Quantum Information Processing (2024) 23:208
https://doi.org/10.1007/s11128-024-04421-x

Software-based remote memory attestation using quantum entanglement

Jesse Laeuchli[1] · **Rolando Trujillo-Rasua**[2]

Received: 26 November 2023 / Accepted: 7 May 2024

Abstract
Software-based remote memory attestation is a method for determining the state of a remote device without relying on secure hardware. In classical computing devices, the method is vulnerable to proxy and authentication attacks, because an infected device has no means of preventing the leak of its cryptographic secrets. In this paper, we demonstrate how these attacks can be mitigated by making use of quantum effects, while remaining within the class of software-based methods. In particular, we make use of entanglement and the inability of an attacker to clone qubits. Our proposed protocol is lightweight and can be implemented by near-term Quantum Computing techniques. The resulting protocol has the unique feature of resisting collusion between two dishonest devices, one of which has unbounded computational resources.

Keywords Memory attestation · Entanglement · Distance bounding · Distant attacker

1 Introduction

Detecting whether an embedded device is infected with malware is essential to maintaining a secure critical infrastructure, as malicious code is used, for example, to compromise secrets and escalate privileges remotely. A popular approach to this problem is remote attestation (RA): a trust establishment mechanism whereby a device, called *verifier*, ensures the memory content of another device, called *prover*, is in a safe (possibly initial) state, thereby ensuring the absence of malicious code in memory. The term *remote* in this case means that the verification procedure is executed over a network, rather than by having direct physical access to the embedded device. In this setting, authentication becomes an important security requirement, allowing the

✉ Jesse Laeuchli
j.laeuchli@unsw.edu.au; jesse@laeuchli.com

Rolando Trujillo-Rasua
rolando.trujillo@urv.cat

[1] University of New South Wales Canberra, Northcott DR, Campbell, ACT 2612, Australia

[2] Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Spain

Published online: 27 May 2024

verifier to ensure that the intended prover is attested rather than a third device, possibly placed by the attacker.

Because the goal of remote attestation is to verify whether the prover is in an initial safe state, none or few security assumptions can be made about the cryptographic keys stored in the device prior the execution of the protocol. Hence, RA protocols usually rely on non-cryptographic means, such as secure hardware, timed channels, visual inspection, signal jamming, device fingerprinting, etc. Amongst these techniques, only secure hardware offers cryptographic proofs of authentication by protecting secrets and preventing malicious code from interfering with the attestation routine [9, 18].

Relying on secure hardware to achieve authentication poses two problems, though. First, it increases the manufacturing cost. Second, it makes a vulnerability in the hardware itself costly to fix and impossible to patch at a large scale. While software vulnerabilities can be patched remotely, hardware vulnerabilities likely require a call out of vulnerable devices. This makes the problem of *designing a remote attestation protocol that achieves authentication without relying on secure hardware worth addressing.*

Prior research results [2, 11, 28] have shown the problem above to be unsolvable when the prover can receive help from an attacker. If the embedded device has been compromised by an attacker and the device has no secure hardware, then all its memory content, including all cryptographic keys, can be leaked. The attacker can thus dump the memory content of the embedded device into a more powerful device that would take over the communication with the verifier, or instruct the prover to relay its communication with the verifier to a more powerful device. In the literature, the latter attack is called a proxy attack [27], while the former attack is usually neglected. While defeating these attacks is impossible under the standard computational model, we show in this article that achieving secure remote attestation without secure hardware is possible using quantum methods.

Contributions We put forward in this work the first remote attestation protocol that satisfies authentication and resists collusion with remote conspirators without relying on secure hardware, relying instead on information and quantum theory. Our protocol requires the embedded device and the verifier to share some quantity of maximally entangled qubit pairs (EPRs), needing only low depth quantum circuits that have already been experimentally achieved. That is, our method requires relatively simple quantum capabilities on both the verifier and the embedded device.

Rather than using secret keys, our protocol uses entangled qubit pairs for authentication. However, different to current quantum authentication mechanisms, our approach restricts the prover in its capability to transmit qubits. Such a restriction is necessary, as otherwise the corrupt prover could transmit them to a conspirator. To deter proxy attacks, our approach takes advantage of the classical bits necessary for quantum teleportation to provide the protocol with a distance-bounding mechanism.

We acknowledge that our proposal does require a prover to hold entangled qubits. As far as secure hardware is concerned, we make no assumption. We assume the attacker has full control of the prover and it is only limited by the laws of physics that restrict the transmission of information, including Quantum information.

2 Related work

Designing secure remote attestation protocols is notoriously challenging because the prover is assumed to be corrupt. This assumption would break, for example, the standard authentication properties used to evaluate the security of internet protocols, such as TLS (Transport Layer Security). The earliest works to tackle this challenge focused on limiting the adversary's corruption capability, notably on protecting cryptographic secrets. This requires secure hardware on the prover's side, leading to the terminology *hardware-based attestation* when referring to these approaches.

Francillon et al. established that, in addition to protecting secret keys, secure hardware is needed to prevent malware from modifying or interrupting the attestation routine [11]; otherwise, the malware may give itself time to move to safe parts of the memory during attestation or to leak information about secret keys. Many works in the literature aim at achieving the above security features for various computer architectures and with the use of minimal hardware [9]. We refer the reader to [18] for a comprehensive survey.

Our focus is on *software-based attestation*: an approach towards remote attestation that assumes no secure hardware on the remote device. This makes software-based approaches more cost-effective, while also allowing devices to patch vulnerabilities without the need to physically modify the device with additional hardware.

SWATT [26] is one of the earliest software-based RA protocols, where the verifier sends a random challenge to the prover, which is used as a seed for a pseudo-random function to obtain a sequence of random memory indexes. The challenge posed to the prover is that of calculating, as quickly as possible, a checksum of its memory in the order specified by the pseudo-random function. SWATT claims that, if the checksum computation cannot be optimised and every bit of memory is hit (with high probability) by the pseudo-random function, then the verifier can attest the prover memory by evaluating the checksum value and the time taken to calculate it. SWATT's construction of a time-optimal checksum function consists of a computation loop where the value of a random memory index is merged with the checksum of the previous round.

Because of the time constraint, software-based remote attestation has traditionally relied on non-standard checksum calculations [17, 26, 27, 29], opening themselves to attacks [6, 20]. Armknecht et al. argued that the insecurity of software-based approaches is compounded by the lack of a formal treatment of their security analysis [2]. Hence, they introduced a security framework to analyse software-based remote attestation for low-cost embedded devices. When looking at the adversary capabilities against remote attestation, Armknecht et al. deemed it necessary to restrict the adversary to use the hardware characteristics of the prover during the execution of the challenge–response protocol. That is to say, during the protocol execution, the prover cannot receive external help; otherwise, the protocol is trivially broken by outsourcing the attestation task to a more powerful device [27].

To the best of our knowledge, no software-based attestation protocol exists that satisfies authentication or resists proxy attacks. That is an important limitation, which forces the verifier to run the protocol in a clean environment aided by the use of out-of-the-band channels, such as visual inspection. A recent work [28] proposes the use of a

distance-bounding channel, rather than visual inspection, to detect collusion with far-away conspirators during the protocol execution. However, they left the authentication problem open, which we address in this work.

From a methodological point of view, our approach is similar to a recent wave of protocols based on Physical Unclonable Functions (PUFs) [10, 16, 25]. They assume that provers are provided with a PUF implementation, allowing the verifier to precompute secret challenge–response pairs. Their key security assumption is that the attacker cannot tamper with the PUF function nor learn the challenge–response pairs the prover shares with the verifier. We note, however, that this is still a contentious assumption [23]. Particularly dangerous are modelling attacks on Strong PUFs, allowing an attacker to mimic the PUF behaviour by using machine learning. Last but not least, PUF functions do require additional hardware to protect them from effective adversarial manipulation [24], such as re-use attacks, making PUF-based remote attestation not strictly software based.

In parallel to this work, Khan et al.'s [15] published a remote memory attestation procedure using quantum-based PUFs. Like the PUF-based approaches just reviewed, the security of Khan et al.'s protocol rests on engineering guarantees and supply chain security. They, in addition, require the presence of a secure quantum channel for transmission between the prover and verifier. In contrast, we provide a software-based approach that relies solely on information and quantum theory. Our method does not require a secure quantum channel, nor secure hardware, to function. In fact, to eliminate the possibility of an attacker communicating directly with the verifier, we restrict the prover to transmit only classical information.

There have been some prior work on securing memory using Quantum Computing, but this has been for local computers and requires a full range of quantum circuits [7]. In contrast, our work considers remote embedded devices and requires only rotation and measurements.

Also closely related to the problem, we are trying to solve are the series of works on Quantum Authentication based on entanglement and teleportation [8]. They, however, differ from our approach in that they start with a shared classical secret, and use quantum mechanics to *amplify* this key. In contrast, having our device start with a shared classical secret key is not desirable, since a compromised device can easily leak its secrets.

Alternatively, [1, 19] start with shared entangled pairs, which is similar to our approach. However, their method has both parties sending and receiving qubits, which also is undesirable for our application, since if the embedded device has the ability to transmit qubits, the entangled pairs could be transmitted to an attacker for communication.

In general, while these Quantum Authentication methods consider a closely related problem, they do not consider it from the point of view where one party may be compromised, or working against the authenticating authority. In contrast, we consider both the case where one party has been compromised during the initial stage, and the case where one party is subject to ongoing compromise and continues to actively work in conjunction with the attacker.

3 Preliminaries

Here we introduce the notation and tools we intend to make use of later in our paper. We begin by reviewing the basics of quantum states and quantum teleportation. For a complete introduction to these ideas, the reader is recommended to consult [5].

3.1 Quantum Notation

We define a single qubit in superposition using the standard Bra–ket notation as

$$|\psi\rangle = \alpha_1|0\rangle + \alpha_2|1\rangle \tag{1}$$

where $\alpha_1, \alpha_2 \in \mathbb{C}$ and $|\alpha_1|^2 + |\alpha_2|^2 = 1$. We define multi-qubit systems in the same way,

$$|\psi\rangle = \sum_{i=0}^{n} \alpha_i|i\rangle \tag{2}$$

where again $\alpha_i \in \mathbb{C}$ and $\sum_{i=0}^{n} |\alpha_i|^2 = 1$. Our probability of obtaining any basis state $|i\rangle$ on measurement is $|\alpha_i|^2$, according to the Born rule. We note that here we are working in the computational basis $|i\rangle$, but that other basis states are possible, a fact which we will make use of later.

We call any state $|\psi\rangle_c$, which can be written as the tensor product of other states $|\psi\rangle_c = |\psi\rangle_a \otimes |\psi\rangle_b$ separable. States which are not separable are entangled. As an example, $|\psi\rangle_c = |01\rangle = |0\rangle \otimes |1\rangle$ is separable, whereas $|\psi\rangle_c = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ is entangled. Thus, if we have the state $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$, and we provide one half of the entangled pair to one party and the other half to another, if either party measures $|0\rangle$ from their half of the pair, the other party will also measure $|0\rangle$, and similarly for $|1\rangle$. This property of entanglement is the basis for the power of quantum computing in general. In particular, this is the property that enables Quantum Teleportation, which is the basis for our scheme.

3.2 Quantum Teleportation

We review Quantum Teleportation here, following the approach of [5]. Quantum Teleportation is a method for teleporting a qubit from one party to another. The state to be teleported is transmitted instantly over any distance, but cannot be extracted without two classics bits, which can of course be transmitted no faster than the speed of light.

The basis for this approach is that the two parties Alice and Bob share a maximally entangled pair of qubits. There are four such pairs, referred to as the Bell States.

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}|0\rangle_a|0\rangle_b + \frac{1}{\sqrt{2}}|1\rangle_a|1\rangle_b$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}|0\rangle_a|0\rangle_b - \frac{1}{\sqrt{2}}|1\rangle_a|1\rangle_b$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}|0\rangle_a|1\rangle_b + \frac{1}{\sqrt{2}}|1\rangle_a|0\rangle_b$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}|0\rangle_a|1\rangle_b - \frac{1}{\sqrt{2}}|1\rangle_a|0\rangle_b$$

In the equations above, the subscripts a and b denote qubits hold by Alice and Bob, respectively. It does not matter which pair Alice and Bob share, so for convenience we will assume that they shared $|\Phi^+\rangle$. Define $|\psi\rangle_c = \alpha_1|0\rangle_c + \alpha_2|1\rangle_c$, as the state that Alice wishes to transmit to Bob. The state of the system is thus

$$|\psi\rangle_c \otimes |\Phi^+\rangle =$$
$$(\alpha_1|0\rangle_c + \alpha_2|1\rangle_c) \otimes (\frac{1}{\sqrt{2}}|0\rangle_a|0\rangle_b + \frac{1}{\sqrt{2}}|1\rangle_a|1\rangle_b)$$

We can rewrite Alice's qubits using the following identities.

$$|00\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle + |\Phi^{-1}\rangle)$$
$$|01\rangle = \frac{1}{\sqrt{2}}(|\Psi^+\rangle + |\Psi^{-1}\rangle)$$
$$|10\rangle = \frac{1}{\sqrt{2}}(|\Psi^+\rangle - |\Psi^{-1}\rangle)$$
$$|11\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle - |\Phi^{-1}\rangle)$$

The total state of the system can be rewritten as below. Note that this does not change the state of the system, we have merely rewritten it in a new basis.

$$|\psi\rangle_c \otimes |\Psi^+\rangle = \frac{1}{2}(|\Phi^+\rangle_{ca} \otimes (\alpha_1|0\rangle_b + \alpha_2|1\rangle_b) +$$
$$|\Phi^-\rangle_{ca} \otimes (\alpha_1|0\rangle_b - \alpha_2|1\rangle_b) +$$
$$|\Psi^+\rangle_{ca} \otimes (\alpha_1|1\rangle_b + \alpha_2|0\rangle_b) +$$
$$|\Psi^-\rangle_{ca} \otimes (\alpha_1|1\rangle_b - \alpha_2|0\rangle_b) +$$

Alice now measures her qubits c and a in the Bell basis. With equal probability, the system is now in one of the states given below.

$$|\Phi^+\rangle_{ca} \otimes (\alpha_1|0\rangle_b + \alpha_2|1\rangle_b)$$
$$|\Phi^-\rangle_{ca} \otimes (\alpha_1|0\rangle_b - \alpha_2|1\rangle_b)$$
$$|\Psi^+\rangle_{ca} \otimes (\alpha_1|1\rangle_b + \alpha_2|0\rangle_b)$$
$$|\Psi^-\rangle_{ca} \otimes (\alpha_1|1\rangle_b - \alpha_2|0\rangle_b)$$

Alice then informs Bob which of the four states the total system is in, based on the result of her measurement by transmitting 2 classical bits. If the system is in the first state, Bob does nothing; otherwise, he performs a rotation on his qubit to return it to the form $\alpha_1|0\rangle_b + \alpha_2|1\rangle_b$. Thus, as long as Alice can transmit two classical bits, she can teleport an arbitrary quantum state to Bob, as long as they share two maximally entangled pairs.

We note since we intend to make extensive use of it, that since the values α_1, α_2 are complex, a qubit lies on a unit sphere known as the Bloch Sphere. Considering α_1, α_2, as a vector on this sphere with polar angle θ and azimuthal angle ϕ, we can represent any qubit as $cos(\frac{\theta}{2})|1\rangle + sin(\frac{\theta}{2})e^{i\phi}|0\rangle$. Our algorithm mostly makes use of the unit circle on this sphere where $0 \leq \theta \leq \pi$, and $\phi = 0$ or $\phi = \pi$.

4 Remote attestation using quantum entanglement

In this section, we introduce two memory attestation protocols. The first protocol is a composition of a traditional software-based memory attestation protocol with a distance-bounding mechanism that relies on pre-shared entanglement rather than on secret keys. Being a composition, this protocol is relatively simple to analyse and can be instantiated with various memory attestation routines, although at the cost of high communication complexity. The second protocol embeds the distance-bounding mechanism within the generic scheme introduced by Armknecht et al. [2], making it more efficient albeit slightly less secure.

4.1 A protocol composition

This protocol assumes the existence of a time-restricted attestation function Attest in the prover with the following properties, given a time threshold Δ:

- *Correctness* Attest can be computed within Δ time.
- *Uniformity* Every output of the range should be generated with roughly the same probability, given random inputs.
- *Secure* The Attest function allows a verifier, by challenging an isolated prover (i.e. without external help) to compute Attest on input a random challenge, to attest whether the prover is in a correct state with nonzero probability. We note that many software-based memory attestation primitives [2, 13, 17, 26, 27, 29] in the literature aims at achieving this notion of security, which we formalise later together with the threat model.

Setup The setup phase of the protocol allows provers and verifiers to agree on secret cryptographic keys (possibly) used by the Attest function and on a collection of entangled qubits in superposition. This setup phase can be executed several times with the help of a trusted server, once both prover and verifier have been attested uncorrupt.

Challenge The protocol consists of several challenge–response rounds. At each round, the verifier sends a random challenge c to the prover and starts a clock to measure the round-trip time of this exchange with the prover (see Fig. 1). The role of the time

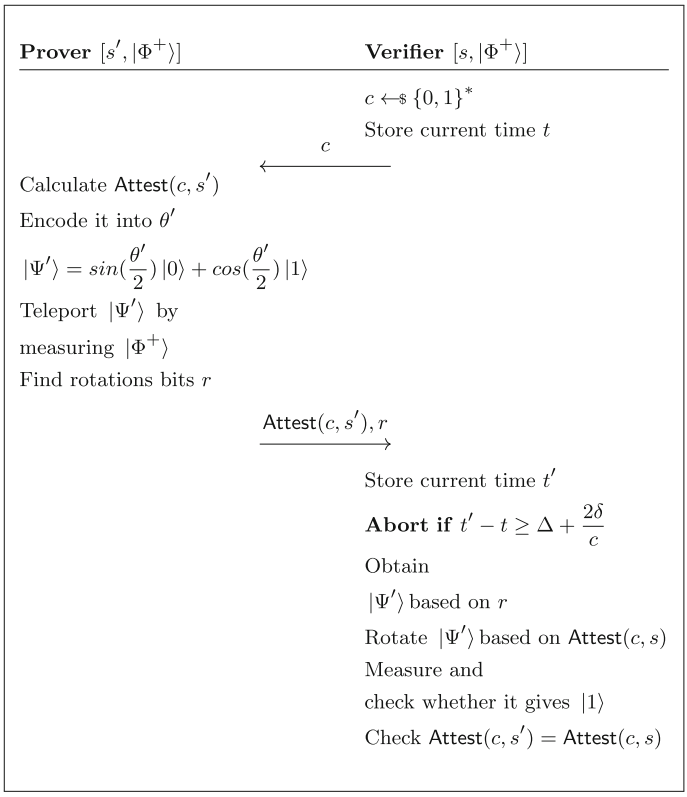


Fig. 1 A single-round remote memory attestation protocol using Quantum entanglement

measurement is counteracting proxy attacks. The role of the random challenge is preventing replay attacks on the Attest primitive.

Response Upon receiving the challenge, the prover calculates $\mathsf{Attest}(c, s')$, where s' is the prover's state. Then the prover teleports the qubit $|\Psi'\rangle = sin(\frac{\theta'}{2})|0\rangle + cos(\frac{\theta'}{2})|1\rangle$, where θ' is the angle obtained from the output of $\mathsf{Attest}(c, s')$ via normalisation into $[0, \pi]$. The teleportation is carried out by entangling $|\Psi\rangle$ with the qubit to be teleported, then measuring in the Bell basis. This measurement provides the teleportation bits. Lastly, the prover provides both the output of $\mathsf{Attest}(c, s')$ and the teleportation bits.

Verification Let s be the expected state of the prover. In the verification phase, the verifier aims to authenticate the prover, check proximity with the prover defined by a distance threshold δ, and check that $\mathsf{Attest}(c, s') = \mathsf{Attest}(c, s)$. Proximity is verified by checking whether the round-trip-time measurement is below $\frac{2\delta}{c} + \Delta$, where δ is a distance threshold and c the propagation speed of the communication channel (e.g. the speed of light for radio waves). Lastly, to authenticate the prover, the verifier rotates the result based on the rotation bits (r) provided by the prover in its response and angle θ. Let $|\Psi\rangle$ be the resulting qubit. The verifier checks that $|\Psi\rangle = |\Psi'\rangle$ by performing an

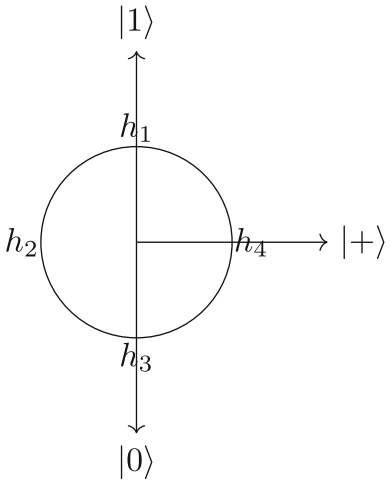
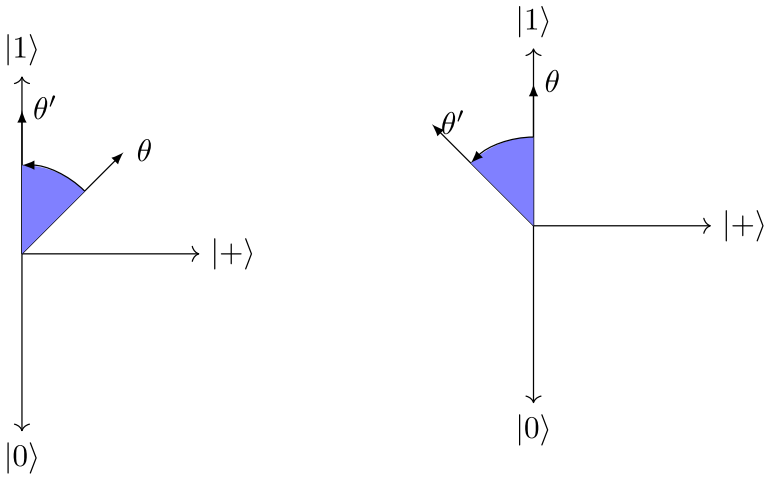


Fig. 2 Potential qubits after teleporting classical bits

inverse rotation $|\Psi\rangle$ with angle θ as defined by $\mathsf{Attest}(c, s)$, measuring, and checking that the result of the measurement is $|1\rangle$ (Fig. 2).

Example 1 Let us consider an example of the protocol running on a compromised device. Further assume that the protocol consists of 2 challenge–response rounds with random challenges c_1 and c_2. Let θ_1' and θ_2' be the angles obtained from normalising the values $\mathsf{Attest}(s_0', c_0)$ and $\mathsf{Attest}(s_0', c_1)$, respectively. Likewise, let θ_1 and θ_2 be the angles obtained from $\mathsf{Attest}(s_0, c_0)$ and $\mathsf{Attest}(s_0, c_1)$, respectively. Suppose $\theta_1' = \frac{\pi}{4}$ and $\theta_2' = \frac{\pi}{2}$, and that the angles generated by the Attest function on the correct state s are $\theta_1 = \frac{\pi}{4}$ and $\theta_2 = \frac{\pi}{4}$. This means that, in the first round, the prover teleports $sin(\frac{\theta_1}{2})|0\rangle + cos(\frac{\theta_1}{2})|1\rangle$ and sends the required classical bits based on the state it finds its qubits to be in post teleport. The verifier after completing the teleportation using the classical bits performs an additional rotation on the qubit of $\frac{\pi}{8}$. This leaves the qubit in the state of $sin(0)|0\rangle + cos(0)|1\rangle = 0|0\rangle + 1|1\rangle$. On performing a measurement, $|1\rangle$ will be measured with certainty, and the protocol continues if $\mathsf{Attest}(s_0', c_0) = \mathsf{Attest}(s_0, c_0)$ and the round-trip time is low enough.

In the second round, the prover then teleports the second qubit $sin(\frac{\theta_2}{2})|0\rangle + cos(\frac{\theta_2}{2})|1\rangle$. The verifier receives $sin(\frac{\theta_2}{2})|0\rangle + cos(\frac{\theta_2}{2})|1\rangle$. Since the correct qubit is $sin(\frac{\pi}{8})|0\rangle + cos(\frac{\pi}{8})|1\rangle$, the verifier performs an additional rotation of $\frac{\pi}{8}$, leaving the qubit in the state of $sin(\frac{\pi}{8})|0\rangle + cos(\frac{\pi}{8})|1\rangle$ (see Fig. 3 for a graphical representation). This means that with probability $cos(\frac{\pi}{8})^2 = 0.8536$, the verifier will measure a $|1\rangle$, causing the protocol to continue, and with probability $sin(\frac{\pi}{8})^2 = 0.1464$ will measure a $|0\rangle$, indicating that the memory of the prover is compromised and causing the protocol to halt. It should be noted that the verifier can also check integrity of the prover's memory by checking whether $\mathsf{Attest}(s_1', c_0) = \mathsf{Attest}(s_1, c_0)$. However, as we will show in the security analysis below, a colluder can always help the prover to send the correct value of the Attest function; hence, our focus in this example is on the teleported quantum state.



(a) Before rotation (b) After rotation

Fig. 3 Example of system in a compromised state after rotation. The qubit should be in state $|1\rangle$, but is instead in superposition between $|0\rangle$ and $|1\rangle$

It is worth noting that the naive approach of the prover sending $|0\rangle$ or $|1\rangle$, depending on the value of Attest (e.g. its parity bit), does not work. For example, suppose the prover incorrectly believes that it has to teleport $|1\rangle$, and after measurement, the system is in the state h_1 (see Fig. 2 showing the four possible states). The prover sends the classical bits denoting this to the verifier. If the attacker, however, knows that the correct state of system is the state h_3, then it can tell the verifier to rotate the qubit by π radians, causing the verifier to measure $|0\rangle$ instead. In this approach, no matter what state the system is in after the teleportation, the attacker can cause the verifier to perform an incorrect rotation of either $\pi/2$ or π, leading the verifier measuring a state orthogonal to what was intended. Our protocol does offer some protection against this attack, as proven in the security analysis given in the next section.

4.2 A modification of Armknecht et al.'s protocol with quantum teleportation

Next we introduce an extension of the generic scheme introduced by Armknecht et al. [2] that detects collusion between provers and external attackers. This extension aims to overcome the two main limitations of the protocol composition described above: the assumption that the Attest function exists and the use of many challenge–response rounds.

A description of Armknecht et al.'s protocol is given next, which can be followed by looking at Fig. 4. Our extensions are described afterwards and depicted in Fig. 4 as well.

Challenge Armknecht et al.'s protocol starts with the verifier sending two nonces n_0 and m_0 to the prover. The former is used to feed a pseudo-random function Gen, which

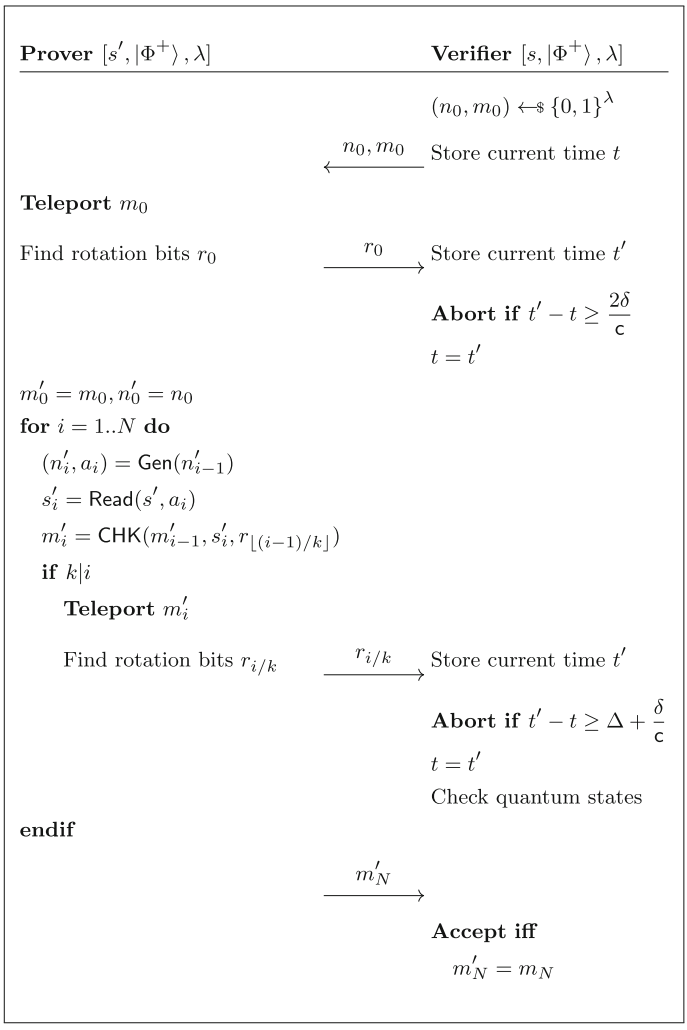


Fig. 4 A memory attestation protocol resistant to proxy attacks with low communication complexity

will create random memory addresses, the latter to calculate a checksum CHK of the prover's memory.

Response The prover prepares a response by iteratively updating n_0 and m_0. At the ith iteration, $\mathsf{Gen}(n_{i-1})$ outputs a pair (n_i, a_i) where n_i is a nonce and a_i a random memory address, s_i stores the memory word at the address a_i, and m'_i is the checksum of m_{i-1} and s_i. After N iterations, the prover sends m'_N to the verifier.

Verification The verifier checks that m'_N is correct and that the round-trip time is below a time threshold Δ.

The modifications we introduce to Armknecht et al.'s protocol are depicted in Fig. 4. First, we employ a distance-bounding mechanism at the start of the protocol. This is

a very efficient distance-bounding phase, as it only requires the prover to teleport a nonce back to the verifier. Second, we require the prover to teleport partial results back to the verifier, which are also used to time the performance of the prover. The teleportation process is identical to the one used in the protocol composition described earlier; hence, we do not provide further details on this mechanism.

Like in the previous protocol, the time threshold Δ represents a (possibly tight) bound on the computational time used by the prover to calculate a response. In the case of the extension to Armknecht et al.'s protocol, partial responses are provided every k iterations of the attestation loop. This means that Δ can be tuned up by adjusting k as needed, which is an advantage over the protocol composition described earlier. Note that the distance-bounding mechanism based on those partial responses are not based on round-trip-time measurements, but on a single-trip-time measurement, which is why we require $t' - t \geq \Delta + \frac{\delta}{c}$. Further note that the first time check employed by the verifier is a very efficient (classical) distance-bounding mechanism, consisting of a prover reflecting back the challenge sent by the verifier. In this case, the time threshold Δ plays no role in the proximity check.

4.3 A brief note on the complexity of the protocols

The classical computational complexity of the first protocol that is depicted in Fig. 1 is dominated by the computational complexity of the Attest function and the number of rounds. Our single-round protocol thus offers the same computational complexity as existing memory attestation protocols that make use of an Attest function, such as [2, 13, 17, 26, 27, 29]. The number of rounds thus becomes a multiplicative constant on the complexity of our protocol. One may consider a trade-off between the size of the memory the Attest function covers, which is proportional to its computational cost, and the number of rounds. Our second protocol is indeed one of such trade-offs. With respect to the quantum operations performed by our protocols, namely qubit rotation and teleportation, we note that they are low circuit depth operations (depending on exactly how they are implemented, one and four gates, respectively); hence, they are considered lightweight.

Our second protocol that is depicted in Fig. 4 has $\mathcal{O}(N)$ time complexity, where N is the number of checks performed by the verifier. The functions used by the prover at each round, namely Gen, Read, and CHK, have constant time complexity. As mentioned earlier, the quantum operations required by this protocol are low circuit depth operations, hence very efficient as far quantum operations go.

We conclude that the main limitation of our protocols is not their computational complexity, but the number of qubits that need to be transmitted to verify the state of the memory in the presence of a remote attacker. We analyse this in the next section.

5 Security analysis

In this section, we analyse the security of the two protocols introduced previously. We shall start with the analysis of the protocol composition in Fig. 1 by making security

assumptions on the Attest function. This will help us to focus on the security added by the distance-bounding mechanism based on quantum entanglement, hence building the basis for the analysis of our extension to Armknecht et al.'s protocol.

5.1 Threat model

Existing security analyses of software-based memory attestation protocols assume and out-of-the-band authentication channel, such as visual inspection, and that prover and verifier communicate in isolation without interference from an external/adversarial device. This is known as the *device isolation assumption*. We, however, build our security proofs against the more realistic *distant-attacker assumption* [12], which considers a standard Dolev–Yao attacker only restricted by its (sufficiently large) distance from prover and verifier. In practice, the distant-attacker assumption can be met by establishing a secure physical perimeter that prevents unauthorised access.

The adversary we consider is a man-in-the-middle attacker in full control of the network, meaning it can intercept and modify messages travelling through the network as well as inject new messages. We assume, nevertheless, adversaries restricted in the following ways:

1. *No tampering* They cannot manipulate the prover's hardware, i.e. they cannot add to the prover more CPU or memory. Notably, an adversary cannot give the prover the capacity to transmit entangled qubits.
2. *Distant attacker* The network distance between the adversary and the prover is larger than the network distance between prover and verifier.

In our security proofs, we further assume:

1. *Synchronised quantum state* At the start of the protocol, prover and verifier are capable of agreeing on a sequence of shared entangled qubits. This is used to rule out denial-of-services attacks whereby the adversary desynchronises prover and verifier by *wasting* their qubits.
2. *Restricted quantum communication* Provers can receive, but not transmit entangled qubits. This restriction is reasonable because transmitted qubits require specialised hardware, depending on the physical instantiation of the qubit being used.
3. *Random values are used just once* In particular, values generated by the adversary do not intersect with challenges generated by the verifier. This assumption further simplifies our security proofs.

Observe that denial-of-service attacks, addressed by the synchronised quantum state assumption, are a common threat to quantum information systems, such as BB84 [14]. Systems such as BB84 and our own system have in common the idea that they are attempting to detect an attacker that is hiding. If the attacker decides to stop hiding and instead consume system resources, it is possible to attempt to mitigate this threat as we do here, or as other systems do [22], but it is not possible to prevent it entirely. Depending on the situation, denial of service may not be an advantage to the attacker, since alerting the verifier to the fact that the prover is under attack through the consumption of system resources may be as disadvantageous to the attacker as alerting the verifier through a failure of the security protocol.

5.2 Analysis of the protocol composition under the distant-attacker assumption

We shall analyse two different scenarios with respect to the relative position of verifier, prover and the attacker. In the first scenario, the attacker is close to the verifier and the prover is far. The second scenario is the opposite, the attacker is far from the verifier while the prover is close.

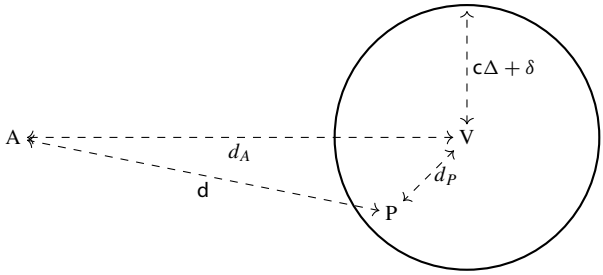
Let Δ be the processing time at the prover side to compute Attest on input the challenge from the verifier. Let d be the distance between the prover and the attacker. Let d_A and d_P be the distances from the attacker and the prover, respectively, to the verifier. And, let δ be a distance upper bound defining proximity, e.g. a few centimetres. Let the function Attest be secure in the following way.

Definition 1 Let P be a corrupt prover with state s' and N a natural number. Given random challenges $c_1, \ldots, c_N$, let $\epsilon(N)$ be the probability of P answering correctly with the sequence $\mathsf{Attest}(c_1, s), \ldots, \mathsf{Attest}(c_N, s)$. We say Attest is secure if $\epsilon(N)$ is negligible.

After the introduction of necessary notation and the security on Attest, we are ready to analyse the simplest case where the attacker is far from the prover and the verifier.

Theorem 1 *Let $\mathcal{P}$ be the protocol in Fig. 1 using N rounds and a time threshold $\Delta + \frac{2\delta}{c}$. Let Attest be a correct attestation function with respect to the time threshold Δ. Assume $d_P < \frac{c(\Delta + \frac{2\delta}{c})}{2}$, i.e. the prover is close to the verifier. If $\mathsf{d} > c(\Delta + \frac{2\delta}{c})$, i.e. the attacker is far, then the probability of success of the attacker is $\epsilon(N)$.*

Proof Because $\mathsf{d} > c(\Delta + \frac{2\delta}{c})$ and $d_A > \mathsf{d} - d_P$ (triangular inequality, see figure below), it follows that $d_A > \frac{c(\Delta + \frac{2\delta}{c})}{2}$. This means that the prover cannot relay information to the attacker, nor the attacker can respond on time to the verifier's challenges. Therefore, the prover has to respond on its own. This gives a probability of success equal to the probability of $\mathsf{Attest}(c, s') = \mathsf{Attest}(c, s)$, which, by Definition 1, is $\epsilon(N)$ over N rounds.



□

The theorem above states that, under that assumption that the attacker is sufficiently far (concretely by $\mathsf{d} > c(\Delta + \frac{2\delta}{c})$), the security of our protocol in the distant-attacker assumption reduces to the security of the Attest procedure in the device isolated assumption. The next result is more relevant, though, as it provides the first security

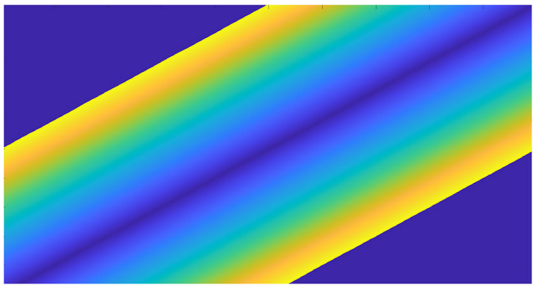


Fig. 5 A graph of $|x - y|, x - y < \frac{1}{2}$

proof of a software-based memory attestation protocol in the presence of a nearby attacker.

Theorem 2 *Let $\mathcal{P}$ be the protocol in Fig. 1 using N rounds and a time threshold $\Delta + \frac{2\delta}{c}$. Let* Attest *be a correct attestation function with respect to the time threshold Δ. Assume $d_A < \frac{c(\Delta + \frac{2\delta}{c})}{2}$, i.e. the attacker is close to the verifier. If $d > c(\Delta + \frac{2\delta}{c})$, then the probability of success of the attacker is $(1 - (1 - 0.1032))^N$.*

Proof As before, we argue that relaying is infeasible. Then, because the attacker has to make the prover to rotate and measure its entangled qubit in order to pass the protocol, it follows that the attacker has to interact with the prover prior the reception of the verifier's challenge. We conclude then that, for the adversary to succeed at a given round, he has to send to the prover a challenge, say c', which is different to the challenge, say c, that the verifier will use for that round. This type of attacks is known in the literature of distance-bounding protocols as *pre-ask* [3] or *pre-computation* attacks [21].

Let θ' be the angle of the qubit transmitted by the prover, on input c'. Let θ be the angle of the qubit expected by the verifier. Because Attest is uniform and the challenge c random, it follows that θ is uniformly distributed, as is the angle between θ and θ'. Because the attacker receives the rotation bits sent by the prover, the attacker has the option of selecting the optimal rotation of θ that better approximates θ'. The four rotations of θ are simply θ rotated by $0, \frac{\pi}{2}, \frac{\pi}{1}, \frac{3\pi}{2}$. The attacker can do so in the following way.

First, observe that if the angle between θ and θ' is lower than $\frac{\pi}{2}$, then there is no benefit to the attacker on changing the rotation established by the prover, since all these vectors are the same or closer to $|0\rangle$, than $|1\rangle$. Now, assume the angle between θ and θ' is greater than or equal to $\frac{\pi}{2}$. If we consider the fixed interval $[0, 1]$, the average distance between two uniformly distributed points can be calculated as $E[D] = \int_0^1 \int_0^1 |x - y| dy dx$. We want to calculate the average distance after all distances $d > .5$ are reduced to $d - .5$. To do this, we need to remove the contribution to this integral from the two regions A, where $|x - y| > .5$, and replace it with two regions B where the contribution is $|x - y| - .5$, corresponding to the adjustment that the attacker is able to make by submitting the wrong rotation. If we consider Fig. 5, we see that A and B

are two triangular regions of a square integral. Therefore, we have

$$A = \int_{\frac{1}{2}}^{1} \int_{0}^{x-\frac{1}{2}} |x-y| dydx$$

$$B = \int_{\frac{1}{2}}^{1} \int_{0}^{x-\frac{1}{2}} |x-y| dydx$$

$$E[D] = \int_{0}^{1} \int_{0}^{1} |x-y| dydx - 2A + 2B$$

$$E[D] = \int_{0}^{1} \int_{0}^{1} |x-y| dydx -$$

$$2(\int_{\frac{1}{2}}^{1} \int_{0}^{x-\frac{1}{2}} |x-y| dydx) +$$

$$2(\int_{\frac{1}{2}}^{1} \int_{0}^{x-\frac{1}{2}} |x-y| dydx)$$

$$E[D] = 0.2082$$

Since we are interested in the fixed interval $[0, \pi]$, we have the average angle after the attack as $\pi * 0.2082$. Returning to the Bloch Sphere representation $sin(\theta/2)|0\rangle + cos(\theta/2)|1\rangle$, we have $0.3212|0\rangle + .0.9470|1\rangle$, which means that there is a $.3212^2 = 0.1032$ chance to measure $|0\rangle$ to detect that the system is in a compromised state. Given the number of rounds N, our chance to detect this attack is $1 - (1 - 0.1032)^N$. □

Two remarks are worth making. First, the probability of success above is far from the ideal $\frac{1}{2^N}$, yet it is still exponentially small. Second, this is a worst case attacking scenario, which defeats existing software-based memory attestation protocols with probability 1.

5.3 Analysis of Armknecht et al.'s protocol extension under the distant-attacker assumption

Our next step is to analyse the security of the protocol in Fig. 4. As before, we will proceed by analysing the two cases arising from the relative position of prover and attacker in relation to the verifier. We shall also build upon the security of the Armknecht et al.'s protocol within the device isolation assumption. Let $\epsilon(N)$ be such security.

Theorem 3 *Let $\mathcal{P}_2$ be the protocol in Fig. 4. Assume $d_P < c\Delta + \delta$, i.e. the prover is close to the verifier. Let p be the probability that, given a random memory address a, $\mathsf{Read}(s', a) = \mathsf{Read}(s, a)$. Let l be the size of the checksum function. If $d > c(\Delta + \frac{2\delta}{c})$, i.e. the attacker is far, then the probability of success of the attacker against $\mathcal{P}_2$ is*

lower than or equal to

$$\max\left(\frac{1}{2^N}, \left(\frac{1-p^N}{2^l} + p^N\right) \sum_{i=1..N} p^{i-1}(1-p)\left(\frac{1}{4}\right)^{N-\lceil i/k \rceil} + p^N\right)$$

Proof First, let's analyse the protocol under the device isolation assumption. Let X be the random variable indicating the round number at which $\mathsf{Read}(s', a_i) \neq \mathsf{Read}(s, a_i)$. It follows that $\Pr[X = i] = p^{i-1}(1-p)$. We use $\Pr[X = 0] = p^N$ to denote the case where, after N rounds, $\mathsf{Read}(s', a_i) = \mathsf{Read}(s, a_i)$ for all $i \in \{1, \ldots, N\}$.

If $X = i$, then, starting from round i and after normalisation of the bit-sequences m'_i and m_i, the difference between the angles m'_i and m_i will be uniformly distributed in $[0, \pi]$. Since the average distance between two uniformly distributed points in an L interval is $\frac{L}{3}$, the average angle between m'_i, m_i is $\frac{\pi}{3}$. Given the Bloch Sphere representation $sin(\pi/6)|0\rangle + cos(\pi/6)|1\rangle$, this means that we have $.5|0\rangle + .8860|1\rangle$, which means that there is a $.8860^2 = .75$ chance to measure $|1\rangle$, and a .25 chance to measure $|0\rangle$. So the chance that we measure $|0\rangle$ and detect that the system is in a compromised state is $\left(\frac{1}{4}\right)^{N-\lceil i/k \rceil}$. This gives a probability of success:

$$\begin{aligned} &= \epsilon(N) \sum_{i=1..N} \Pr[X = i]\left(\frac{1}{4}\right)^{N-\lceil i/k \rceil} + \Pr[X = 0] \\ &= \epsilon(N) \sum_{i=1..N} p^{i-1}(1-p)\left(\frac{1}{4}\right)^{N-\lceil i/k \rceil} + p^N \end{aligned}$$

Now, observe that $\epsilon(N) = \frac{1}{2^l}\sum_{i=1..N} \Pr[X = i] + \Pr[X = 0] = \frac{1-p^N}{2^l} + p^N$. This gives a probability of success equal to:

$$\left(\frac{1-p^N}{2^l} + p^N\right) \sum_{i=1..N} p^{i-1}(1-p)\left(\frac{1}{4}\right)^{N-\lceil i/k \rceil} + p^N$$

Now, let's analyse the case where the prover colludes with the far-away attacker. Assume that m_i was received from the attacker, which means the attacker should have known $r_{\lfloor (i-1)/k \rfloor}$. This could be achieved in three ways:

1. *Relaying* This is infeasible because the distance between the prover and the attacker is larger than $\mathsf{c}(\Delta + \frac{2\delta}{\mathsf{c}})$, which violates the time threshold used by the verifier.
2. *Random guessing* In this case, the probability of success of the attacker is $1/4$, because $r_{\lfloor (i-1)/k \rfloor}$ is a random variable with co-domain $\{00, 01, 10, 11\}$.
3. *Fixing the rotation bits* This is the last resort for the attacker to provide a response for the prover on time. It consists of the prover committing to send a pre-established rotation bit-sequence, say 00, to the verifier, allowing the attacker to calculate m_i without being restricted by the communication channel with the prover. The probability of success of this case is $1/2$.

This gives a maximum probability of success for the prover equal to $1/2^N$. Depending on the value of p, the prover will find colluding more advantageous than responding on its own. □

The proof of the next theorem follows a similar line of reasoning to Theorem 2.

Theorem 4 *Let $\mathcal{P}_2$ be the protocol in Fig. 4. Assume $d_A < c\Delta + \delta$, i.e. the attacker is close to the verifier. If $d > c(\Delta + \frac{2\delta}{c})$, i.e. the prover is far, then the probability of success of the attacker is* $(1 - (1 - 0.1032))^N$.

5.4 The role of the security parameters

We conclude this section by discussing the role of the parameters δ, d and Δ. The key premise in our security proofs is $d > c(\Delta + \frac{2\delta}{c})$. For the protocol in Fig. 1, this means using an efficient Attest function and a tight round-trip-time bound, together with the establishment of a secure perimeter capable of making d as large as possible. For the protocol in Fig. 4, this means adjusting the value of k based on the value of δ. In any case, the bound $d > c(\Delta + \frac{2\delta}{c}))$ is conservative, because it assumes an unbounded attacker that can respond in zero time. The bound does not account either for the time necessary for the prover to relay information back and forth, making our bounds rather conservative.

6 Discussion and conclusions

If we wish to build the device with the technology currently available, we will have to work with the limited quantum memory that exists. While our proposal envisions holding qubits in superposition indefinitely until the verifier wishes to test the prover, in practice current quantum memory does not last that long [4]. Therefore, we will have to exchange the entangled qubits at run time. In practice, this will mean we need to authenticate the prover with the verifier. Quantum Authentication requires a shared secret. In our original proposal, this secret is the pre-shared qubits, but if we exchange these at runtime, we must resort to a pre-shared classical secret (a key), which is a common approach for many Quantum Key Exchange systems.

Therefore, in the near term, our device could be combined with other approaches to this problem, such as a trusted module to store the key material. We then have the benefits described by our system, in combination with the benefits of a classical approach. Once long term Quantum memory is readily available, this will cease to be an issue.

An additional question that we do not address in this paper is the issue of noise. All quantum devices suffer from noise, and the impact of noise measurements must be included in any physical quantum device. We have carried out our analysis with a perfect quantum system in mind, but in practice noise impacts the success rate of all measurements. The typical approach to handling this problem is to increase the number of qubits used. We leave these calculations for future work.

Overall we have presented the theoretical basis for a method for improving the security of remote devices using quantum entanglement. Our method requires a relatively low number of qubits and gates, and as Quantum Information Science advances and the ability for devices to manipulate quantum effects increases, our method is well placed to improve the security of remote devices.

Acknowledgements Rolando Trujillo-Rasua is funded by a Ramon y Cajal grant from the Spanish Ministry of Science and Innovation and the European Union (REF: RYC2020-028954-I). His research also receives funds from INCIBE and NextGenerationEU via the projects HERMES and INCIBE-URV.

Author Contributions JL and RT wrote the main manuscript text. Both authors reviewed the manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Data Availability No datasets were generated or analysed during the current study.

Declaration

Conflict of interest The authors declare no competing interests.

References

1. Abidin, A., Eldefrawy, K., Singelee, D.: Entanglement-based mutual quantum distance bounding (2023). arXiv:2305.09905
2. Armknecht, F., Sadeghi, A.R., Schulz, S., et al.: A security framework for the analysis and design of software attestation. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. Association for Computing Machinery, New York, NY, USA, CCS '13, pp. 1–12 (2013). https://doi.org/10.1145/2508859.2516650
3. Avoine, G., Bingöl, M.A., Boureanu, I., et al.: Security of distance-bounding: a survey. ACM Comput. Surv. (2018). https://doi.org/10.1145/3264628
4. Bashkansky, M., Black, A.T., Kwolek, J.M., et al.: Quantum Memory, pp. 1–17 (2021). https://doi.org/10.1002/047134608X.W8412
5. Bennett, C.H., Brassard, G., Crépeau, C., et al.: Teleporting an unknown quantum state via dual classical and Einstein–Podolsky–Rosen channels. Phys. Rev. Lett. **70**, 1895–1899 (1993). https://doi.org/10.1103/PhysRevLett.70.1895
6. Castelluccia, C., Francillon, A., Perito, D., et al.: On the difficulty of software-based attestation of embedded devices. In: Proceedings of the 16th ACM Conference on Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, CCS '09, pp. 400–409 (2009). https://doi.org/10.1145/1653662.1653711,
7. van Dam, W., Yuan, Q.: Quantum online memory checking. In: Theory of Quantum Computation, Communication, and Cryptography (2009)
8. Dutta, A., Pathak, A.: A short review on quantum identity authentication protocols: how would Bob know that he is talking with Alice? (2021). arXiv:2112.04234 [quant-ph]
9. Eldefrawy, K., Rattanavipanon, N., Tsudik, G.: Hydra: hybrid design for remote attestation (using a formally verified microkernel). In: Proceedings of the 10th ACM Conference on Security and Privacy

in Wireless and Mobile Networks. Association for Computing Machinery, New York, NY, USA, WiSec '17, pp. 99–110 (2017). https://doi.org/10.1145/3098243.3098261
10. Feng, W., Qin, Y., Zhao, S., et al.: AAoT: lightweight attestation and authentication of low-resource things in IoT and CPS. Comput. Netw. **134**, 167–182 (2018). https://doi.org/10.1016/j.comnet.2018.01.039
11. Francillon, A., Nguyen, Q., Rasmussen, K.B., et al.: A minimalist approach to remote attestation. In: 2014 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1–6. IEEE (2014)
12. Gil-Pons, R., Horne, R., Mauw, S., et al.: Is eve nearby? Analysing protocols under the distant-attacker assumption. In: 2022 2022 IEEE 35th Computer Security Foundations Symposium (CSF) (CSF), pp. 17–32. IEEE Computer Society, Los Alamitos, CA, USA (2022). https://doi.org/10.1109/CSF54842.2022.00002
13. Gligor, V.: Establishing and maintaining root of trust on commodity computer systems. In: Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, Asia CCS '19, pp. 1–2 (2019). https://doi.org/10.1145/3321705.3329913
14. Hong, K.W., Foong, O.M., Low, T.J.: Challenges in quantum key distribution: a review. In: Proceedings of the 4th International Conference on Information and Network Security. Association for Computing Machinery, New York, NY, USA, ICINS '16, pp. 29–33 (2016). https://doi.org/10.1145/3026724.3026738
15. Khan, M., Aman, M., Sikdar, B.: Soteria: a quantum-based device attestation technique for the internet of things. IEEE Internet Things J. (2023). https://doi.org/10.1109/JIOT.2023.3346397
16. Kong, J., Koushanfar, F., Pendyala, P.K., et al.: PUFatt: embedded platform attestation based on novel processor-based PUFs. In: 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6 (2014). https://doi.org/10.1145/2593069.2593192
17. Kovah, X., Kallenberg, C., Weathers, C., et al.: New results for timing-based attestation. In: 2012 IEEE Symposium on Security and Privacy, pp. 239–253 (2012). https://doi.org/10.1109/SP.2012.45
18. Kuang, B., Fu, A., Susilo, W., et al.: A survey of remote attestation in internet of things: attacks, countermeasures, and prospects. Comput. Secur. **112**, 102498 (2022). https://doi.org/10.1016/j.cose.2021.102498
19. Li, X., Zhang, D.: Quantum information authentication using entangled states. In: International Conference on Digital Telecommunications (ICDT'06), pp. 64–64 (2006). https://doi.org/10.1109/ICDT.2006.66
20. Li, Y., Cheng, Y., Gligor, V., et al.: Establishing software-only root of trust on embedded systems: facts and fiction. In: Revised Selected Papers of the 23rd International Workshop on Security Protocols XXIII—Volume 9379, pp. 50–68. Springer, Berlin (2015). https://doi.org/10.1007/978-3-319-26096-9_7
21. Mauw, S., Toro-Pozo, J., Trujillo-Rasua, R.: A class of precomputation-based distance-bounding protocols. In: IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21–24, 2016, pp. 97–111. IEEE (2016). https://doi.org/10.1109/EuroSP.2016.19
22. Price, A.B., Rarity, J.G., Erven, C.: A quantum key distribution protocol for rapid denial of service detection. EPJ Quantum Technol. **7**(1), 8 (2020)
23. Rührmair, U., Sehnke, F., Sölter, J., et al.: Modeling attacks on physical unclonable functions. In: Proceedings of the 17th ACM Conference on Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, CCS '10, pp. 237–249 (2010). https://doi.org/10.1145/1866307.1866335
24. Rührmair, U., Schlichtmann, U., Burleson, W.: Special session: how secure are PUFs really? On the reach and limits of recent PUF attacks. In: 2014 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1–4 (2014). https://doi.org/10.7873/DATE.2014.359
25. Schulz, S., Sadeghi, A.R., Wachsmann, C.: Short paper: lightweight remote attestation using physical functions. In: Proceedings of the Fourth ACM Conference on Wireless Network Security. Association for Computing Machinery, New York, NY, USA, WiSec '11, pp. 109–114 (2011). https://doi.org/10.1145/1998412.1998432
26. Seshadri, A., Perrig, A., van Doorn, L., et al.: Swatt: software-based attestation for embedded devices. In: Proceedings of the IEEE Symposium on Security and Privacy, 2004, pp. 272–282 (2004). https://doi.org/10.1109/SECPRI.2004.1301329

27. Seshadri, A., Luk, M., Shi, E., et al.: Pioneer: verifying code integrity and enforcing untampered code execution on legacy systems. SIGOPS Oper. Syst. Rev. **39**(5), 1–16 (2005). https://doi.org/10.1145/1095809.1095812
28. Trujillo-Rasua, R.: Secure memory erasure in the presence of man-in-the-middle attackers. J. Inf. Secur. Appl. **57**, 102730 (2019)
29. Yan, Q., Han, J., Li, Y., et al.: A software-based root-of-trust primitive on multicore platforms. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, ASIACCS '11, pp. 334–343 (2011). https://doi.org/10.1145/1966913.1966957