Securing Reset Operations in NISQ Quantum Computers

Allen Mi
Yale University
New Haven, CT, USA
allen.mi@yale.edu

Shuwen Deng
Yale University
New Haven, CT, USA
shuwen.deng@yale.edu

Jakub Szefer
Yale University
New Haven, CT, USA
jakub.szefer@yale.edu

ABSTRACT

A secure reset operation could be an enabling technology that allows sharing of a quantum computer among different users, or among different quantum programs of the same user. A secure reset operation could allow for resetting a subset of qubits of the quantum computer between each user or program so that their state is erased and a new program or user can use the qubits while computation continues on the other qubits. Today the dominant method to erase the qubit state is a full system wipe, which effectively resets all the qubits at the same time. In today's superconducting qubit machines from IBM, for example, a full system wipe takes up to 1000 μs, and it fully erases all information in the system. However, with a full system wipe there is no means for only a few qubits to be cleared and assigned to a new user or program, everything has to be erased at the same time. A secure reset operation could allow resetting only a subset of qubits, and it could be built upon existing (insecure) reset operation available from superconducting qubit machines from IBM, for example. The (insecure) reset operation is available today, which can be used to reset the state of a qubit in a time on the order of 10 μs down to 1 μs. The reset operation is thus much faster than a full system wipe. However, as this work demonstrates, today it is possible to leak some information across the (insecure) reset operation as it does not perfectly reset the qubit state between two users or programs who may be sequentially scheduled on the same qubit. Further, crosstalk-like effects are observed where reset behavior of one qubit can be inferred from an adjacent qubit. This work analyzes the existing (insecure) reset operation in order to understand how a secure reset operation could be built upon it. This work then describes the design, implementation, and evaluation of the proposed secure reset operation which can reset qubits without leaking information, and at the same time is still about 300x faster than a full system wipe.

CCS CONCEPTS

• **Security and privacy → Security in hardware**; • **Hardware → Quantum technologies**.

KEYWORDS

quantum computers, information leakage, reset gates

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
CCS '22, November 7–11, 2022, Los Angeles, CA, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9450-5/22/11.
https://doi.org/10.1145/3548606.3559380

1 INTRODUCTION

Today's quantum computers are commonly called Noisy Intermediate-Scale Quantum (NISQ) quantum computers [18], as they are too small for quantum error correction (QEC) or even for large benchmarks, but already have applications in optimization, chemistry, and other important areas [11, 12, 14]. Further, quantum computing hardware keeps evolving at a fast pace, with 100-qubit quantum computers being now a reality, and 1000-qubit quantum computers being projected to come online in the next few years [8].

Quantum computers of these sizes have the potential to fundamentally alter what types of algorithms that can run on them, but require specialized facilities and equipment. In order to make these quantum computer accessible to users. There is a growing interest in, and practical deployments of, cloud-based quantum computers, also called Quantum as a Service (QaaS) or Quantum Computing as a Service (QCaaS). Cloud-based services such as IBM Quantum, Amazon Bracket, and Microsoft Azure already provide access to quantum computers remotely for users. Following the past success of classical computer cloud-based services, we expect that cloud-based access for remote users to rent quantum computers to be a dominant use-case in the future.

To maximize efficiency and utilization of the quantum computers, they need to have a way to efficiently and quickly switch between users and programs running on these computers. At the same time, cloud-based quantum computers are vulnerable to many threats not present in in-house uses of quantum computers. Especially, remote users could be malicious and try to learn about the infrastructure, harm the infrastructure, attack other users, or leak information from other users. Consequently, when switching between users and programs, there is a need to ensure strong isolation and that no information is leaked.

A secure reset operation could be an enabling technology that could allow sharing of a quantum computer among different users, or even among different quantum programs of the same user. Today, the main method to clear the qubit state is through a full system wipe. A full system wipe in today's superconducting qubit machines such as from IBM takes on the order of 1000μs, and fully erases all information in the system. However, full system erases all the qubits at the same time, preventing useful multi-tenant setting where different users or programs can share the quantum computer at the same time and thus may need to have the qubits cleared at different times as users or programs start or finish their jobs on the assigned qubits.

One building block for a secure reset operation that can reset individual qubits but allow others to continue executing is an existing (insecure) reset operation. This operation can be used to reset state of a qubit in only about 10μs down to less than 1μs. The existing reset operation is then almost 1000 times faster than a full wipe employed between users today in IBM machines. However, it has

not been previously evaluated or analyzed for possible information leakage or crosstalk-like effects.

As we have learned from classical computing, many remote attacks become feasible when computers are put in public cloud computing data centers [9, 26–29]. The threats can be categorized into *attacks on the infrastructure*, e.g., reverse engineering the infrastructure or harming the infrastructure, and *attacks on other users*, e.g., attacking (or interfering with) other users or leaking information. The same types of threats will become applicable to quantum computers, especially as many are already available as cloud-based services. Thus to further secure today's single-tenant quantum computers, and to enable multi-tenant quantum computers, security of the reset operations needs to be analyzed, and secure reset operation developed.

In particular, this is the first work to explore the existing (insecure) reset operations used in superconducting quantum computers from IBM Q and to show that they do not protect fully from information leakage. A reset operation is composed of a measurement operation and a conditional X gate (the X gate is the quantum computer equivalent of the `NOT` gate for classical computers with respect to the standard basis $|0\rangle$ and $|1\rangle$, c.f. Section 2 for more details). Since the reset operation is conditional on measurement results, its outcomes are closely associated with the error characteristics of the measurement operation. As we demonstrate with repeated testing, an attacker measuring the qubit state post-reset can statistically recover some information about the qubit's state prior to the reset, thus leaking information from the victim user who was using the same qubit prior to the attacker. In addition, this work further exposes a crosstalk-like behavior where information is leaked from a victim qubit to an adjacent attacker qubit where a reset or measurement is performed. The new observed crosstalk-like behavior occurs since we observe that victim's measurement or reset operation on one qubit impacts the results of measurement or reset operation performed by the attacker on an adjacent spectator qubit.

Nevertheless, the existing reset operation can be a building block for a secure reset operation that could be an enabling technology for sharing of quantum computers and for multi-tenant quantum computers. In particular, as we demonstrate in this work, a new operation based on a randomized number of resets can significantly limit the amount of information leaked, while still being faster than a full system wipe, thus enabling fast switching between users or programs in a multi-tenant quantum computer setting.

1.1 Contributions

The contributions of this work are:

- Formulating problem analysis and developing threat model for thinking about single-tenant and multi-tenant quantum computers.
- Demonstrating information leakage which exists across reset operations in superconducting IBM machines, which could leak information in both single-tenant and multi-tenant sharing settings.
- Uncovering crosstalk-like information leakage from a victim measurement or reset operation to an attacker measurement or reset operation happening on an adjacent qubit, which could leak information in multi-tenant settings.
- Demonstrating a potential bug or flaw in realization of reset operations on one of most recent IBM machines, the Perth backend.[1]
- Developing design, implementation, and evaluation of first secure reset operation for quantum computers, evaluated and deployed for testing on real quantum computer hardware and not in simulation.

1.2 Code Availability

The code used in this work is available under open-source license at https://caslab.csl.yale.edu/code/qc-secure-resets/.

2 BACKGROUND

This work focuses on superconducting qubit quantum computers [19], with specific evaluation and analysis done on publicly accessible IBM quantum computers [14]. There are also other types of quantum computers such as ones using trapped ion qubits [4]. While they are not the focus of this work, we believe that secure reset operations for these machines also need to be developed, and are the focus of our future work. For now, the focus is on IBM machines, and below we summarize some useful terminology and ideas regarding superconducting qubit machines.

2.1 Quantum Computer Concepts

Qubits - are building blocks of quantum computers, and they represent data as quantum states. The data can be in superposition, a combination of classical 0 and 1. The qubit state has to be collapsed (via a measurement operation) to a classical 0 or a 1 during the measurement, also called readout. The classical bits are measured by projecting the state onto the z-axis of the Bloch sphere, where the two eigenstates are $|0\rangle$ and $|1\rangle$. They correspond to the measurement results of `0` and `1` respectively.

Bloch sphere - is a geometrical representation of the Hilbert space of a two-level quantum mechanical system. The Bloch sphere is a unit 2-sphere, with antipodal points corresponding to a pair of mutually orthogonal state vectors. The north and south poles of the Bloch sphere typically correspond to the standard basis vectors $|0\rangle$ and $|1\rangle$, respectively. Given an orthonormal basis, any pure state $|\psi\rangle$ of a two-level quantum system can be written as a superposition of the computational basis vectors $|0\rangle$ and $|1\rangle$. We also know from quantum mechanics that the total probability of the system has to be one: $\langle\psi|\psi\rangle = 1$, or equivalently $\|\psi\|^2 = 1$. Denote parameters θ and ϕ in spherical coordinates to be respectively the colatitude with respect to the z-axis and the longitude with respect to the x-axis. The constraint is satisfied by the Bloch sphere representation of an arbitrary pure state:

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\phi}\sin(\theta/2)\,|1\rangle$$

where commonly $0 \leq \theta \leq \pi$ and $0 \leq \phi < 2\pi$.

Quantum gates - are used for computation or measurement to set the state of the qubits. Quantum gates can be represented by unitary

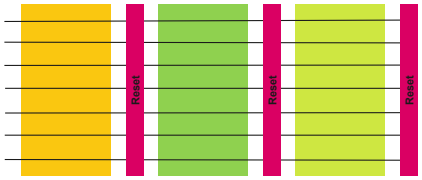
[1] The bug or flaw is in addition to the security problems which we demonstrate for all the backends. Security issues with resets in all the backends are discussed in Section 6.5, while the particular Perth machine bug is in Section 6.7.

matrices and carry out reversible operations. There are most often single-qubit gates such as the Hadamard gate H and two-qubit gates such as the CNOT gate. Some gates are natively supported by quantum computer hardware, while other gates can be created from these native gates. Most NISQ quantum computers, including the ones available through IBM, support only a few native gates: four single-qubit gates (I, R_z, $\sqrt{X}$, and X) and one two-qubit gate (CNOT). Any other gate needed by a program needs to be decomposed to these native gates, increasing the number of gates and running time of the program. Some of the gates can be executed conditionally, where a classical bit determines whether operation occurs or does not on the quantum state. To the best of our knowledge, conditional gates such as conditional X gate are not yet available to users using IBM machines; but they are implicitly used inside a reset operation, discussed later. The gates operating on multiple qubits require a coupling (a connection) between the qubits, otherwise, the two-qubit gate cannot be executed on these two qubits. It may be possible to perform quantum program transformations to use other intermediate qubits to emulate the two-qubit gate on qubits that are not connected directly, e.g., by use of SWAP gates, but these also increase the complexity of the program.

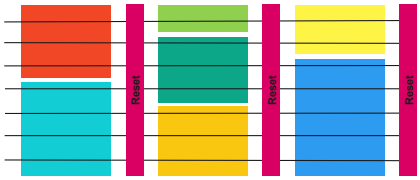
Measurement – is used to extract classical (digital) information from qubits. In addition to the unitary gates, the measurement operation M is an essential building block of almost all quantum circuits. For IBM's superconducting devices, measurement also enables the implementation of the reset operation. We discuss this implementation in Section 5. As a single qubit operation, a measurement reads out the state of the qubit and maps it to classical bit `0` or `1`. During this process, the qubit state is projected along the z-axis of the Bloch sphere and collapsed to either $|0\rangle$ and $|1\rangle$, respectively for the `0` or `1` measurement outcomes. The measurement operation is irreversible, as information contained in the original qubit state can no longer be recovered after the collapse. For general qubit states, the measurement outcome is non-deterministic, with probabilities given by the Born rule. Specifically, given θ in the Bloch sphere characterization of a qubit state, measurement yields `0` with a probability of $\cos(\theta/2)^2$ and `1` with $\sin(\theta/2)^2$. Therefore, a large number of measurement trials (shots) on identically prepared qubit states is required to approximate the probabilities.

Transpiler – is the software that maps algorithms or programs to the specific quantum computer hardware for execution. It may be required to translate the gates or operations specified by the user into the gates or operations supported by the target hardware. It may also optimize the programs, similar to optimizations done by classical compilers, by, for example, combining sequences of operations. The transpiler, which is part of the Qiskit software development kit used by IBM Q, does not insert additional reset gates or otherwise modify the circuit to help mitigate information leakage from resets. To the best of our knowledge, today's cloud-based quantum computers such as IBM Q and Qiskit do not conduct mandatory optimizations or defenses. However, defense ideas based on adding security features to the transpiler are discussed in Section 7.

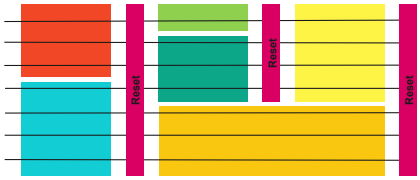
Scheduler – is the software tasked with assigning programs (or users) to specific quantum computers, or specific parts of a quantum computer if multi-tenancy is considered. The scheduler needs to ensure that the target quantum computer, or sub-region of a bigger quantum computer in a shared setting, has the required topology to run the target program. The scheduler may be aware of or try to mitigate different sources of noise. It may also map programs to quantum computer hardware by using different optimization goals, such as minimizing the number of quantum computers needed by multiple users (by maximizing sharing), for example. These optimizations are not mandatory and are off by default.



(a) Example of single-tenant uniform-batch (STUB) sharing.



(b) Example of multi-tenant uniform-batch (MTUB) sharing.



(c) Example of multi-tenant heterogeneous-batch (MTHB) sharing.

Figure 1: Example diagrams of the three possible sharing paradigms of quantum computers. The x-axis represents time, black lines represent qubits and the colored blocks represent different users' temporal and spatial allocations. The "Reset" blocks represent points where secure reset operation would need to be used to quickly and securely reset the qubit state. Note that the figure is not to scale.

3 ENABLING SECURE MULTI-TENANT QUANTUM COMPUTERS

Secure reset operation can be enabling technology for secure multi-tenant quantum computers. Multi-tenant quantum computers are now being actively researched [6], although how to actually realize them in detail, such as with secure reset operations, has not been explored before. In particular, the existing full system wipe is not sufficient to support multi-tenancy.

Below, we present details of three possible multi-tenancy sharing scenarios, to illustrate why it is necessary for different scenarios to be able to (securely) reset only some of the qubits, while others keep running.

First, *single-tenant uniform-batch (STUB)* sharing occurs when each user gets all the qubits of a backend dedicated to them (even if they may not need all of them), shown in Figure 1a. When the user finishes, there is a wipe (or equivalently all qubits are reset with a reset operation) and the next user is loaded. This model

of sharing corresponds directly to what is available today from IBM and other cloud-based quantum computer providers such as Amazon Bracket. To the best of our knowledge, today STUB is realized by utilizing a full system wipe, but this is expensive in terms of time and much faster sharing could be achieved if a secure reset operation is realized.

Second, *multi-tenant uniform-batch (MTUB)* sharing occurs when different users may be using (mutually disjoint) sets of qubits, but all users are scheduled in batches which end at the same time, shown in Figure 1b. Uniform-batch sharing makes the scheduling easier, but all concurrent users have to fit into same-length time slots. MTUB could be realized at the transpiler level (by having multiple programs or users compiled together) or by the scheduler (by placing different users or programs on disjoint stats of qubits at runtime). Because resets of all qubits happen at the same time, either a full system wipe can be done between each batch, or a secure reset operation on all qubits could be leveraged for faster operation.

Third, *multi-tenant heterogeneous-batch (MTHB)* sharing occurs when different users may be using (mutually disjoint) sets of qubits, shown in Figure 1c, but not all users or programs have to end at exactly the same time. This allows for overlap of resets of some users, while other users execute on adjacent qubits. This is the most flexible way of allocating users compared to STUB and MTUB and allows for maximum usage of the machines. MTHB cannot be realized with a full system wipe as not all quits are always reset at the same time, and secure reset operations are the only way to make MTHB a reality.

4 THREAT MODEL

In order to analyze the problem and develop secure reset operations, we propose a below threat model so that our corresponding secure mechanism can effectively prevent the attacks even with strong assumption. In this model, the attacker has control over the execution of the victim program, can repeat measurements, and can be conveniently co-located with a target victim of choice in a predictable manner.

We assume scenario where the victim program runs on certain qubits of a quantum computer. A strong attacker is able to run both in parallel (on a disjoint set of qubits from qubits used by the victim) to measure crosstalk-like effects from the victim and at the same time he or she is also able to run after the victim, on the same qubits as the victim used. We assume the qubits used by the victim are reset before attacker is able to use them. Demonstrating how to securely reset the qubits so that attacker learns no information is the objective of this paper.

We assume the objective of the attacker is to learn the information about the state of the victim's qubits after the victim has finished his or her computation and read out the qubits. Especially, we assume that the quantum computer provider has strong logical isolation so that outputs of the victim cannot be directly accessed by the attacker, otherwise it would be trivial to learn the results of the victim's computation and attackers would not have to resort to use of information leakage and side channels.

We assume that attacker has some degree of knowledge about the algorithm being executed by the victim. We consider two cases. First, the attacker has full knowledge of the victim algorithm, e.g., he or she knows victim is executing Grover's search algorithm, but not the inputs. Consequently they can try to learn the results of the victim's algorithm from the information leakage from the output even if they don't know the inputs. Second, the attacker has some knowledge of the victim algorithm, e.g., he or she knows that it is a quantum machine learning algorithm, and knows the input, but does not know specific parameters of the algorithm. Consequently they can learn some information about the structure of the algorithm given the inputs and the output. Considering that the attacker's goal is to learn the output, we further assume a scenario advantageous to the attacker where he or she knows that the output of a qubit will be either $|0\rangle$ (which $\theta = 0$) or $|1\rangle$ (which is $\theta = \pi$). This is easiest scenario for the attacker since they only need to distinguish the two ends of the measured output frequencies (only for $\theta = 0$ and $\theta = \pi$). If the output distribution can contain other values of θ or if the attacker does not know the output distributions then they have worse chance to learn the output. Thus we assume scenario best for the attacker where they only have to guess between two most distant values of θ.

We also give the attacker advantage of always being co-located with the victim. Based on existing work on quantum computer fingerprinting [15] we assume the attacker is able to identify the quantum computer hardware and can consistently be co-located with the victim.

5 ANALYSIS OF EXISTING RESETS

Compared to a full system wipe, secure reset operations are possibly much faster alternative to reinitialize qubits between users, and are necessary and enabling technology for implementing multi-tenant quantum computers. Secure reset operations can be built upon existing (insecure) reset operations, such as ones available on IBM superconducting qubit quantum computers. However, it is first necessary to examine the behavior and potential limitations of the existing reset operations in order to build the secure reset operations we propose in this work. In this section we analyze existing reset operations and demonstrate that some information can be leaked across the resets between two users sequentially assigned to the same qubit and that there is crosstalk-like effect leaking information from victim qubit to a different attacker qubit when two are used in parallel. These findings are later used to build secure reset operations.

5.1 Existing Reset Operation

As shown in Figure 2, a reset operation consists of a measurement operation M which yields the classical bit c from the qubit q. Following the measurement there is a conditional X gate which will set the qubit to the $|0\rangle$ state if it is not already in that state. Specifically, the X gate, also called Pauli-X gate, is the quantum equivalent of the classical `NOT` gate with respect to the standard basis $|0\rangle$ and $|1\rangle$. When conditioned on the measurement outcome, the X gate will not be invoked if the qubit returns a measurement result of `0` and its post-measurement state is already in $|0\rangle$. On the other hand, if the qubit returns a measurement result of 1 and is collapsed to $|1\rangle$, the X gate will flip the state back to $|0\rangle$. In the ideal scenario, this effect ensures that the qubit is always in the $|0\rangle$ after the reset. Nonetheless, we show in Section 5.3 the reset is not perfect in

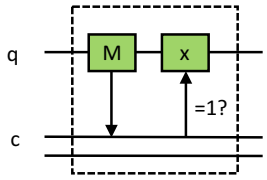


Figure 2: Reset operation is composed from a measurement operation, followed by a conditional X gate, which conditionally flips the post-measurement state from $|1\rangle$ to $|0\rangle$ if the measurement result is 1. Here q is the target qubit and c is the classical register corresponding to the qubit.

real-world scenarios, leading to potential information leaks due to errors in the measurement operation.

To achieve the measurement operation required by the reset operation R, the quantum computer control hardware needs to read out the value of the qubit. The readout is done via a measurement channel and leverages a readout resonator on the quantum computer's chip. When a reset operation is used, the control hardware needs to couple the to-be-reset qubit to the resonator. In our experiments, the readout operation seems to induce a crosstalk-like behavior that leaks information to other qubits on the same device.

We believe there are three key features of reset operations that are related to potential information leakage and need to be considered for security. Recall that the M operation and the conditional X gate are integral parts of the reset R operation. The three features are:

(1) **Timing** of the M and conditional X
(2) **Error channel** of the M and conditional X
(3) **Coupling** of the qubit of interest to the readout resonator

We discuss each of the three features below.

5.2 Timing of the Reset Operation

According to our experiments with the IBM machines, the reset operations are uniform in timing for each qubit, regardless of the outcome of M. This makes timing-based attack on the reset operation impossible on its own. Considering the operation timing, the current generations of IBM computers seem to do a very good job with regards to the added delay to make the reset operation uniform regardless of whether the X gate is invoked or not.

However, making reset operations faster can improve the performance of the circuits running on the quantum computers, by for example making them non-constant time. For the latest backends, a typical measurement operation takes about 700 ns while the X gate itself takes about 36 ns. A designer can improve the operation time by 2% to 3% on average by making it non-constant time. Without mitigation, this small gain would lead to simple timing-based attacks, and is not worth the trade-off.

5.3 Error Channel of the Reset Operation

At this time, due to the unavailability of the conditional X gate on IBM's backends, we elect to characterize the error channel of the reset operation based on known error characteristics of the measurement. Recall that a measurement M projects a quantum state $|\psi\rangle$ onto the computational basis, that is, the basis spanned by $\{|0\rangle, |1\rangle\}$. For any $|\psi\rangle$, the output state of M is therefore a probability mixture of $|0\rangle$ and $|1\rangle$. Suppose $i \in \{0, 1\}$, let $P(|i\rangle)$ denote the probability that the qubit attains $|i\rangle$ post-measurement, and let $P(\mathtt{i})$ denote the probability of reading out a $\mathtt{i}$ from the measurement. Ideally, we have $P(|0\rangle) = P(\mathtt{0})$ and $P(|1\rangle) = P(\mathtt{1})$. In reality, however, there exists a nonzero probability of misattribution. When such misattribution occurs, the conditional X gate is provided with an input opposite the correct value. In these cases, the reset operation outputs $|1\rangle$ instead of $|0\rangle$.

Since qubits are commonly implemented with a two-level quantum system with $|1\rangle$ being the higher-energy state, the probability of mislabeling $|1\rangle$ is higher than that of mislabeling $|0\rangle$. Therefore, victim qubit states that consist of a greater amplitude of $|1\rangle$ would yield a higher frequency of $|1\rangle$ post-reset.

Recall the Bloch sphere representation

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\phi}\sin(\theta/2)\,|1\rangle\,.$$

By the Born rule, the probability of yielding $|1\rangle$ post-measurement is $\sin^2(\theta/2)$. In light of this fact, we define the reset operation error channel Q via the post-reset probability of yielding $|1\rangle$, given θ in the victim state $|\psi\rangle$. The probability is also known as the heavy-output probability:

$$Q(\theta) = a[b\sin^2(\theta/2) + (b-1)\theta/\pi] + c \tag{1}$$

where $a \in [-1, 1], b \in [0, 1], c \in [0, 1]$ are device- and qubit-specific parameters. For $\theta \in [0, \pi]$, $Q(\theta)$ should follow a continuous sigmoid-like pattern. Within the parameter space and the θ domain, $Q(\theta)$ is also monotone (apart from when $a = 0$), and therefore has an inverse. Given this property, we can uniquely recover θ from any $Q(\theta)$ value when the parameters are known. The parametrization can be interpreted as follows:

- a controls the amplitude of the pattern. Observe that $a = Q(\pi) - Q(0)$ regardless of the other parameters. Since $Q(\theta)$ is monotone within the domain, $|a|$ describes the interval length of the image under Q.
- b controls the curvature of $Q(\theta)$. When b varies from 0 and 1, $Q(\theta)$ has increasingly pronounced curvatures. b offers the flexibility of modeling nonlinear probability decay patterns over different values of θ.
- c controls the intercept of $Q(\theta)$. It is helpful for modeling probability variations that are constant over θ.

This parametrization is central to our evaluation of the information leakage described in Section 6.

5.4 Coupling of the Qubit during Reset

Recall that the reset R operation is made up of a measurement M operation and conditional X gate. To perform a measurement, also called a readout, there is a physical readout resonator. To the best of our knowledge, the readout resonator can be shared by multiple qubits. We assume the control hardware only couples a qubit to the readout resonator if a measurement operation is scheduled to occur on the qubit, otherwise it is not coupled. Since reset R operation includes a measurement M operation, the qubit that is being reset is assumed to be consequently coupled to the resonator because of the implicit measurement operation.

Further, the qubits can be accessed via drive channels and measurement channels. Control pulses are sent on the measurement channels to obtain information from the readout resonator. If there is no M operation used on a qubit (due to explicitly measurement or implicitly as part of reset operation), the measurement channel will not be utilized. Our experiments in Section 6.6 indicate that the use of the measurement channel, and readout resonator, is directly related to crosstalk-like effects between qubits.

6 EXPERIMENTAL EVALUATION OF EXISTING RESETS

In this section, we demonstrate the means of acquiring, characterizing, and leveraging information leakage across resets and through crosstalk-like effects.

When analyzing possible information leakage across reset operations, our objective is to show how to reconstruct the Bloch sphere θ angle of the victim qubit state with adequate accuracy, effectively approximating the measurement probabilities of the victim state. When analyzing leakage, we consider that the inputs to the reset operation are different in the cases of whether the victim performs a measurement. First, if the victim does not measure the qubit at end of its execution, then the input to the reset is a pure state of the qubit. This may be the case for ancillary qubits that the victim does not measure at the end of their execution. Second, if the victim measures the qubit, then the input to the reset is a probabilistic mixture of $|0\rangle$ and $|1\rangle$, the two eigenstates corresponding to a Z-basis measurement. We have observed slightly different behavior of the existing reset operation in the two cases, and hence consider them both in our evaluation.

When analyzing possible crosstalk-like effects, our objective is to show how to infer the length of the victim circuit and the delay between its final measurement and the end of the circuit. This method would enable the attacker, for example, to an approximate number of reset operations used by the victim.

6.1 Evaluated Real Devices

Our experiments are performed on current-generation (r5.11) H7 devices of IBM machines. This scope is chosen as only r5.11 devices are capable of mid-circuit measurements. These consist of quantum computers Jakarta, Lagos, and Perth. As shown in Figure 3, these devices consist of seven qubits arranged in an H-shaped topology.

For this generation of superconducting devices, design improvements target speed-ups in qubit state readout. Demonstrating error mitigation is essential for fast readout. To enable this, advanced filtering techniques and fine-tuning of various components' couplings on-chip accomplishes the paradoxical requirements of stronger readout coupling yet protection from qubit relaxation, which enables mid-circuit measurements. Table 1 displays the duration of measurement and reset operations on each device and qubit. Observe that the newer devices (Lagos and Perth) display time costs an order of magnitude smaller than Jakarta.

6.2 Victim and Attacker Circuits

In this section, we describe the quantum circuits used for evaluating the existing reset operations.

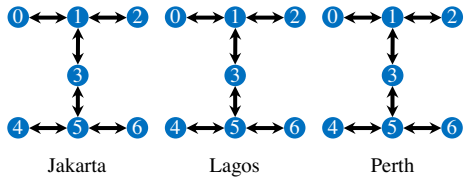


Figure 3: The 3 IBM machines (backends) used in the evaluation. The figure shows the qubits and physical topologies for each backend. The backends can be grouped according to their H-shaped topology. These are represented in text as H_7 backends.

Qubit	Jakarta		Lagos		Perth	
	t_M	t_R	t_M	t_R	t_M	t_R
q_0	9632	13216	1267	1804	1216	1433
q_1	9632	13216	1267	1804	1216	1382
q_2	9632	13216	1267	1792	1216	1600
q_3	9632	13216	1267	1779	1216	1433
q_4	9632	13216	1267	1804	1216	1433
q_5	9632	13216	1267	1804	1216	1433
q_6	9632	13216	1267	1779	1216	1433

Table 1: Per-qubit measurement (t_M) and reset (t_R) durations for backends Jakarta, Lagos, and Perth at the time of writing. Units are in ns. For reference, recall that operations in IBM quantum computers are also sometimes expressed in units of dt, where 1dt = 2/5ns. Compared to Jakarta, measurement and reset operations on Lagos and Perth are approximately one order of magnitude faster.

6.2.1 Evaluating Leakage Across Resets. In this scenario, the victim operates on a single qubit. To provide sufficient coverage of the Bloch sphere, we tailor a series of victim circuits that produce qubit states given by Bloch sphere parametrization $\theta, \phi \in \{0, \pi/8, \ldots 7\pi/8, \pi\} \times \{0, \pi/4, \ldots, 7\pi/4\}$, with a total of 72 configurations. At the end of the victim circuit, we also consider both cases of whether a measurement operation is performed. Immediately after, we insert different numbers of resets, up to 32 for some tested systems. The attacker circuit then follows, which only consists of a single measurement operation. For each configuration, the circuit is run for 8192 shots, and the victim (if applicable) and attacker measurement results are saved as 1-output frequencies.

6.2.2 Evaluating Leakage Through Crosstalk. For this case, we examine the crosstalk leakage between two qubits across the (q_0, q_1) coupling with q_0 as the $|0\rangle$-initialized victim qubit and q_1 as the $|1\rangle$-initialized attacker qubit. We perform three consecutive reset or measurement operations on the victim qubit q_0 at regular intervals of two times the reset operation length. Additionally, for control group experiments, we leave q_0 idle for the same total time, before eventually performing one of the following:

(1) End the victim circuit.
(2) Add a Hadamard H gate to q_0.
(3) Measure q_0.

For the attacker qubit q_1, we idle the qubit for various amounts of delay, before eventually measuring it. By studying how the 1-output frequency of the measurement result as a function of delay, we aim to characterize the impact of various operations on q_0 to q_1 through crosstalk-like effects. Again, all experiments are performed with 8192 shots.

6.3 Testing and Training Phases

For leakage across resets, the data collection in the evaluation is divided into two phases: testing and training. The two phases consist of two identical sets of experiments defined in Section 6.2.1 run in a back-to-back fashion. For each phase, the results are organized with respect to the angle θ. For the training phase, the error channel parametrization defined in Section 5.3 is fitted to the data via a mean-squared error loss. The learned parameters a, b, c thus constitute a quantification of the error channel specific to its scope (i.e., the device, qubit, number of resets, and whether the victim qubit is measured). In the testing phase, the inverse of the fitted function is then used to reconstruct the θ angles that correspond to the 1-output frequencies in the identical scope. In the next section, we discuss a few metrics that characterize the fidelity of this reconstruction.

Since there are no online information dependencies between the two phases, they can run in either order. The ability to perform the training phase after testing means that the attacker can limit training to the qubits where the collection of testing data has succeeded. This simplifies the attack and reduces its training time cost.

6.4 Fidelity Metrics

We propose three metrics to characterize the fidelity of the reconstruction.

6.4.1 Signal-to-Noise Ratio. For each error channel characterization with parameters a, b, c, we take the testing data of the same scope. For each θ value in the testing data, we compute the standard deviation $\sigma(\theta)$ of its 1-output frequency over different values of ϕ. Recall that a represents the amplitude of $Q(\theta)$. Therefore, we take $\sigma(\theta)/a$ as the local signal-to-noise (SNR) ratio at θ. We then take the mean SNR across all θ values to produce the output.

6.4.2 Binary Classification Accuracy. This metric is restricted to testing data of $\theta \in \{0, \pi\}$ (i.e., qubit states that approximate $|0\rangle$ or $|1\rangle$). For each θ, we acquire reconstruction θ^* from the corresponding 1-output frequency via the channel characterization. We then compare the proximity of θ^* to 0 and π, and choose the reconstructed qubit state to be $|0\rangle$ or $|1\rangle$ correspondingly. We repeat this process for all scopes and output the mean classification accuracy. This metric is especially useful for evaluating attack performance on victim circuits with a 1-output frequency close to 100% on some qubits, such as Shor's factorization and Grover's search [10, 24].

6.4.3 Angle Prediction Loss. This metric applies to all testing data and operates similarly to the binary classification accuracy. Instead of performing a $|0\rangle, |1\rangle$ classification, we note the difference between θ and reconstruction θ^*. We then output the θ-specific L_2 norm of this difference across all ϕ values for each scope in the testing data.

6.5 Characterizing State Retention Across Resets

As shown in Figure 4, the reset error channel can be closely modeled by our $Q(\theta)$ characterization and displays a sigmoid-like pattern for the majority of cases. The amplitude of the pattern is significantly compressed after one reset, with further compressions of a lesser degree after additional resets. In some cases (e.g., qubit 4), inverted sigmoid patterns can be observed after a few (e.g., 3 or 4) resets. Note that the Perth case is anomalous, and will be discussed in Section 6.7.

As shown in Figure 5, the three proposed metrics exhibit significant correspondence with each other, especially between the SNR and the angle prediction loss, which show mirror-like patterns. These results indicate that the metrics proposed to corroborate each other. Figure 6 focuses on the binary classification accuracy under various configurations. This metric describes the attacker's mean accuracy of distinguishing between 0-output and 1-output victims, given 8192 trials for both training and testing. Observe that in the general case, the classification accuracy remains at or close to 100% after a single reset, and drops significantly after the second reset. However, further resets show little impact on the classification accuracy, and may even increase it in some cases. Across all configurations tested, the mean accuracy on the testing set reaches a minimum of around 72% after four resets. This result demonstrates the effectiveness of the attack in recovering victim information leaked across reset operations.

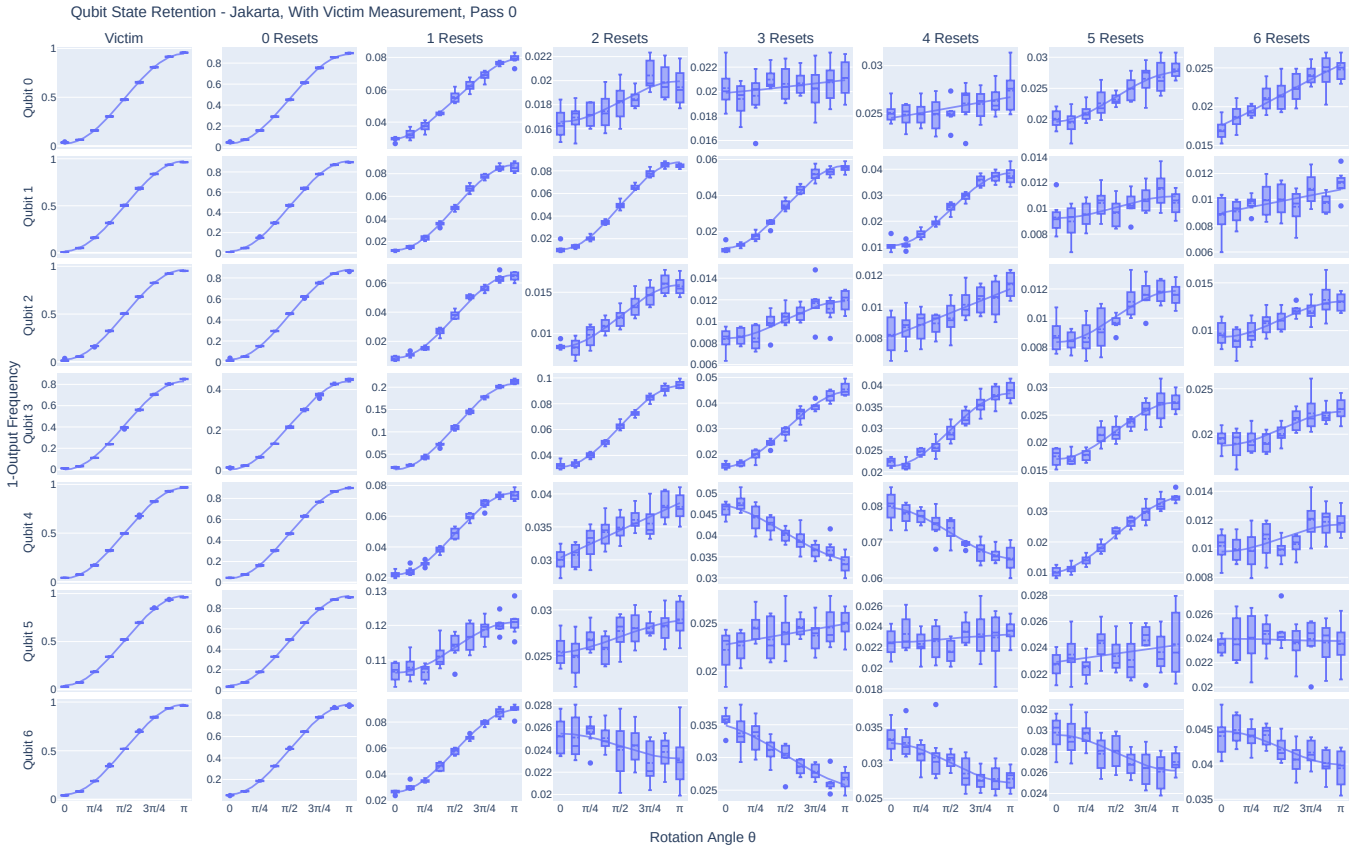
Finally, further extended testing up to 32 resets shown in Figure 7 reveals the large extent of victim state retention even after a large number of repeated resets. Observe that with a sufficient number of shots, the retained states from the 0-output and 1-output victims remain highly distinct in a large portion of the cases, including after 27. 29 and 31 repeated resets. this result further highlights the ineffectiveness of simple repetition as a means of securing reset operations against state retention.

6.6 Characterizing Crosstalk-Like Behavior

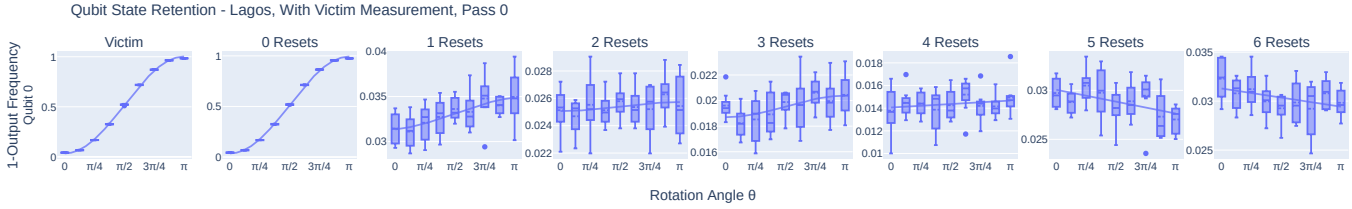
As shown in Figure 8, the 1-output frequency of measurement-free control groups (idle and H) follows an exponential decay pattern, while the measurement control group retains a constant frequency throughout. On the other hand, the experiment groups start at a constant frequency, with decays starting when the attacker measurements overlap with the victim operations. Interestingly, these starting points also roughly coincide with the intersections with the control groups. Combined, these results indicate that the decay does not start until the readout resonator becomes uncoupled with the victim qubit. By observing the start of the decay, the attacker can thus determine two important pieces of victim information: the duration between the initialization of a victim qubit and its last measurement operation, and the duration between the last measurement and the end of the victim's allocated share.

6.7 Buggy or Faulty Reset Operation on Perth

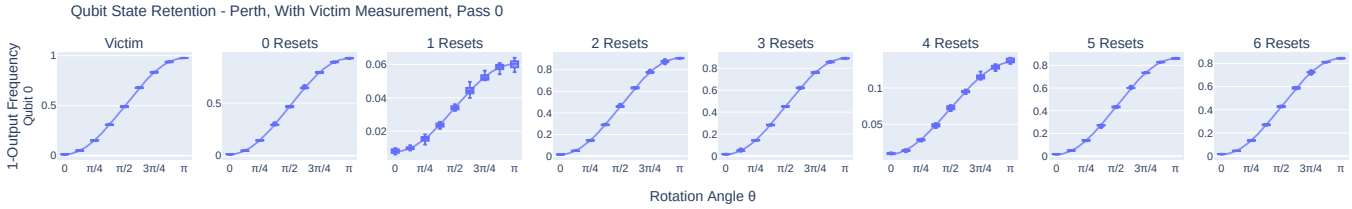
As shown in Figure 4c, results from the latest Perth backend are anomalous when the victim measurement is performed. For all qubits, the results exhibit no range compression for the majority of numbers of resets, negating the entire effect of reset operations.



(a) Jakarta all qubits reset channel $Q_m(\theta)$ with number of resets $m \in \{1, \ldots, 6\}$.



(b) Lagos q_0 reset channel $Q_m(\theta)$ with number of resets $m \in \{1, \ldots, 6\}$.



(c) Perth q_0 reset channel $Q_m(\theta)$ with number of resets $m \in \{1, \ldots, 6\}$.

Figure 4: Retention of victim qubit state in various scopes on the three backends. The rows depict different qubits. The first column shows the 1-output frequency of the victim measurement, and the proceeding columns show the attacker measurement frequency after various numbers of reset operations. Each panel is indexed by θ, and the error bars depict variations of the frequency in ϕ. For simplicity, only q_0 is shown for Lagos and Perth. The curves represent the best-fitting characterizations of the channels.

Figure 5: Training (blue) and testing (red) fidelity under various metrics on each qubit of Jakarta with victim measurement, with respect to the number of reset operations between the victim and the attacker. For the angle prediction loss, the error bars represent variations over θ.

Figure 6: Training (blue) and testing (red) binary prediction accuracy under various configurations, with respect to the number of resets between the victim and the attacker.

This error may be due to implementation-specific hardware optimizations IBM has taken to achieve high fidelity and speed on the device. We have disclosed this issue to IBM, and it is currently pending investigation.

7 SECURE RESET OPERATIONS

A secure reset operation could be an enabling technology that would allow sharing of a quantum computer among different users, or even among different quantum programs of the same user. A secure reset operation could allow for resetting a subset of qubits of the quantum computer between each user or program so that their state is erased, and a new program or user can use the qubits while computation continues on the other qubits. In this section we present the first proposal in literature for a secure reset operation.

7.1 Approaches to Secure Resets

In the case of repeated resets, if the attacker knows the number of reset operations that have been deterministically applied, then through training on the same machine and same qubit, they can almost always infer the θ angle from the output frequencies that they observe, except for the very unlikely case when the output frequency sigmoid curve is perfectly flat. Therefore, an entry point to designing secure resets would be to avoid using a deterministic number of reset operations. In the finest granularity, this calls for independent, per-shot randomization of the number of resets applied.

Based on observations from our evaluation of crosstalk like behavior in Section 6.6, an attacker running on qubits adjacent to the victim is able to identify the end of the victim's operations. In this case, the attacker may learn the number of resets used and refer to training data in order to recover the victim's output distribution. This opportunity for timing-based attacks requires that the duration of the reset sequence be constant, regardless of the actual number of resets used. As a result, given a set of possible number of resets to insert, the secure reset should prepend the reset sequence with sufficient number of delay gates, such that regardless of the number of resets inserted, the total length of the sequence is equal to the maximum length corresponding to the maximum number of resets that could be inserted.

We again consider a strong attacker who can run a large number of shots. Within each shot, the attacker can get the victim to produce the same output distribution, while each time being conveniently co-located with the victim and able to operate on the victim's qubit after the provider-inserted reset sequence. We restrict the victim to output either all $|0\rangle$ or $|1\rangle$. Through repeated measurements, the attacker attempts to tell apart 0-output and 1-output victims. Furthermore, the attacker knows the possible numbers of resets in use in the randomization scheme, as well as their probabilities of being selected.

Since the attacker is aware of the randomization scheme, they are able to perform training with each candidate number of resets,

Figure 7: Retention of the victim qubit state in extended higher-fidelity testing with 65536 shots and up to 32 resets for the Lagos backend. The error bars correspond to one binomial standard deviation around the data points.

and derive the expected distributions of 1-output frequency conditioned on a 0-output or 1-output victim. Therefore, if the expected distributions of 1-output frequency in attacker measurements corresponding to the two victim cases are distinguishable, then attacker can eventually distinguish the cases with enough shots. In light of this, when deriving the randomization scheme, we wish to minimize the difference in the expected attacker-side 1-output frequency distributions between the victim cases. Therefore, we select the Kullback–Leibler (K-L) divergence between the distributions as an important security parameter for randomization schemes.

7.2 Secure Reset Design

Based on the design discussed above, we design the secure reset operation as follows. For each quantum computer backend and each qubit, there is a set $\mathbb{X}$ of the possible number of resets that should be applied to a particular qubit after victim finished. The set contains at most two elements. Given a budget on the maximum number of resets, the two chosen numbers of resets and their respective probability are selected based on constrained optimization that minimizes the expected K-L divergence security parameter. Empirically, given a budget of at most r resets, the provider obtains the K-L divergence for each $i \in [1, r]$ via experiments. Afterward, the optimization can be performed via enumeration in $O(n^2)$ time. When more than one option for $\mathbb{X}$ eliminate the divergence, we set the tie-breaking condition to minimize the expected attacker 1-output probability. Overall, finding the optimal $\mathbb{X}$ requires $O(r)$ online time (i.e., time required for quantum computer operations) and $O(r^2)$ offline time.

Once the set $\mathbb{X}$ and the probability distribution is established, then each time one shot of a circuit is executed on a quantum computer backend on the corresponding qubit. A random number of x of resets, drawn from $\mathbb{X}$ and its corresponding distribution, is inserted.

Finally, a padding of $p = \max(\mathbb{X}) - x$ idle delays needs to be inserted. Each delay has to have same timing as one reset operation. As result, regardless the number x selected, $x + p$ is always going to equal $max(\mathbb{X})$ and attacker will not be able to use the crosstalk-like behavior to guess the number x of resets used.

Figure 9 shows an example diagram of the secure rest. In this example, the $\mathbb{X}$ contains 4 and 7 for the two possible numbers of resets to be used. As can be seen from the diagram, each number of resets is selected with some probability p and $1 - p$. For each shot, i.e., circuit execution, again the strong attacker is always located after the victim, but the number of resets would be each time randomly drawn from $\mathbb{X}$. For each shot then a random value needs to be generated. Uniform random values could be generated and then converted to target distribution using rejection sampling or other methods.

7.3 Secure Reset Evaluation

To illustrate the performance of the proposed secure reset scheme, we test the aforementioned scheme on the q_0 qubit of the Lagos backend. Note that on this device/qubit, a single reset takes ~ 1 μs, and a full-system wipe takes ~ 1000 μs. Specifically, we focus on two performance metrics: security and fidelity. We compare the proposed scheme against idle thermalization and repeated resets. Recall that the full-system wipe is implemented via long idle thermalization sequences.

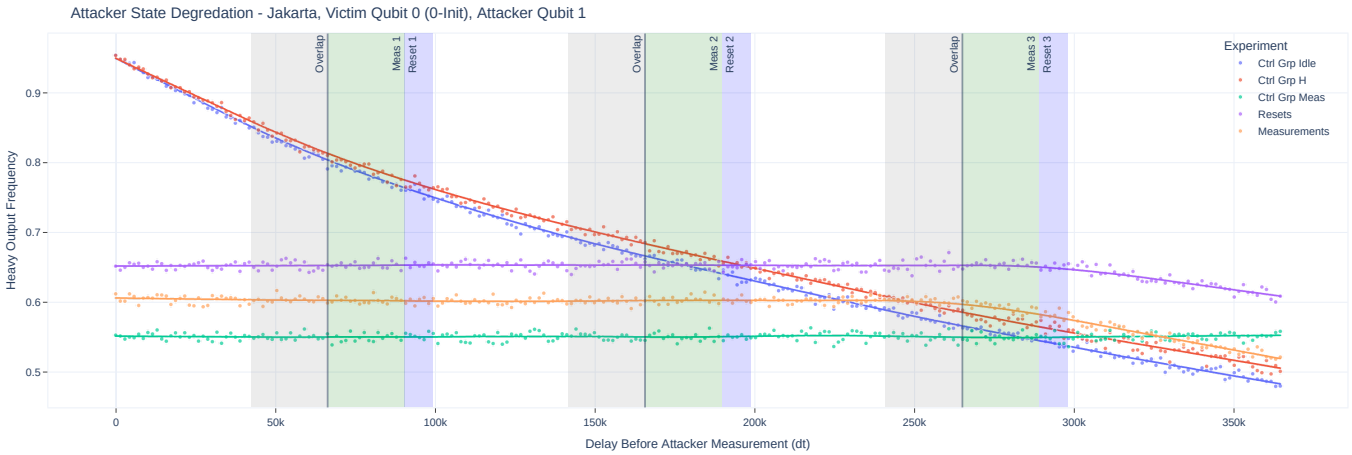
In terms of security, we examine the expected K-L divergence of attacker-side 1-output distributions between 0-output and 1-output victims. In Figure 10, observe that the divergence decays very slowly for thermalization, and heavily oscillates in the case of repeated resets. Repeated resets are incapable of sustaining a divergence level below the full-system wipe value, even when a large time budget is in use. In contrast, the secure reset eliminates the expected divergence when there exists time budget for at least 3 resets (~ 3 μs).

In terms of fidelity, we consider the attacker-side 1-output frequency conditioned on an 1-output victim. This corresponds to a worst-case scenario, as the reset operation needs to reinitialize all $|1\rangle$ states into $|0\rangle$. Again, as shown in Figure 11, both repeated resets and the secure reset outperform thermalization. However, the repeated resets exceeds the full-system wipe value within a large portion of the tested domain. The secure reset continues to outperform the alternatives, as it maintains a stable 1-output frequency below the reference value when there exists time budget for at least 2 resets (~ 2 μs).

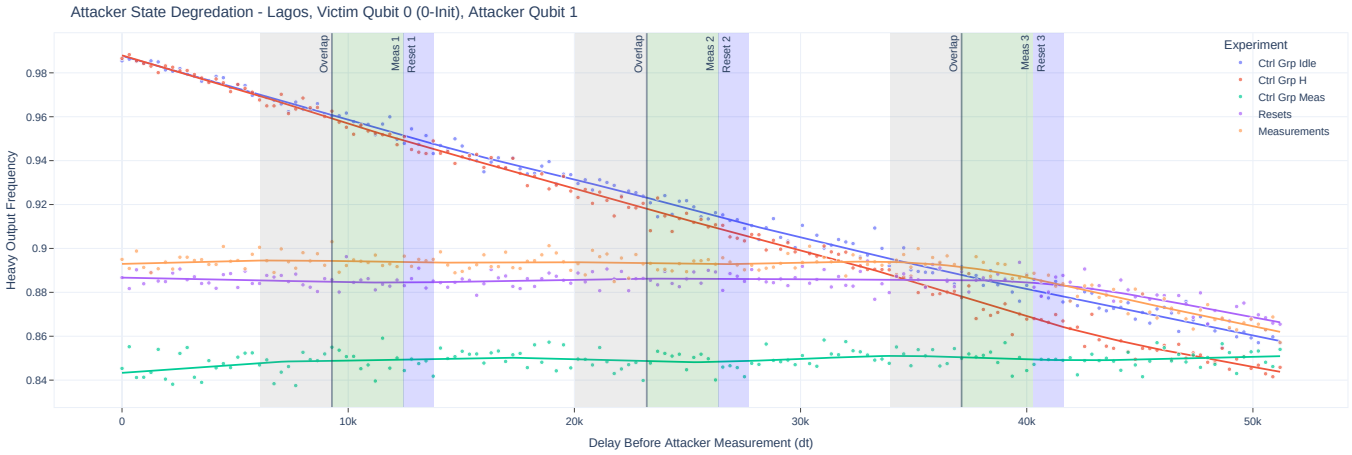
Overall, the proposed secure reset scheme performs significantly better in both aspects than repeated resets and thermalization. The time budget required for it to exceed the full-system wipe typical values is ~ 3 μs, which is over 300 times shorter than the full-system wipe (~ 1000 μs).

7.4 Secure Reset Takeaways

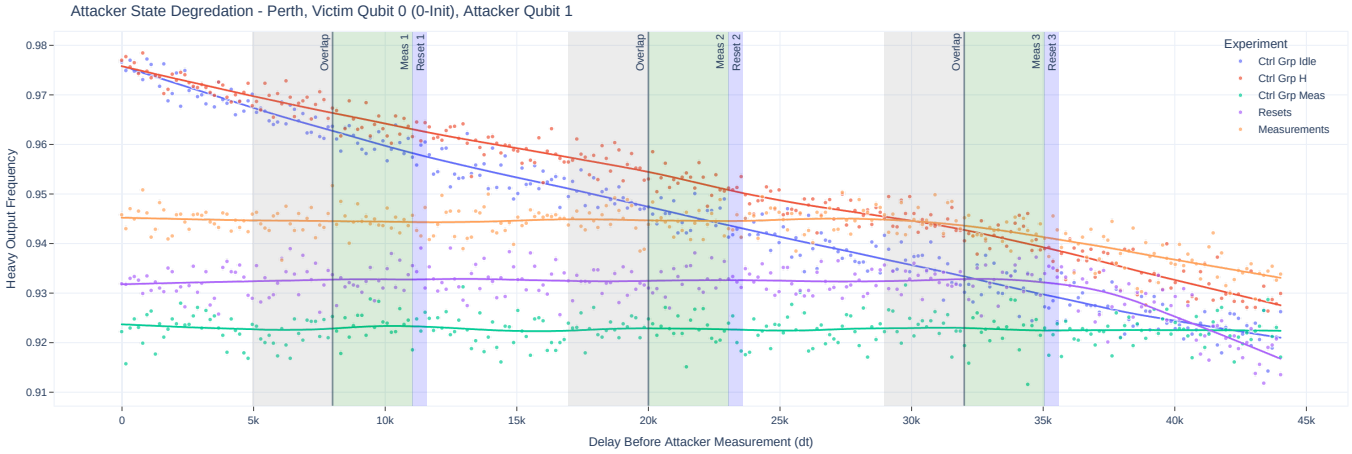
A broader security audience may take away a number of ideas from the secure reset operation design, implementation, and evaluation.



(a) Jakarta, attacker q_1 and victim q_0.



(b) Lagos, attacker q_1 and victim q_0.



(c) Perth, attacker q_1 and victim q_0.

Figure 8: Attacker 1-output degradation as a function of delay prior to attacker measurement, given $|0\rangle$-initialized q_0 as victim and $|1\rangle$-initialized q_1 as attacker. The colored regions represent overlaps between attacker measurement and victim measurement/reset.

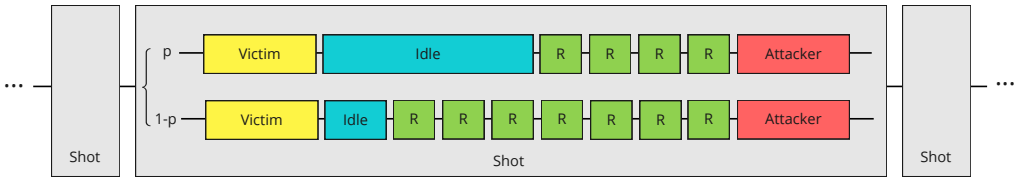


Figure 9: Secure reset operation block diagram for one qubit showing two possible reset sequences corresponding to p and $1-p$ probabilities. The probabilities and the number of reset operations for either option are determined empirically for each qubit and each machine to eliminate the K-L divergence between 0/1-output victims. The diagram is not drawn to scale.

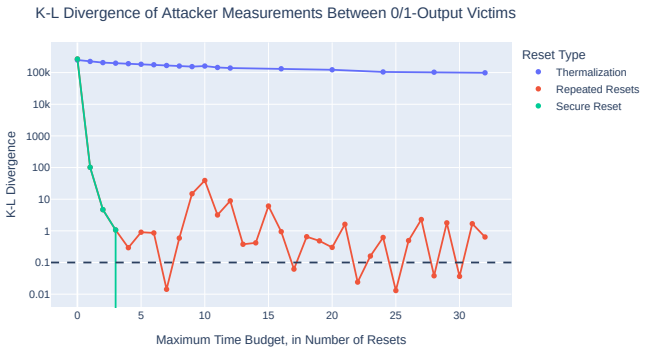


Figure 10: Expected K-L divergence of attacker measurements between the two victim cases. The horizontal dashed line represents an experimentally-derived typical value of 0.1 corresponding to a full-system wipe (i.e., thermalization of ~ 1000 μs).

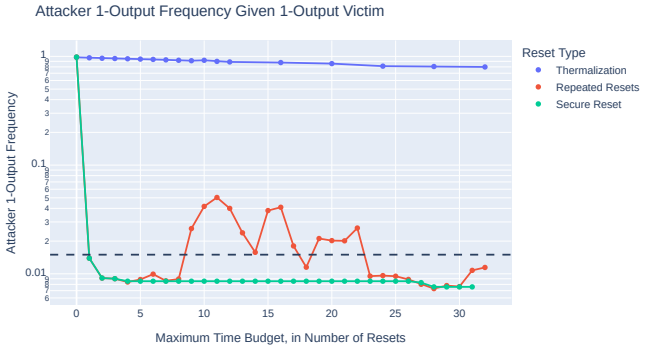


Figure 11: Attacker-side 1-output frequency conditioned on an 1-output victim. The horizontal line represents a typical value of 0.015 drawn from current device calibration data obtained via Qiskit[1].

For example, unique properties of the hardware need to be considered and the defense needs to be implemented by considering particular qubit's or machine's behavior, thus the security is intertwined with the physical hardware properties. Further, side-channels such as the crosstalk-like behavior can leak information about the defense (i.e., leak the number of resets used) so the defense needs to account for the side channels. The defense in practice ends up following the well-established constant-time principle where the operation (here the secure reset operation) takes a fixed amount of time regardless of the machine, qubit, or number of resets used. The padding with delays is used to achieve this.

The proposed secure reset is characterized by the K-L divergence (for security) and the attacker 1-output frequency (for fidelity). From our evaluation, the secure reset is capable of surpassing typical values of the full-system wipe with a maximum time budget of 3 resets or ~ 3 μs, providing a $\sim$ 300x speedup over the full-system wipe. This establishes the proposed scheme as an attractive alternative to the full-system wipe, even outside sharing or multitenacy settings.

For this work we have focused only on timing and crosstalk-like channels when considering how the secure reset itself could be attacked. Interesting orthogonal research may emerge on power side-channels to detect reset vs. delay operations.

8 RELATED WORK

This section lists related work on multi-programming and shared NISQ quantum computers, crosstalk and noise mitigation, and existing security work in this area.

8.1 Multi-programming and Shared NISQ Quantum Computers

The ideas about multi-programming and shared NISQ quantum computers now start to emerge. A recent work [6] has explored multi-programming of quantum computers as a way to better utilize the resources of NISQ quantum computers. The proposed approach [6] is similar to batch processing from classical computers, where multiple programs are scheduled at the same time, and run in parallel on a quantum computer. The authors showed how to fairly allocate reliable qubits to the different programs. They also showed how to adjust the start time of each program so they all end up at about the same time and thus minimize measurement errors for final measurements which are performed together at the end. And the work considered a run-time monitor to switch to single-program execution if the reliability impact of multi-programming is greater than a predefined threshold. These ideas could be used to allow multiple users to share the quantum computer, but authors did not explore any security considerations explicitly, but do mention "signals applied to one qubit can leak on to the other qubits causing unwarranted fluctuations in their quantum states" [6].

Another work [13] also explored how to partition physical qubits among concurrent quantum programs, with the goal of avoiding the waste of resources. The work also proposed a compilation task scheduler that schedules concurrent quantum programs to be

compiled and executed based on estimated fidelity. This approach is also similar to batch processing in classical computers where many programs are compiled together and scheduled to execute on the target machine together. Authors mention crosstalk noise caused by simultaneously executed quantum gates as one challenge, but focus on reliability and errors due to crosstalk rather than intentional information leaks.

8.2 Crosstalk and Noise Mitigation

As crosstalk is now well-known to cause noise and even errors, a number of papers have focused on mitigation of crosstalk from the reliability perspective. To the best of our knowledge, the mitigation techniques almost always analyze only one program and consequently assume that all the code is available for analysis before runtime.

Within a single quantum program, recent work [16] has analyzed when multiple instructions executed in parallel, how the crosstalk between the instructions can corrupt the quantum state and lead to incorrect program execution. The authors proposed a software (scheduling) based reliability solution via instruction scheduling which serializes instruction pairs that could be affected by crosstalk if they were executed in parallel while letting other instruction pairs to be less affected by crosstalk executed in parallel. The work targeted IBM Q machines, which have fixed qubit and fixed coupler designs. This approach could be applied to multi-programmed or shared NISQ quantum computer setting but would require parallel analysis of programs from different users. Also, in a dynamic, shared setting, the programs running in parallel can change in real time, making static analysis at compile time difficult to extend to the shared setting. There is also work [30] demonstrating that the dressing from qubit-qubit coupling can cause significant cross-driving errors if the qubits operate at the near frequency collision regions due to the crosstalk.

Authors of separate work [7] recognized the unwanted crosstalk between neighboring qubits due to a phenomenon called frequency crowding as one major source of gate error. They also proposed to trade parallelism for higher gate fidelity when necessary, but targeted tunable qubits and fixed coupler quantum computers, presenting results in IBM Qiskit software [5] as such machines are not available to research for remote access. The tunable qubits allow the researchers to adjust frequencies of the qubits most affected by crosstalk based on software analysis and adjust their frequencies. This approach could also be extended to multi-programmed or shared NISQ quantum computers, but requires analysis of attacker programs.

8.3 Information Leakage in Quantum Computers

Recent work characterized crosstalk in NISQ quantum computers using idle tomography and simultaneous randomized benchmarking [3]. The work focused on enabling the simulation of quantum circuits by including experimental crosstalk error rates, so that the simulation better reflects real devices, compared to simulations that only consider gate errors. Another work [2] has present a crosstalk modeling analysis framework for near-term quantum computers after extracting the error rates experimentally from IBM Q quantum computers. The authors also proposed adversarial fault injection using crosstalk in a multi-programming environment where the victim and the adversary share the same quantum hardware, as well as create repeated shuttle operations to increase quantum bit energy and degrade the reliability of computations (fidelity) for constructing adversarial program [23]. There is also work [15] that demonstrates crosstalk-induced errors of NISQ quantum computers can perform idle tomography-based fingerprinting. The prediction accuracy of the device- and location-specific fingerprinting results can be higher than 95%. On the other hand, obfuscation of quantum circuits [25] is developed to hide the functionality of using reverse engineering to extract sensitive parameters, e.g., circuit topology, program, and its properties for the quantum circuit through untrusted third-party compilers.

In recent research [20], authors present an overview of various noise sources and their impact on the resilience and the security of quantum circuits. The authors considered fault-injection attacks and information leakage. In fault-injection attacks, the adversary may be interested in launching a denial-of-service (DoS) attack by corrupting the victim's computational outcome. In information leakage, the readout of qubits shows state-dependent error probability. Since error rates are correlated among qubits, an adversary can exploit this property to sense a victim's output by reading out his or her qubits whose readout state is affected by nearby qubits of the victim [20].

Other recent, but not peer-reviewed work [22], also analyzed readout or measurement error and how it can leak information. Authors used this to sense victim output which may contain sensitive information. During the attack, the adversary can only read his or her qubit, whose output depends on the state of the victim's nearby qubits.

Another work [17] proposed Quantum Physically Unclonable Functions (QuPUFs) based on superposition or based on decoherence. The QuPUF were proposed to address the problem of identifying quantum computer hardware to find out, for example, if an untrustworthy provider allocated less-reliable quantum computers to users to save money or resources. The QuPUF responses can be used to identify the hardware and establish the "identity (trust) of a quantum computer" [17]. The work explored only a very simple QuPUF design based on readout error or one-qubit gate error. The evaluation only considered two older IBM Q machines.

Last but not least, a very recent, but not peer-reviewed, survey [21] summarizes number of additional security ideas in quantum computers, including limited connectivity, gate error, loss of qubit states, readout error, and crosstalk that can be used in a fault-injection attack.

9 CONCLUSION

In this work, we examined how the reset operation enables the sharing of cloud-based quantum computers. Currently, reset operations are approximately 1000 times faster than a full context wipe. Yet, we discovered that they also come with significant security issues. We demonstrated how information can be leaked across reset gates on the same qubit, and how reset gates emit information via crosstalk to adjacent qubits. We highlighted the ineffectiveness of deterministic repeated resets in enhancing security, and proposed a scheme

for secure reset operations. Compared to the full-system wipe, the proposed scheme attains higher security and fidelity in empirical testing, while simultaneously achieving a ∼ 300x speedup. The secure resets may significantly benefit the deployment and use of shared, cloud-based quantum computers, especially in multitenant scenarios.

ACKNOWLEDGMENTS

The authors would like to thank IBM and Yale University for providing access to IBM's superconducting devices. This work was supported in part by NSF grant 1901901. Shuwen Deng was supported through the Google PhD Fellowship. The authors would like to thank Chuanqi Xu for helpful discussions.

REFERENCES

[1] MD SAJID ANIS, Héctor Abraham, AduOffei, Rochisha Agarwal, Gabriele Agliardi, Merav Aharoni, Ismail Yunus Akhalwaya, Gadi Aleksandrowicz, Thomas Alexander, Matthew Amy, Sashwat Anagolum, Eli Arbel, Abraham Asfaw, Anish Athalye, Artur Avkhadiev, Carlos Azaustre, PRATHAMESH BHOLE, Abhik Banerjee, Santanu Banerjee, Will Bang, Aman Bansal, Panagiotis Barkoutsos, Ashish Barnawal, George Barron, George S. Barron, Luciano Bello, Yael Ben-Haim, Daniel Bevenius, Dhruv Bhatnagar, Arjun Bhobe, Paolo Bianchini, Lev S. Bishop, Carsten Blank, Sorin Bolos, Soham Bopardikar, Samuel Bosch, Sebastian Brandhofer, Brandon, Sergey Bravyi, Nick Bronn, Bryce-Fuller, David Bucher, Artemiy Burov, Fran Cabrera, Padraic Calpin, Lauren Capelluto, Jorge Carballo, Ginés Carrascal, Adam Carriker, Ivan Carvalho, Adrian Chen, Chun-Fu Chen, Edward Chen, Jielun (Chris) Chen, Richard Chen, Franck Chevallier, Rathish Cholarajan, Jerry M. Chow, Spencer Churchill, Christian Claus, Christian Clauss, Caleb Clothier, Romilly Cocking, Ryan Cocuzzo, Jordan Connor, Filipe Correa, Abigail J. Cross, Andrew W. Cross, Simon Cross, Juan Cruz-Benito, Chris Culver, Antonio D. Córcoles-Gonzales, Navaneeth D, Sean Dague, Tareq El Dandachi, Animesh N Dangwal, Jonathan Daniel, Marcus Daniels, Matthieu Dartiailh, Abdón Rodríguez Davila, Faisal Debouni, Anton Dekusar, Amol Deshmukh, Mohit Deshpande, Delton Ding, Jun Doi, Eli M. Dow, Eric Drechsler, Eugene Dumitrescu, Karel Dumon, Ivan Duran, Kareem EL-Safty, Eric Eastman, Grant Eberle, Amir Ebrahimi, Pieter Eendebak, Daniel Egger, Emilio, Alberto Espiricueta, Mark Everitt, Davide Facoetti, Farida, Paco Martín Fernández, Samuele Ferracin, Davide Ferrari, Axel Hernández Ferrera, Romain Fouilland, Albert Frisch, Andreas Fuhrer, Bryce Fuller, MELVIN GEORGE, Julien Gacon, Borja Godoy Gago, Claudio Gambella, Jay M. Gambetta, Adhisha Gammanpila, Luis Garcia, Tanya Garg, Shelly Garion, Tim Gates, Leron Gil, Austin Gilliam, Aditya Giridharan, Juan Gomez-Mosquera, Gonzalo, Salvador de la Puente González, Jesse Gorzinski, Ian Gould, Donny Greenberg, Dmitry Grinko, Wen Guan, John A. Gunnels, Harshit Gupta, Naman Gupta, Jakob M. Günther, Mikael Haglund, Isabel Haide, Ikko Hamamura, Omar Costa Hamido, Frank Harkins, Areeq Hasan, Vojtech Havlicek, Joe Hellmers, Łukasz Herok, Stefan Hillmich, Hiroshi Horii, Connor Howington, Shaohan Hu, Wei Hu, Junye Huang, Rolf Huisman, Haruki Imai, Takashi Imamichi, Kazuaki Ishizaki, Ishwor, Raban Iten, Toshinari Itoko, Alexander Ivrii, Ali Javadi, Ali Javadi-Abhari, Wahaj Javed, Qian Jianhua, Madhav Jivrajani, Kiran Johns, Scott Johnstun, Jonathan-Shoemaker, JosDenmark, JoshDumo, John Judge, Tal Kachmann, Akshay Kale, Naoki Kanazawa, Jessica Kane, Kang-Bae, Annanay Kapila, Anton Karazeev, Paul Kassebaum, Josh Kelso, Scott Kelso, Vismai Khanderao, Spencer King, Yuri Kobayashi, Kovi11Day, Arseny Kovyrshin, Rajiv Krishnakumar, Vivek Krishnan, Kevin Krsulich, Prasad Kumkar, Gawel Kus, Ryan LaRose, Enrique Lacal, Raphaël Lambert, Haggai Landa, John Lapeyre, Joe Latone, Scott Lawrence, Christina Lee, Gushu Li, Jake Lishman, Dennis Liu, Peng Liu, Yunho Maeng, Saurav Maheshkar, Kahan Majmudar, Aleksei Malyshev, Mohamed El Mandouh, Joshua Manela, Manjula, Jakub Marecek, Manoel Marques, Kunal Marwaha, Dmitri Maslov, Paweł Maszota, Dolph Mathews, Atsushi Matsuo, Farai Mazhandu, Doug McClure, Maureen McElaney, Cameron McGarry, David McKay, Dan McPherson, Srujan Meesala, Dekel Meirom, Corey Mendell, Thomas Metcalfe, Martin Mevissen, Andrew Meyer, Antonio Mezzacapo, Rohit Midha, Zlatko Minev, Abby Mitchell, Nikolaj Moll, Alejandro Montanez, Gabriel Monteiro, Michael Duane Mooring, Renier Morales, Niall Moran, David Morcuende, Seif Mostafa, Mario Motta, Romain Moyard, Prakash Murali, Jan Müggenburg, David Nadlinger, Ken Nakanishi, Giacomo Nannicini, Paul Nation, Edwin Navarro, Yehuda Naveh, Scott Wyman Neagle, Patrick Neuweiler, Aziz Ngoueya, Johan Nicander, Nick-Singstock, Pradeep Niroula, Hassi Norlen, NuoWenLei, Lee James O'Riordan, Oluwatobi Ogunbayo, Pauline Ollitrault, Tamiya Onodera, Raul Otaolea, Steven Oud, Dan Padilha, Hanhee Paik, Soham Pal, Yuchen Pang, Ashish Panigrahi, Vincent R. Pascuzzi, Simone Perriello, Eric Peterson, Anna Phan, Francesco Piro, Marco Pistoia, Christophe Piveteau, Julia Plewa, Pierre Pocreau, Alejandro Pozas-Kerstjens, Rafał Pracht, Milos Prokop, Viktor Prutyanov, Sumit Puri, Daniel Puzzuoli, Jesús Pérez, Quintiii, Isha R, Rafey Iqbal Rahman, Arun Raja, Roshan Rajeev, Nipun Ramagiri, Anirudh Rao, Rudy Raymond, Oliver Reardon-Smith, Rafael Martín-Cuevas Redondo, Max Reuter, Julia Rice, Matt Riedemann, Drew Risinger, Marcello La Rocca, Diego M. Rodríguez, RohithKarur, Ben Rosand, Max Rossmannek, Mingi Ryu, Tharrmashastha SAPV, Arijit Saha, Abdullah Ash-Saki, Sankalp Sanand, Martin Sandberg, Hirmay Sandesara, Ritvik Sapra, Hayk Sargsyan, Aniruddha Sarkar, Ninad Sathaye, Bruno Schmitt, Chris Schnabel, Zachary Schoenfeld, Travis L. Scholten, Eddie Schoute, Mark Schulterbrandt, Joachim Schwarm, James Seaward, Sergi, Ismael Faro Sertage, Kanav Setia, Freya Shah, Nathan Shammah, Rohan Sharma, Yunong Shi, Jonathan Shoemaker, Adenilton Silva, Andrea Simonetto, Divyanshu Singh, Parmeet Singh, Phattharaporn Singkanipa, Yukio Siraichi, Siri, Jesús Sistos, Iskandar Sitdikov, Seyon Sivarajah, Magnus Berg Sletfjerding, John A. Smolin, Mathias Soeken, Igor Olegovich Sokolov, Igor Sokolov, Vicente P. Soloviev, SooluThomas, Starfish, Dominik Steenken, Matt Stypulkoski, Adrien Suau, Shaojun Sun, Kevin J. Sung, Makoto Suwama, Oskar Słowik, Hitomi Takahashi, Tanvesh Takawale, Ivano Tavernelli, Charles Taylor, Pete Taylour, Soolu Thomas, Mathieu Tillet, Maddy Tod, Miroslav Tomasik, Enrique de la Torre, Juan Luis Sánchez Toural, Kenso Trabing, Matthew Treinish, Dimitar Trenev, TrishaPe, Felix Truger, Georgios Tsilimigkounakis, Davindra Tulsi, Wes Turner, Yotam Vaknin, Carmen Recio Valcarce, Francois Varchon, Adish Vartak, Almudena Carrera Vazquez, Prajjwal Vijaywargiya, Victor Villar, Bhargav Vishnu, Desiree Vogt-Lee, Christophe Vuillot, James Weaver, Johannes Weidenfeller, Rafal Wieczorek, Jonathan A. Wildstrom, Jessica Wilson, Erick Winston, WinterSoldier, Jack J. Woehr, Stefan Woerner, Ryan Woo, Christopher J. Wood, Ryan Wood, Steve Wood, James Wootton, Matt Wright, Bo Yang, Daniyar Yeralin, Ryota Yonekura, David Yonge-Mallo, Richard Young, Jessie Yu, Lebin Yu, Christopher Zachow, Laura Zdanski, Helena Zhang, Christa Zoufal, aeddins ibm, alexzhang13, b63, bartek bartlomiej, bcamorrison, brandhsn, catornow, charmerDark, deeplokhande, dekel.meirom, dime10, ehchen, fanizzamarco, fs1132429, gadial, galeinston, georgezhou20, georgios ts, gruu, hhorii, hykavitha, itoko, jliu45, jscott2, klinvill, krutik2966, ma5x, michelle4654, msuwama, ntgiwsvp, ordmoj, sagar pahwa, pritamsinha2304, ryancocuzzo, saswati qiskit, septembrr, sethmerkel, shaashwat, sternparky, strickroman, tigerjack, tsura crisaldo, vadebayo49, welien, willhbang, wmurphy collabstar, yang.luh, and Mantas Čepulkovskis. 2021. Qiskit: An Open-source Framework for Quantum Computing. https://doi.org/10.5281/zenodo.2573505

[2] Abdullah Ash-Saki, Mahabubul Alam, and Swaroop Ghosh. 2020. Analysis of crosstalk in nisq devices and security implications in multi-programming regime. In *International Symposium on Low Power Electronics and Design (ISLPED)*. 25–30.

[3] Abdullah Ash-Saki, Mahabubul Alam, and Swaroop Ghosh. 2020. Experimental characterization, modeling, and analysis of crosstalk in a quantum computer. *Transactions on Quantum Engineering* 1 (2020), 1–6.

[4] Juan I Cirac and Peter Zoller. 1995. Quantum computations with cold trapped ions. *Physical Review Letters* 74, 20 (1995), 4091.

[5] Andrew Cross. 2018. The IBM Q experience and QISKit open-source quantum computing software. In *APS March Meeting Abstracts*, Vol. 2018. L58–003.

[6] Poulami Das, Swamit S Tannu, Prashant J Nair, and Moinuddin Qureshi. 2019. A case for multi-programming quantum computers. In *International Symposium on Microarchitecture (MICRO)*. 291–303.

[7] Yongshan Ding, Pranav Gokhale, Sophia Fuhui Lin, Richard Rines, Thomas Propson, and Frederic T Chong. 2020. Systematic crosstalk mitigation for superconducting qubits via frequency-aware compilation. In *International Symposium on Microarchitecture (MICRO)*. 201–214.

[8] Jay Gambetta. 2020. IBM's Roadmap For Scaling Quantum Technology.

[9] Ilias Giechaskiel, Kasper Bonne Rasmussen, and Jakub Szefer. 2020. C3APSULe: Cross-FPGA Covert-Channel Attacks through Power Supply Unit Leakage. In *Symposium on Security and Privacy (S&P)*. 1728–1741.

[10] Lov K. Grover. 1996. A Fast Quantum Mechanical Algorithm for Database Search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, Gary L. Miller (Ed.). ACM, 212–219. https://doi.org/10.1145/237814.237866

[11] Jonathan A Jones, Michele Mosca, and Rasmus H Hansen. 1998. Implementation of a quantum search algorithm on a quantum computer. *Nature* 393, 6683 (1998), 344–346.

[12] Benjamin P Lanyon, James D Whitfield, Geoff G Gillett, Michael E Goggin, Marcelo P Almeida, Ivan Kassal, Jacob D Biamonte, Masoud Mohseni, Ben J Powell, Marco Barbieri, et al. 2010. Towards quantum chemistry on a quantum computer. *Nature Chemistry* 2, 2 (2010), 106–111.

[13] Lei Liu and Xinglei Dou. 2021. QuCloud: A New Qubit Mapping Mechanism for Multi-programming Quantum Computing in Cloud Environment. In *International Symposium on High-Performance Computer Architecture (HPCA)*. 167–178.

[14] N David Mermin. 2007. *Quantum computer science: an introduction.* Cambridge University Press.

[15] Allen Mi, Shuwen Deng, and Jakub Szefer. 2021. Device- and Locality-Specific Fingerprinting of Shared NISQ Quantum Computers. In *Workshop on Hardware and Architectural Support for Security and Privacy (HASP)*.

[16] Prakash Murali, David C McKay, Margaret Martonosi, and Ali Javadi-Abhari. 2020. Software mitigation of crosstalk on noisy intermediate-scale quantum computers. In *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. 1001–1016.

[17] Koustubh Phalak, Abdullah Ash-Saki, Mahabubul Alam, Rasit Onur Topaloglu, and Swaroop Ghosh. 2021. Quantum PUF for Security and Trust in Quantum Computing. *Journal on Emerging and Selected Topics in Circuits and Systems* 11, 2 (2021), 333–342.

[18] John Preskill. 2018. Quantum computing in the NISQ era and beyond. *Quantum* 2 (2018), 79.

[19] Chad Rigetti, Jay M Gambetta, Stefano Poletto, Britton LT Plourde, Jerry M Chow, Antonio D Córcoles, John A Smolin, Seth T Merkel, Jim R Rozen, George A Keefe, et al. 2012. Superconducting qubit in a waveguide cavity with a coherence time approaching 0.1 ms. *Physical Review B* 86, 10 (2012), 100506.

[20] Abdullah Ash Saki, Mahabubul Alam, and Swaroop Ghosh. 2021. Impact of Noise on the Resilience and the Security of Quantum Computing. In *International Symposium on Quality Electronic Design (ISQED)*. 186–191.

[21] Abdullah Ash Saki, Mahabubul Alam, Koustubh Phalak, Aakarshitha Suresh, Rasit Onur Topaloglu, and Swaroop Ghosh. 2021. A Survey and Tutorial on Security and Resilience of Quantum Computing. *arXiv preprint arXiv:2106.06081* (2021).

[22] Abdullah Ash Saki and Swaroop Ghosh. 2021. Qubit Sensing: A New Attack Model for Multi-programming Quantum Computing. *arXiv preprint arXiv:2104.05899* (2021).

[23] Abdullah Ash Saki, Rasit Onur Topaloglu, and Swaroop Ghosh. 2021. Shuttle-Exploiting Attacks and Their Defenses in Trapped-Ion Quantum Computers. *arXiv preprint arXiv:2108.01054* (2021).

[24] Peter W. Shor. 1999. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* 41, 2 (1999), 303–332.

[25] Aakarshitha Suresh, Abdullah Ash Saki, Mahabubul Alam, Dr Ghosh, et al. 2021. A Quantum Circuit Obfuscation Methodology for Security and Privacy. *arXiv preprint arXiv:2104.05943* (2021).

[26] Jakub M Szefer. 2013. *Architectures for secure cloud computing servers*. Ph. D. Dissertation. Princeton University.

[27] Shanquan Tian and Jakub Szefer. 2019. Temporal Thermal Covert Channels in Cloud FPGAs. In *International Symposium on Field-Programmable Gate Arrays (FPGA)*.

[28] Shanquan Tian, Wenjie Xiong, Ilias Giechaskiel, Kasper Rasmussen, and Jakub Szefer. 2020. Fingerprinting Cloud FPGA Infrastructures. In *International Symposium on Field-Programmable Gate Arrays (FPGA)*.

[29] Zhenyu Wu, Zhang Xu, and Haining Wang. 2014. Whispers in the hyper-space: high-bandwidth and reliable covert channel attacks inside the cloud. *Transactions on Networking* 23, 2 (2014), 603–615.

[30] Peng Zhao, Kehuan Linghu, Zhiyuan Li, Peng Xu, Ruixia Wang, Guangming Xue, Yirong Jin, and Haifeng Yu. 2021. Quantum crosstalk analysis for simultaneous gate operations on superconducting qubits. *arXiv preprint arXiv:2110.12570* (2021).