Securing Quantum Computer Reset with One-Time Pads

Chuanqi Xu
Department of Electrical & Computer Engineering
Yale University
New Haven, CT, USA
chuanqi.xu@yale.edu

Jamie Sikora
Department of Computer Science
Virginia Tech
Blacksburg, VA, USA
sikora@vt.edu

Jakub Szefer
Department of Electrical and Computer Engineering
Northwestern University
Evanston, IL, USA
jakub.szefer@northwestern.edu

Abstract

The rapid expansion of cloud-based access to quantum computers has significantly democratized their usage, enabling a more diverse range of users to explore and utilize quantum computing technologies. However, this increased accessibility also introduces security and privacy concerns. Cloud-based access and sharing of quantum computers require secure means to isolate different users, such as through the use of reset operations. However, current reset operations, including direct thermalization and fast reset instructions, are vulnerable to information leakage due to imperfections in quantum computer operations. To counteract these vulnerabilities, our work proposes multiple implementations of the one-time pad (OTP) defense mechanism. These implementations, specifically *random execution*, *dynamic circuit*, and *control gate*, involve applying Pauli or control gates randomly before executing standard reset operations. We analyze and compare these implementations in detail, demonstrating their effectiveness in mitigating state leakage. This work offers innovative approaches to enhancing the security of reset operations and the safety of cloud-based quantum computers.

CCS Concepts

• **Security and privacy → Hardware-based security protocols**; *Trusted computing*; • **Hardware → Quantum error correction and fault tolerance**.

Keywords

Quantum Computing, Computer Security, Qubit Reset, One-time Pad, Information Leakage

ACM Reference Format:
Chuanqi Xu, Jamie Sikora, and Jakub Szefer. 2025. Securing Quantum Computer Reset with One-Time Pads. In *Proceedings of the 2025 Quantum Security and Privacy Workshop (QSec '25), October 13–17, 2025, Taipei, Taiwan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3733825.3765278

1 Introduction

The concept of Noisy Intermediate-Scale Quantum (NISQ) characterizes the current generation of quantum computers [20]. These NISQ devices have promising applications in fields such as natural sciences and optimization [15, 18]. Further, the evolution of quantum computers is rapid. For instance, 1121-qubit machines are now operational and quantum computers with 200 qubits capable of running 100 million gates are expected before 2030 [14].

The availability of quantum computers from diverse manufacturers via cloud-based platforms such as IBM Quantum, Amazon Braket, and Microsoft Azure, has further revolutionized accessibility, eliminating the need for physical ownership and maintenance. Yet, this ease of access introduces significant security and privacy challenges. Malicious users could exploit this openness to infer sensitive information of others. One source of such leakage is noise and errors, such as in qubit resetting, which is necessary between circuit executions. These noisy and erroneous resets can inadvertently leak information to subsequent runs, presenting a vulnerability that has been exploited in various attacks, such as reset attacks [19, 22], side-channel attacks [6], and higher-energy state attacks [23, 24, 26, 28]. This leakage, which we name "horizontal" leakage, involves sequential information transfer from earlier to later executions. On the other hand, "vertical" leakage occurs simultaneously across qubits, and it is another form of vulnerability, as evidenced in crosstalk attacks [1, 2, 9, 10] and qubit sensing [4].

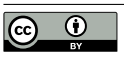
Quantum algorithms are inherently probabilistic, often requiring many executions to yield reliable outcomes. To ensure each execution starts from a predetermined ground state and to prevent interference of previous results, a reliable reset mechanism between iterations is essential. However, noise and errors prevalent in quantum computers, particularly within these reset mechanisms, can not only introduce systematic errors in computations because of the dependence on the computational results, but also create security and privacy risks through information leakage.

To mitigate the "horizontal" leakage, a previous work proposed to use the one-time pad (OTP) [27]. The idea of the cryptography technique is to XOR, or *pad*, the plaintext into a perfectly secure ciphertext. The authors thoroughly studied this problem and demonstrated that OTP can be used to mitigate the state leakage. However, they do not provide any insight into how OTP can be implemented in contemporary quantum computers, nor the influence of randomness on the effectiveness.

In this paper, we seek to propose practical implementations for both classical (COTP) and quantum (QOTP) one-time pads, tailored to current quantum computer architectures, while also being feasible to be incorporated in the future. Our goal is to demonstrate the feasibility and effectiveness of these implementations in mitigating "horizontal" leakage, as well as discuss the advantages and disadvantages of these implementations in depth.

2 Background

This section introduces the necessary topics of quantum computing and the one-time pad needed for this paper.



QSec '25, Taipei, Taiwan

ACM ISBN 979-8-4007-1913-4/2025/10
https://doi.org/10.1145/3733825.3765278

2.1 Quantum Computing Basics

Quantum states are often denoted in Dirac notation as $|\psi\rangle$. A qubit is a 2-dimensional quantum state which can be written as $|\psi\rangle = (\alpha, \beta)^T$ where $|\alpha|^2 + |\beta|^2 = 1$. In quantum circuits, qubits are controlled and evolve under quantum gates, which are unitary operations. At the end of a quantum circuit, the final state can be measured to get the computation results. According to Born's rule, the probability of measuring $|i\rangle$ is given by $P(|0\rangle) = |\alpha|^2$ and $P(|1\rangle) = |\beta|^2$. Moreover, the measurement leads to the collapse of the quantum state. A general representation of quantum states is the density matrix $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$. The probability of measuring $|i\rangle$ is given by $P(|i\rangle) = \langle i| \rho |i\rangle$. If $\rho = \frac{1}{n} I_n$, where I_n is the n-dimensional identity matrix, it is the maximally mixed state, and the probability of measuring any state $|i\rangle$ is $\frac{1}{n}$.

2.2 Classical One-Time Pad (COTP)

One-time pad (OTP) is widely used in cryptography [7]. The idea is to generate a random key with the same length as the plaintext. The ciphertext is generated by XORing, or padding, the plaintext with the key. Due to the property of XOR, the plaintext can be retrieved later by XORing the ciphertext with the key. COTP can be extended to quantum computing by padding the input with Pauli-X, which is similar to a bit-flip or NOT gate in classical computers. Whether Pauli-X is applied is based on a bit of a random number with the same probability to be 0 or 1. As a result, there is probability $\frac{1}{2}$ the state will be ρ and probability $\frac{1}{2}$ the state will be $X\rho X$, leading to $\rho' = \frac{1}{2}\rho + \frac{1}{2}X\rho X$. However, this direct extension is not secure. If $\rho = |0\rangle\langle 0|$ or $|1\rangle\langle 1|$, then ρ' is a maximally mixed state. But for other states, e.g., $\rho = |+\rangle\langle +|$, all items in ρ' are $\frac{1}{2}$, which only measures +1 along the X axis.

2.3 Quantum One-Time Pad (QOTP)

QOTP extends COTP by randomly applying both Pauli-X and Pauli-Z [8]. This requires two random bits: one for Pauli-X and one for Pauli-Z, which results in four cases with probability $\frac{1}{4}$, so the state is $\rho' = \frac{1}{4}\rho + \frac{1}{4}X\rho X + \frac{1}{4}Z\rho Z + \frac{1}{4}XZ\rho ZX$. It can be proved that ρ' is a maximally mixed state regardless of ρ, and thus all measurements yield the same results.

2.4 Execution Pattern of Cloud Quantum Computing

Due to quantum algorithms' probabilistic nature, a circuit is executed multiple times (or *shots*) to gather statistical results. Qubit resetting between shots, typically to the $|0\rangle$ state, is crucial for ensuring each shot starts consistently, facilitating accurate, repeatable measurements. Users submit jobs comprising multiple shots to the quantum hardware, which executes them and returns the aggregated results. This process highlights the importance of the reset mechanism in shot-based quantum computing.

3 Threat Model

In order to provide strong assurances about the security of our defense, we assume a powerful attacker. In particular, we assume that quantum computers can be shared, allowing for the alternate execution of circuits from various users. We assume part of the shots of quantum computing jobs, or even individual shots, from different users can alternate execution on a set of qubits, leading to possible attacks leveraging "horizontal" information leakage. A similar counterpart in classical computing is the sharing of the central processing unit (CPU) or memory among different programs or users, which is a universal standard in modern classical architecture.

The purpose of this study is to show how to safely reset the qubits so that an attacker cannot gain any information about the victim. We assume that the victim quantum program operates on certain quantum computer qubits. On the same qubits that the victim utilized, a powerful attacker can execute their circuit after the victim's circuit. We assume that the victim's qubits are reset before the attacker can utilize them, and that the attacker cannot modify the reset scheme used.

We assume that the attacker's goal is to recover the victim's quantum programs' results after the victim has completed their computation and read out their qubits. We assume that the attacker can execute the quantum programs following the victim's quantum programs enough times, allowing the attacker to measure some statistical results. In particular, we assume that the provider of the quantum computer has strong logical isolation so that the victim's outputs cannot be directly accessed by the attacker. Thus the attacker's goal is to try to use leakage due to the hardware defects to gain the information of victim's results, which we seek to prevent.

4 One-Time Pad Implementation

The core principle underlying OTP is the random application of selected quantum gates to qubits to effectively mask the states before attackers can try to learn them. We choose Pauli-X and Pauli-Z, or the controlled version of these gates. We refer to these gates as OTP gates, though gates for OTP are not limited to these two. The OTP gates are applied in a randomized fashion discussed later. This section focuses specifically on implementation details of the OTP gates; we utilize IBM Quantum for experimental testing due to its advanced features and widespread adoption.

4.1 OTP Variants

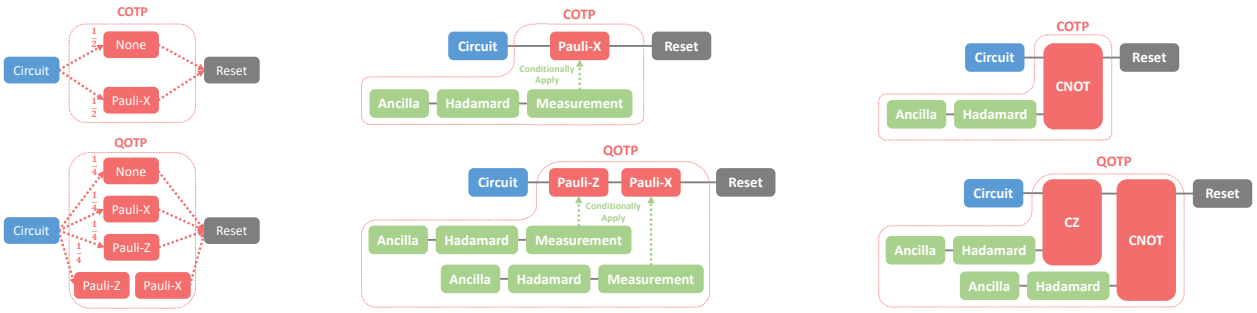
In assessing the implementations of the one-time pads for quantum computing, we consider several key factors: randomness, time and space complexity, system requirements, optimization strategies, and potential error sources, as summarized in Table 1.

Random Execution OTP (RE-OTP): This approach diverges from the standard practice of repeatedly running an identical quantum circuit. Instead, it involves the sampling of a unique circuit for each execution selected from a predefined set of circuits according to a specific random distribution. For RE-OTP implementation, each circuit shot is the original circuit appended with random OTP gates. This randomness can be generated through an external program or a random number generator integrated with the quantum computer. Prior to each shot, a random number determines the application of OTP gates. We assume attackers have no access to the randomness source and cannot influence it.

Dynamic Circuit OTP (DC-OTP): Dynamic circuits refer to executing operations conditioned on classical computations or mid-circuit measurement outcomes. Rather than running a static circuit, the gates in dynamic circuits can be dynamically altered in real-time.

Table 1: Comparison between three implementations of OTP.

Implementation	Randomness	Complexity		System Requirement	Optimization	Error Source
		Time	Space			
Random Execution	Pseudo / True (Depend on the random number generator)	Random number geneneration + Control system + Pauli Gates	Storagage for random numbers + Number of shots	Random execution of circuits in a job	Precompute Random numbers	Bias in random numbers + Pauli gate errors
Dynamic Circuit	Pseudo / True (Depend on how gate is conditioned)	Hadamard gate / Preload data + Measurement + Conditional Pauli Gates	Ancilla qubits / Storage for preload data + Number of shots	Dynamic circuit	Qubit reuse, Error-aware gate	(Bias in random numbers +) Gate errors
Control Gate	True	Hadamard gate + Control Gate	Ancilla qubits	Qubit connection	Qubit reuse, Error-aware gate	Gate errors



(a) Random Execution (RE-OTP) **(b) Dynamic Circuit (DC-OTP)** **(c) Control Gate (CG-OTP)**

Figure 1: Schematic of three implementations for the classical one-time pad (COTP) and the quantum one-time pad (QOTP). Blue blocks are victim circuits. Green blocks are the circuits to generate the randomness. Red blocks are OTP gates. Grey blocks are reset operations. Delay padding can be added in OTP gates to prevent timing attacks.

This flexibility is particularly advantageous for randomly applying OTP gates. To exemplify, for applying COTP to a single qubit, one can use a Hadamard gate followed by a dynamic Pauli-X conditioned on the measurement result. The Hadamard gate creates a superposition state, yielding 50% probability for the qubit to be measured as either $|0\rangle$ or $|1\rangle$. This state serves as the control for Pauli-X, thereby enabling the application of Pauli-X with 50% probability. Considering the noise and errors of Hadamard and measurement, it can still mitigate enough leakage compared with no OTP schemes. For QOTP, the process is similar to one additional control for Pauli-Z. Currently, the application of dynamic circuit gates relies on mid-circuit measurement results, which might necessitate extra qubits for controlling DC-OTP gates. Future advancements may enable implementation using preloaded data or real-time random number generation for even greater cost and efficiency.

Control Gate OTP (CG-OTP): Randomness can also be obtained with multi-qubit gates. Rather than applying dynamic OTP gates, two-qubit control gates are applied. For COTP, CNOT is used to control Pauli-X while for QOTP, CZ gate is further used for the correspondence of Pauli-Z.

4.2 Implementation Feasibility

The proposed implementations of OTP, while feasible for user execution on current quantum computers, present an opportunity for cloud quantum providers and quantum computer systems to integrate these schemes natively. Such integration would optimize the workflow and simplify the user experience by abstracting the execution process. A key advantage of OTP is its independence from the specific circuits, allowing for its integration as a native feature in quantum computing systems.

For RE-OTP, native support could be relatively straightforward by modifying quantum hardware or systems to enable the automatic application of OTP schemes as part of the reset protocols. The native support for DC-OTP and CG-OTP is more complex, potentially requiring the use of ancilla qubits. Still, ancilla qubits are a critical component in quantum error correction schemes, and their utility could extend to facilitating OTP. In this context, OTP gates would be integrated into the reset mechanisms by reusing ancilla qubits.

4.3 Comparison Between Implementations

In assessing the implementations, we consider five key factors as summarized in Table 1.

Randomness: Randomness for OTP implementations directly comes from the random number generation, which is typically classified as pseudo or truly random. The source of randomness in RE-OTP can be either pseudo-random, typically used in software programs, or truly random, which can be generated by additional hardware. This randomness is crucial for sampling distinct circuits for each execution shot. In DC-OTP, randomness is derived from the method used to control the OTP gates. If the control is based on the measurement results of ancilla qubits, it results in true randomness due to the probabilistic nature of quantum mechanics. However, if preloaded or computed data is used, the randomness is akin to RE-OTP, depending on the randomness of the data. The CG-OTP implementation inherently provides true randomness, as it directly exploits quantum mechanical principles.

Time Complexity: The time complexity for all implementations is constant since only one or two random numbers are needed for each qubit, and can be well paralleled. In RE-OTP, the overhead involves random number generation, the process of the quantum computer control systems, and the OTP gates themselves. For DC-OTP, additional considerations include the time for the Hadamard gate or loading pre-specified data, mid-circuit measurement, and conditional OTP gates. CG-OTP mainly involves the time overhead of the Hadamard gate and the control gates. As the time of these implementations is typically much smaller than user circuits, and they do not require the information from user circuits, they can be easily paralleled with the execution of user circuits, in which case the additional overhead can then be reduced to only OTP gates for RE-OTP and DC-OTP, or control gates for CG-OTP. Nevertheless, to prevent timing attacks [25], the overhead should be made the same regardless of what OTP gates will be executed, i.e., the overhead will be the longest one, and delay should be padded in other cases.

Space Complexity: The space complexity depends on three aspects: (1) randomness: whether it is from random numbers or qubits; (2) parallelism: whether OTP gates are performed in parallel; (3) policy: whether the same OTP gates will be applied to all qubits each qubit is protected separately. For instance, if the same OTP gates are applied to all qubits, then RE-OTP has the space complexity encompassing any additional instructions for OTP gates and the storage for random numbers that is linear to the number of shots. But if each qubit is protected separately, then the number of random numbers must also be multiplied by a factor linear to the number of qubits. In most other cases, the space complexity scales linearly with the number of qubits, especially for DC-OTP and CG-OTP which need ancilla qubits.

System Requirement: System requirements vary across these implementations. RE-OTP necessitates a system capable of sampling each shot randomly. This is not supported yet for any platform. Nevertheless, this can be simulated on contemporary quantum computers by executing a series of circuits appended with random OTP gates, with one shot for each circuit. DC-OTP depends on platforms that support this feature, which is currently only supported on platforms like IBM Quantum. CG-OTP can be readily implemented since the circuits are normal quantum circuits. However, it requires well-connected qubits for efficiency without switch gates.

Optimization: For RE-OTP, the random number generation can be precomputed or paralleled with the circuit execution. For the other two, since the qubit number is important in NISQ, linear scaling of ancilla qubits with the qubit number for DC-OTP and CG-OTP is not practical. Optimizations include techniques such as qubit reuse [13]. However, it might impact the constant time due to the break of parallelism.

Error Source: The errors mainly result from random number generation and OTP gates. In addition, because RE-OTP is independent of the circuit execution, it does not affect the circuits. However, the other two implementations may affect circuits at the end through error channels like crosstalk.

5 Noise and Error Analysis

As mentioned previously, the noise and errors come from bias in random numbers and noise in OTP gates.

5.1 Bias in Random numbers

A general single-qubit quantum state can be represented as:

$$\rho(\vec{r}) = \frac{1}{2}(I + \vec{r}\cdot\vec{\sigma}) = \frac{1}{2}\begin{pmatrix} 1 + r\cos\theta & re^{-i\phi}\sin\theta \\ re^{i\phi}\sin\theta & 1 - r\cos\theta \end{pmatrix} \tag{1}$$

where I is the identity matrix, $\vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ is the vector of three Pauli matrices, and $\vec{r} = r(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$ is the vector in Bloch sphere to represent the states.

For COTP, suppose the bias in random numbers leads to Pauli-X being applied with probability $\frac{1}{2}(1-\Delta_X)$. The state after COTP is:

$$\begin{aligned} \rho_{COTP}(\vec{r}) &= \frac{1}{2}(1+\Delta_X)\rho(\vec{r}) + \frac{1}{2}(1-\Delta_X)X\rho(\vec{r})X^\dagger \\ &= \frac{1}{2}\begin{pmatrix} 1+\Delta_X r\cos\theta & r\sin\theta(\cos\phi - i\Delta_X\sin\phi) \\ r\sin\theta(\cos\phi + i\Delta_X\sin\phi) & 1-\Delta_X r\cos\theta \end{pmatrix} \end{aligned} \tag{2}$$

Similarly, for QOTP, suppose the bias leads to Pauli-X being applied with probability $\frac{1}{2}(1-\Delta_X)$ and Pauli-Z being applied with probability $\frac{1}{2}(1-\Delta_Z)$. The state after QOTP is:

$$\begin{aligned} &\rho_{QOTP}(\vec{r}) = \frac{1}{4}(1+\Delta_Z)(1+\Delta_X)\rho(\vec{r}) + \frac{1}{4}(1+\Delta_Z)(1-\Delta_X)X\rho(\vec{r})X^\dagger + \\ &\frac{1}{4}(1-\Delta_Z)(1+\Delta_X)Z\rho(\vec{r})Z^\dagger + \frac{1}{4}(1-\Delta_Z)(1-\Delta_X)XZ\rho(\vec{r})Z^\dagger X^\dagger \\ &= \frac{1}{2}\begin{pmatrix} 1+\Delta_X r\cos\theta & \Delta_Z r\sin\theta(\cos\phi - i\Delta_X\sin\phi) \\ \Delta_Z r\sin\theta(\cos\phi + i\Delta_X\sin\phi) & 1-\Delta_X r\cos\theta \end{pmatrix} \end{aligned} \tag{3}$$

For the measurement along the Z axis, the probability of measuring -1 for both COTP and QOTP is $\frac{1}{2}(1-\Delta_X r\cos\theta)$, which only depends on the bias in random numbers controlling Pauli-X. When $\Delta_X \neq 0$, the probability is dependent on the victim state, and thus attackers can potentially measure such dependence.

For the measurement along the X axis, the probability of measuring -1 for COTP is $\frac{1}{2}(1 - r\sin\theta\cos\phi)$, which does not depend on Δ_X. For QOTP, the probability is $\frac{1}{2}(1-\Delta_Z r\sin\theta\cos\phi)$. When $\Delta_Z \neq 0$, the dependence exists and can be potentially measured.

The only dependence on Δ_X or Δ_Z is not surprising and is due to the choice of the measurement axis. For example, for measuring along the Z axis, the probability is related to the projection of the Bloch state into the Z axis. This projection is the same for both the original state and the state rotated by Pauli-Z, so Δ_Z does not influence the results when measuring along the Z axis. Similarly, Δ_X does not impact the measurement along the X axis. However, for a general measurement axis, both of them will affect the results.

Importantly, the bias does not depend on our scheme, but on the source of randomness. Research on random number generators is an active area [11, 12], and we assume a non-biased random number generator will be used. There are many tests, such as the NIST Test Suites [5], to determine how good a random number generator is, if it has any temporal correlation [21], etc. For RE-OTP, the bias can be reduced by directly using a good random number generator. For the case of DC-OTP and CG-OTP, which use ancilla qubits, one optimization is to use a rotational gate instead of a Hadamard gate as discussed in [3, 16] to change rotational angles a little to offset their bias.

5.2 Noise and Errors in Gates

The errors can also come from Pauli-X and Pauli-Z for RE-OTP and DC-OTP, and the control gates for CG-OTP. There can be many error channels, and it is beyond the scope of this paper to analyze

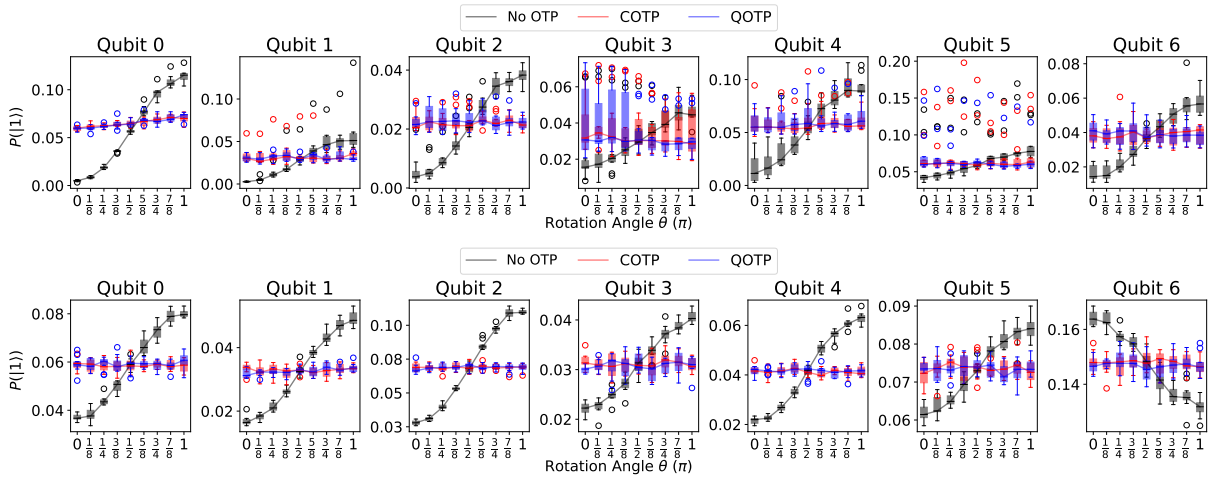


Figure 2: Probability of measuring $|1\rangle$ on all 7 qubits of the real quantum computer `ibmq_jakarta` with no one-time pad (No OTP), classical one-time pad (COTP), and quantum one-time pad (QOTP) before reset instruction. (Top Row) Results from the real quantum machine `ibm_jakarta`. (Bottom Row) Results from the simulator with the noise model imported from `ibm_jakarta`. The flatter the lines means the better the mitigation on state leakage, and thus harder for attackers to retrieve information related to previous executions.

them. For simplicity, if only the bit-flip error is considered in Pauli-X, such an error is the same as adding bias in random numbers since it is equivalent to not applying the gate. For Pauli-Z, it can be implemented as the virtual RZ gate [17], which does not introduce errors. If the virtual RZ gate is not supported on a target quantum computer, similar issues to Pauli-X gate will exist. Multi-qubit gates are typically much more erroneous than single-qubit gates, leading to larger errors, which need to be considered.

6 Evaluation

This section presents the evaluation of the proposed OTP variants for securing quantum computer reset operations. We emulate the victim and attacker by running both of them in a single quantum program. The circuits are tested on both quantum computers and simulators. Each shot of the evaluation quantum circuit is divided into three parts:

Victim Circuit: This is the part to emulate the victim circuit. To test the results of different qubit states, one rotational X gate with different rotational angles θ and one rotational Z gate with rotational angle $\frac{\pi}{2}$ are applied to evolve qubit states to: $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)|1\rangle$, where $\theta \in \{0, \frac{1}{8}\pi, \frac{1}{4}\pi, \ldots, \pi\}$, nine values from 0 to π. Qubits are then followed by measurements to emulate the process of measuring and obtaining results.

Reset Mechanism: This is the part to emulate the reset mechanism between shots. In the beginning, three schemes: no OTP, COTP, or QOTP are performed, and then they are followed by the reset instruction. On the real quantum machine, the reset instruction is the supported reset instruction. On the simulator, the reset instruction is mimicked by the mid-circuit measurement and conditional Pauli-X, because the reset instruction has a very simple noise model that cannot show the same behavior on the cloud. Note that the reset mechanism is not restricted to the reset instruction as discussed in [27]. The default thermalization can also be tested.

Attacker Circuit: This is the part to emulate the attacker circuit. There is only one measurement in the circuit.

6.1 State Leakage in Different Schemes

This section provides the evaluation results of different OTP schemes, i.e., no OTP, COTP, and QOTP, on the quantum computer and simulator. The experiments are performed on all seven qubits on `ibmq_jakarta` and the Qiskit Aer simulator with `ibmq_jakarta`'s noise model. The tested variant is the RE-OTP. 10 experiments are done for each θ and scheme, with each experiment 10,000 shots.

Figure 2 shows the results of the probability of measuring $|1\rangle$ with the three OTP schemes. Without any OTP, the normal reset instruction shows a dependence on the victim states. Such statistical patterns due to leakage can be measured by attackers to retrieve the victims' qubit states, as discussed in [19, 27].

With OTP, the statistical results can be flattened, and thus it is hard or impossible for attackers to employ the same attack to acquire leaked information regarding victim states. However, the results of COTP and QOTP do not show a large difference. This is because, after the victim's measurement, the states are collapsed into either $|0\rangle$ or $|1\rangle$, in which case both COTP and QOTP are secure. Though in quantum computers, the states after the measurement may be more complex and may include superposition and mixed constituents due to noise and errors, the amount is small enough so that the difference between COTP and QOTP cannot be detected. Notice that the calibration time of the noise model is different from when the experiments on the real quantum computer were performed, so the results from the real quantum computer and the simulator cannot be directly compared with each other.

6.2 State Leakage in Different Implementations

This section provides the evaluation results of different implementations, i.e., RE-OTP, DC-OTP, and CG-OTP, on the real quantum computer and simulator. The experiments are performed on `ibm_nairobi` and the Qiskit Aer simulator with the noise model imported from `ibm_nairobi` 8 experiments are done for each θ and implementation, with each experiment 1,000 shots. Due to the limitation of qubit connection in quantum simulation, qubit 1 is

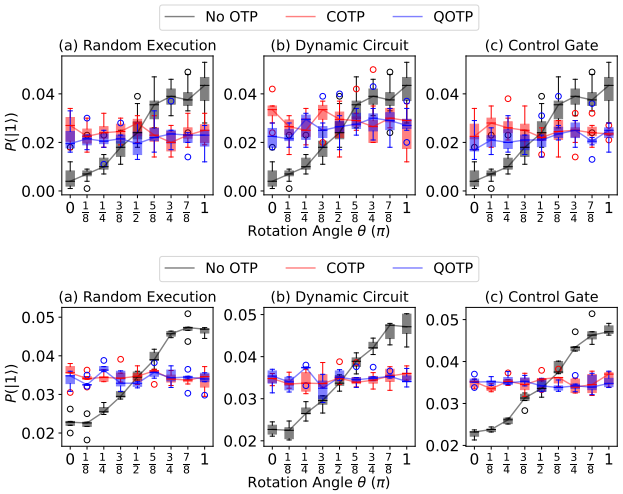


Figure 3: Probability of measuring $|1\rangle$ with different implementations of OTP. (Top Row) Results from the real quantum machine `ibm_nairobi`. (Bottom Row) Results from the simulator with the noise model imported from `ibm_nairobi`. The flatter the lines means the better the mitigation on state leakage, and thus harder for attackers to retrieve information related to previous executions.

chosen due to its connection to more than 1 qubit so that QOTP can be implemented without the swap gate.

Figure 3 shows the results of the probability of measuring $|1\rangle$ with three implementations. The results show that there is no significant difference in them, though RE-OTP is less noisy since it does not include the random number generation process in quantum computers. The implementation thus may rely more on the tradeoff between advantages and disadvantages discussed in Section 4.

7 Conclusion

Our research addresses the issue of information leakage in quantum computers, by proposing different OTP variants to secure reset operations. Our approach involves the random application of rotational gates before reset operations. These solutions are low-overhead and easy to adapt to various quantum computing systems.

Acknowledgments

The authors would like to thank IBM and Yale University for quantum computer access, and the support by NSF grant 2332406.

References

[1] 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)*. Association for Computing Machinery, 25–30.

[2] Abdullah Ash-Saki, Mahabubul Alam, and Swaroop Ghosh. 2020. Experimental Characterization, Modeling, and Analysis of Crosstalk in a Quantum Computer. *IEEE Transactions on Quantum Engineering* 1 (2020), 1–6.

[3] Abdullah Ash-Saki, Mahabubul Alam, and Swaroop Ghosh. 2020. Improving Reliability of Quantum True Random Number Generator using Machine Learning. In *International Symposium on Quality Electronic Design (ISQED)*. 273–279.

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

[5] Lawrence E Bassham III, Andrew L Rukhin, Juan Soto, James R Nechvatal, Miles E Smid, Elaine B Barker, Stefan D Leigh, Mark Levenson, Mark Vangel, David L Banks, et al. 2010. *Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications.* National Institute of Standards & Technology.

[6] Brennan Bell and Andreas Trügler. 2022. Reconstructing quantum circuits through side-channel information on cloud-based superconducting quantum computers. In *Conference on Quantum Computing and Engineering (QCE)*. 259–264.

[7] Steven M Bellovin. 2011. Frank Miller: Inventor of the one-time pad. *Cryptologia* 35, 3 (2011), 203–222.

[8] P. Oscar Boykin and Vwani Roychowdhury. 2003. Optimal encryption of quantum bits. *Phys. Rev. A* 67 (April 2003), 042317. Issue 4.

[9] Sanjay Deshpande, Chuanqi Xu, Theodoros Trochatos, Yongshan Ding, and Jakub Szefer. 2022. Towards an Antivirus for Quantum Computers. In *International Symposium on Hardware Oriented Security and Trust (HOST)*. 37–40.

[10] Sanjay Deshpande, Chuanqi Xu, Theodoros Trochatos, Hanrui Wang, Ferhat Erata, Song Han, Yongshan Ding, and Jakub Szefer. 2023. Design of Quantum Computer Antivirus. In *International Symposium on Hardware Oriented Security and Trust (HOST) (HOST)*. 260–270.

[11] Z. Gutterman, B. Pinkas, and T. Reinman. 2006. Analysis of the Linux random number generator. In *2006 IEEE Symposium on Security and Privacy (S&P'06)*. 15 pp.–385. https://doi.org/10.1109/SP.2006.5

[12] Miguel Herrero-Collantes and Juan Carlos Garcia-Escartin. 2017. Quantum random number generators. *Reviews of Modern Physics* 89, 1 (2017), 015004.

[13] Fei Hua, Yuwei Jin, Yanhao Chen, Suhas Vittal, Kevin Krsulich, Lev S. Bishop, John Lapeyre, Ali Javadi-Abhari, and Eddy Z. Zhang. 2023. CaQR: A Compiler-Assisted Approach for Qubit Reuse through Dynamic Circuit. In *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. Association for Computing Machinery, 59–71.

[14] IBM Quantum. 2023. The hardware and software for the era of quantum utility is here. https://research.ibm.com/blog/quantum-roadmap-2033.

[15] 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.

[16] Yuanhao Li, Yangyang Fei, Weilong Wang, Xiangdong Meng, Hong Wang, Qianheng Duan, and Zhi Ma. 2021. Quantum random number generator using a cloud superconducting quantum computer based on source-independent protocol. *Scientific Reports* 11, 1 (2021), 23873.

[17] David C. McKay, Christopher J. Wood, Sarah Sheldon, Jerry M. Chow, and Jay M. Gambetta. 2017. Efficient Z gates for quantum computing. *Phys. Rev. A* 96 (Aug 2017), 022330. Issue 2.

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

[19] Allen Mi, Shuwen Deng, and Jakub Szefer. 2022. Securing Reset Operations in NISQ Quantum Computers. In *Conference on Computer and Communications Security (CCS)*. Association for Computing Machinery, 2279–2293.

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

[21] Yutaka Shikano, Kentaro Tamura, and Rudy Raymond. 2020. Detecting Temporal Correlation via Quantum Random Number Generation. *Electronic Proceedings in Theoretical Computer Science* 315 (April 2020), 18–25.

[22] Jerry Tan, Chuanqi Xu, Theodoros Trochatos, and Jakub Szefer. 2024. Extending and Defending Attacks on Reset Operations in Quantum Computers. In *2024 International Symposium on Secure and Private Execution Environment Design (SEED)*. 73–83. https://doi.org/10.1109/SEED61283.2024.00018

[23] Yizhuo Tan, Chuanqi Xu, and Jakub Szefer. 2025. *Exploration of Timing and Higher-Energy Attacks on Quantum Random Access Memory.* Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3676536.3689916

[24] Chuanqi Xu, Jessie Chen, Allen Mi, and Jakub Szefer. 2023. Securing NISQ Quantum Computer Reset Operations Against Higher Energy State Attacks. In *Conference on Computer and Communications Security* (Copenhagen, Denmark) *(CCS)*. Association for Computing Machinery.

[25] Chuanqi Xu, Ferhat Erata, and Jakub Szefer. 2023. Exploration of Power Side-Channel Vulnerabilities in Quantum Computer Controllers. In *Conference on Computer and Communications Security (CCS)*. Association for Computing Machinery.

[26] Chuanqi Xu, Ferhat Erata, and Jakub Szefer. 2024. Quantum Computer Fault Injection Attacks. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 01. 331–337. https://doi.org/10.1109/QCE60285.2024.00047

[27] Chuanqi Xu, Jamie Sikora, and Jakub Szefer. 2024. A Thorough Study of State Leakage Mitigation in Quantum Computing with One-Time Pad. In *2024 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. 55–65. https://doi.org/10.1109/HOST55342.2024.10545386

[28] Chuanqi Xu and Jakub Szefer. 2025. Security Attacks Abusing Pulse-level Quantum Circuits . In *2025 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, Los Alamitos, CA, USA, 222–239. https://doi.org/10.1109/SP61157.2025.00083