, S. Porter, and D. Thomas, Analytical formulation for the shielding effectiveness of enclosures with apertures, IEEE Trans. Electromagn. Compatib. **40**, 240 (1998).

[48] M. Guri, B. Zadov, A. Daidakulov, and Y. Elovici, ODINI: Escaping sensitive data from Faraday-caged, air-gapped computers via magnetic fields, ArXiv:1802.02700 (2018).

[49] A. Baliuka, M. Stöcker, M. Auer, P. Freiwang, H. Weinfurter, and L. Knips, Software for deep learning based radio frequency side-channel attack on quantum key distribution (2023).

[50] A. Baliuka, M. Stöcker, M. Auer, P. Freiwang, H. Weinfurter, and L. Knips, Datasets for deep learning based radio frequency side-channel attack on quantum key distribution (2023).

[51] M. Abadi, *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems (2015). Software available from https://www.tensorflow.org/.

[52] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Boston, MA, 2016).

[53] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, ArXiv:1412.6980 (2015).

[54] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, Hyperband: A novel bandit-based approach to hyperparameter optimization, J. Mach. Learn. Res. **18**, 1 (2018).