Virtual Proofs of Reality

Ulrich Rührmair
ruehrmair@ilo.de

***Abstract*—In this paper, we discuss the question how *physical statements* can be proven remotely over digital communication channels *without* using classical secret keys, and *without* assuming tamper-resistant and trusted measurement hardware in the location of the prover. Examples for the considered physical statements are: *(i)* "the temperature of a certain object is X°C", *(ii)* "two certain objects are positioned at distance X", or *(iii)* "a certain object has been irreversibly altered or destroyed". In lack of an established name, we would like to call the corresponding security protocols *"virtual proofs of reality"* (VPs).**

While a host of variants seems conceivable, this paper focuses on VPs in which the verifier has handed over one or more specific physical objects O_i to the prover at some point prior to the VP. These *"witness objects"* assist the prover during the proof, but shall not contain classical digital keys nor be assumed tamper-resistant in the classical sense. The prover is allowed to open, inspect and alter these objects in our adversarial model, only being limited by current technology, while he shall still be unable to prove false claims to the verifier.

In order to illustrate our concept, we give example protocols built on temperature sensitive integrated circuits, disordered optical scattering media, and quantum systems. These protocols prove the temperature, destruction/modification, or relative position of witness objects in the prover's location. Full experimental realizations of these schemes are beyond the scope of this paper. But the protocols utilize established technologies from the areas of physical unclonable functions and quantum cryptography, and hence appear plausible also without such proof. Finally, we also discuss potential advancements of our method in theory, for example *"public virtual proofs"* that function without exchanging witness objects O_i between the verifier and the prover.

Our work touches upon and partly extends several established cryptographic and security concepts, including physical unclonable functions, quantum cryptography, and interactive proof systems.

***Keywords*-Virtual Proofs of Reality (VPs), Physical Unclonable Functions (PUFs), Interactive Proof Systems, Quantum Cryptography, Physical Cryptography**

I. Introduction and Overview

The archetypical cryptographic setting consists of two or more remote parties who are connected via a digital channel. By using the latter, they want to accomplish a certain cryptographic or security task. Popular examples include the secure exchange of a secret key; confidential or authenticated communication; secure mutual identification; interactive proofs [22]; or zero-knowledge protocols [21]. All these tasks predominantly have a logical or mathematical nature, and can be expressed in a purely mathematical framework: The involved parties are usually modeled as Turing machines; the goals they want to accomplish are formulated as mathematical statements; and their security often relies on the conjectured hardness of mathematical problems.

A second aspect said tasks have in common is that their practical realization usually involves secret keys or other secret information. However, the secure storage of such keys in hardware has turned out non-trivial in practice: Malware, invasive techniques or sidechannel attacks can extract secrets and lead to security breaks [1]. As Ron Rivest put it in a keynote speech at Crypto 2011, *"calling a key 'secret' does not make it so, but rather identifies it as an interesting target for the adversary"* [42]. This suggests that standard secret keys should be avoided whenever possible.

In this paper, we thus investigate a twofold extension of the above classical security setting. We consider the following questions:

(i) How can one party *(the "prover")* prove *physical* statements over digital communication lines to another party *(the "verifier")*? Which set-up assumptions are required for such proofs?

(ii) How can such proofs be led *without* using classical secret keys and tamper-resistant security hardware at the location of the prover?

To develop a feeling for the problem, consider the classical solution to question (i). It would consist of placing a piece of tamper-resistant hardware in the prover's location. The hardware would be set-up by the verifier in a phase prior to the proof, containing a secret key of his. It could verify a physical statement claimed by the prover simply by making autonomous and independent measurements. Subsequently, it would send some measurement values in a cryptographically authenticated form to the verifier, allowing him to check the claim of the prover. Well-known examples for this technique include classical security sensors and cameras. Quite obviously this approach does not address or resolve question (ii), however, as it presupposes secret keys. If such keys shall be avoided, new mechanisms are required.

Along these lines, we present four methods in this paper that can interactively prove certain physical statements over digital channels, and which do so without using classical secret keys or tamper-resistant hardware at the location of the prover. All presented approaches assume that in some secure set-up phase prior to the actual proof, the verifier has

fabricated a number of physical objects O_i and transfered them to the location of the prover. These so-called *"witness objects"* later assist the prover in the virtual proof (VP), and can be re-used in several executions of the VP. They do not contain any classical secret keys and are not assumed as tamper-resistant, i.e., a malicious prover can try to open and inspect them, only being limited by current technology in his efforts. The general employment of witness objects is not as restrictive as it may seem at first glance. In practice, both the prover and the verifier depend on electronic devices to carry out their communication, which makes some sort of "additional" object or device on both sides inevitable. The secure set-up phase, on the other hand, can be laborious in certain practical appliances, while it may simple in others.

Our first method utilizes a special type of integrated circuit as witness object. Its unclonable manufacturing variations and temperature-sensitive input-output behavior are exploited to prove its temperature to the verifier. Our second method proves the relative distance of two witness objects. It employs optical techniques, using witness objects that are reminiscent of Pappu's optical PUF [40], [39]. The third method employs a sequence of random quantum systems, and proves that these systems have been measured or irreversibly modified, respectively. Our technique exploits that quantum systems with an unknown state can neither be cloned nor measured without altering them. Our fourth and final method uses a complex, disordered optical system as witness object which is reminiscent of Pappu et al.'s optical PUF [39], [40], and proves that this object has been irreversibly modified. Its security is built on the unclonability of the witness object and on the complexity and non-simulatability of its internal optical scattering process.

In the last part of the paper, we discuss possible extensions of the above type of VP. We investigate whether it could be possible to avoid the private set-up phase, in which the witness objects are prepared by the verifier and transferred to the prover. In particular, we ask whether the *prover himself* could prepare the witness objects, calling the resulting schemes *"public virtual proofs" (public VPs)*. If realized, public VPs would have groundbreaking advantages over standard virtual proofs, comparable to the upsides of public key cryptography over symmetric cryptography. Remarkably, they have no counterpart based on standard security techniques, for example via classical secret keys: If these keys are chosen by the prover himself, he could set up any manipulated hardware to his like, and prove false claims to the verifier. Who would trust the values that are reported by a temperature sensor that has been set-up by the adversary himself? Based on an approach that is related to the recent concept of SIMPL systems [44], [50], [13], [46], [48] or public PUFs [2], [37], [41], however, such public VPs seem possible at least in principle. In the final part of the paper, we lead an abstract discussion that sketches our ideas in this direction.

Related Work: Our work relates to several known notions in cryptography and security. First and foremost, it can be seen as an extension of interactive proofs (IPs) [22] into the physical domain. While IPs let the prover show a mathematical statement, our virtual proofs show statements about the physical world.

In contrast to classical secure sensors or sensor networks, which could also be seen as reporting or "proving" their measured values "interactively" to some central authority, VPs furthermore avoid any classical secret keys in the sensor hardware. They do not assume trust in the sensor hardware, nor suppose that some piece of hardware is untampered!

Another obvious tie exists to physical unclonable functions (PUFs). Similar to PUFs, VPs exploit the physical unclonability of certain disoderded systems for security purposes. Furthermore, some of them employ structures that are very similar to PUFs, or which even have already been used as PUFs in a different context. This includes our VP of temperature, which could be based on temperature-dependent systems like the Bistable Ring PUF of Chen et al. [11], and also our VPs of destruction and distance, which are based on optical systems similar to Pappu et al.'s optical PUFs [40], [39]. Our VPs partly exploit new features of these PUFs, however; for example, the VP of temperature uses the temperature-sensitive behavior of the Bistable Ring PUF, which had not been utilized in earlier security appliances, but rather had been regarded as a practical disadvantage of this PUF architecture. In a similar fashion, our public VPs of Section VI are strongly connected to SIMPL systems [44], [50], [13], [46], [48] and public PUFs [2], [37], [41]. Our work also relates to the Sensor PUFs of Rosenfeld et al. [43]; please note, however, that the latter work merely considers PUF-based cameras, while we take a broader and more foundational approach, considering different VPs. Sensor PUFs and the first appearances of our approach in patent writings in 2009/2010 [61] seem to have been conducted independently of each other.

Finally, our work is linked to the field of quantum cryptography in two ways. Firstly and foremostly, we exploit quantum systems and quantum unclonability in one of our VPs of destruction. Secondly, position-based quantum cryptography [5] has obvious ties to our VPs of relative location. Comparing the two concepts, one advantage of VPs is that several negative findings and impossibility results have been discussed recently on position-based quantum crypto [5], while we present some positive results on VPs of relative distance in this paper. Another upside of VPs is that they require only classical communication channels, no costly quantum communication infrastructure. On the downside, VPs of relative distance are currently limited to small distances and to proving the distance of special witness objects, while quantum proofs of location would potentially be applicable in a broader context.

Earlier Version of this Work: The material in this manuscript has been presented in various forms at earlier occasions, which we would like to mention for completeness: First of all, the basic idea of a VP, as well as concrete VPs of temperature and destruction, have been discussed already in a patent writing from 2009/2010 [61]. The author has also discussed the idea in public, invited talks at SOFSEM 2011 [62] and at the RISC seminar at CWI in Amsterdam in 2012 [63].

Organization of this Paper: Our paper is organized as follows. Section II stipulates the general setting and terminology of VPs. VPs of sensor data, location, and destruction, are treated in Sections III to V, respectively, together with plausibilizations of the experimental viability of our concepts. Public VPs are discussed in theory in Section VI. We conclude the paper in Section VII.

II. General Setting and Terminology

Before moving on, let us detail the exact setting of virtual proofs (VPs) and their terminology. We assume in a VP that two parties are sitting in two different physical systems S_1 and S_2, and can communicate with each other over a digital channel. Third parties can eavesdrop this channel, but cannot insert or modify messages in any way. Apart from the digital content of the messages, also the timing by which they arrive at the two parties may be exploited in the proof.

The party sitting in S_1, the *"prover"*, wants to prove a physical statment to the other party, the *"verifier"*, over the channel. The statement describes some physical feature or phenomenon in the prover's system S_1. The proof shall achieve completeness in the sense that the prover can indeed convince the verifier with high probability if the claimed statement is true. It shall also achieve correctness in the sense that the verifier will notice with high probability if the prover tries to convince him of a false statement.

Even though we make no such general assumption, we may optionally assume in *some* of our arguments that S_1 is a "closed" physical system, i.e., that S_1 has no physical exchange of any sort with the outside, apart from the (abstract and idealized) digital channel. This reflects practical situations where the prover sits in a closed and controlled environment, for example where the role of the prover is played by a bank card inside an automated teller machine (ATM). Such closedness assumptions can also be helpful in deriving certain impossibility arguments on VPs (compare the *"Future Work"* paragraph of Section VII). Similar closedness assumptions can in principle also be made on the verifier's system S_2 whenever appropriate.

Two different types of virtual proofs must be distinguished. In a VP with a private set-up phase (also called *"private VP"*), we allow the verifier to prepare k physical objects $O_1, \ldots, O_k$ prior to the start of the actual proof. In this phase, he can measure some characteristics of these objects and store them privately, without the prover knowing what was stored. After the set-up phase, the objects O_i are transfered to the prover's system S_i, and are being used in the VP later on. In a so-called *"public virtual proof (public VP)"*, the prover may still use a number of objects O_i in the proof, but no secure set-up phase or transfer of objects prior to the proof is assumed. The prover is indeed allowed to fabricate all objects O_i by himself. Both in a private and a public VP, the objects O_i are termed *"witness objects (WOs)"*. As mentioned earlier, these WOs shall not contain any classical secret keys nor be assumed actively tamper-resistant. The prover is allowed to open and inspect them, only being limited in his efforts by current technology.

The situation is summarized in Figure 1.

A. Interpreting Certain PUF-Protocols as Virtual Proofs

Given the setting described above, some well-known protocols that are based on PUFs and related structures can be interpreted as special cases of VPs. For example, the classic PUF-based identification protocol by Pappu et al. [39], [40] could be seen as a *private virtual proof of possession*: Any prover who is sitting in a *closed* system S_1 can show to a verifier, who fabricated a PUF and holds a private CRP-database of it, that he is now in possession of this PUF, i.e., that the PUF is located within S_1. Without the assumption that the prover's system S_1 is closed, the above identification protocol can still be seen as a *private virtual proof of access* to the PUF.

A closely related example is an identification protocol based on *"SIMPL systems"* suggested by Rührmair [44], [46]. This protocol can be seen as a *public* virtual proof of possession: A prover sitting in a closed physical system S_1, who fabricated a SIMPL system by himself, shows to a verifier over a digital communication channel that he indeed holds this SIMPL system. The proof exploits the timinig by which the messages arrive over the digital channel. In this case, no data on the SIMPL system needs to be stored privately at the verifier. Rather, a full description of the SIMPL system can be made public without compromising the protocol's security. The prover could also fabricate the SIMPL system himself, again without compromising security. Similar schemes have been suggested in the context of the public PUFs of Beckmann and Potkonjak [2], [37], [41], or of the time-bounded authentication methods of Majzoobi, Elnably and Koushanfar [34], [35].

As a final example, consider the well-known use of PUFs for tamper detection. Pappu et al. [39], [40], Gassend [18], and Tuyls et al. [69] suggest that tamper-sensitive PUFs can encapsulate valuable hardware systems. Previously collected PUF-CRPs can then be used to check that the physical integrity of the encapsulating PUF has not been violated or damaged — for example, that no one has drilled into it or to remove it. This could be interpreted as a *virtual proof of the physical integrity* of the surrounding PUF-layer within our new framework. Again, we remark that public variants

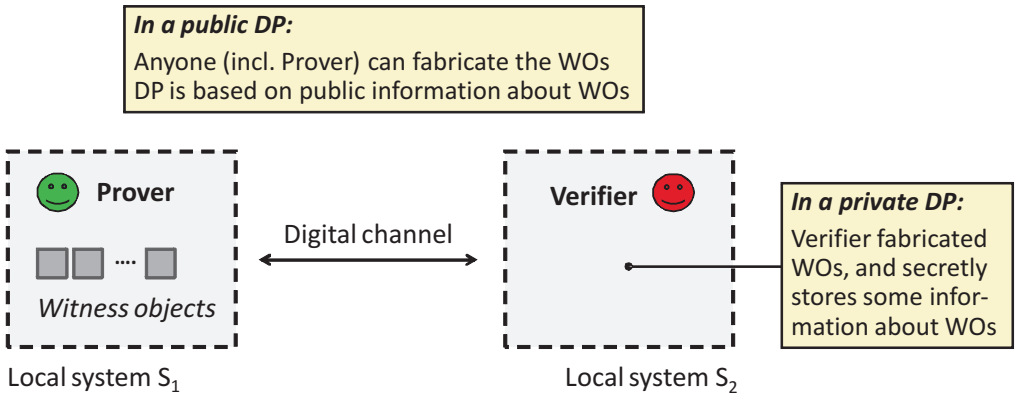

Figure 1. The general setting of public and private VPs based on witness objects (WOs). The WOs shall neither contain secret keys nor be assumed as tamper-resistant.

of this approach could be built on SIMPL systems or public PUFs.

The fact that several known PUF-protocols can be regarded as special cases of VPs indicates the generality of our new concept. In the remainder of this paper, we will now deal with novel, previously unconsidered VPs.

III. Virtual Proofs of Sensor Data

One of our novel VPs are so-called *VPs of sensor data*, in which the prover wants to show that some claimed sensor data measured in his system S_1 is correct. Potential examples include sensors for relatively simple variables like temperature, pressure, humidity, current, etc., as well as more complex sensors such as cameras or microphones.

We deal with one exemplary case, namely *VPs of temperature*, throughout this section. Their general idea is described in Section III-A, and possible extensions to other physical variables are discussed in theory in Section III-B. We focus on *private* VPs of sensor data throughout the entire section, i.e., we assume that the verifier has fabricated the sensor (but the sensor is not allowed to contain any secret keys).

A. Virtual Proofs of Temperature

Strong PUFs are a PUF variant which, by definition, possesses a particularly complex input-output behavior and a very large number of possible challenges [57], [49]. One known aim of electrical Strong PUF design is to combine said complexity with a stable input/output behavior, in particular with stability against temperature variations. [1] We show in this section that such temperature dependence is not necessarily a curse only. It can also be a blessing, since it facilitates VPs of temperature. The following protocol describes our approach.

Protocol 1: Electrical VP of Temperature

[1]We stress that in practice, this goal has already been solved efficiently, for example by the differential design of Arbiter PUFs [19], [67], which exploit relative (not absolute) signal delays.

Assumptions:

- We assume that the prover wants to show the temperature of one specific witness object, in this case one particular electrical Strong PUF, within a temperature range $\mathbf{R_T}$. This temperature range is discretized at a certain resolution, resulting in k discrete temperature levels $t_1, \ldots, t_k \in \mathbf{R_T}$.
- The electrical Strong PUF is assumed to be temperature dependent in its behavior. I.e., its responses R_j^i are a function not only of the applied challenges C_j, but also of the current (discretized) temperature t_i of the PUF: $R_j^i = F_{\mathsf{PUF}}(C_j, t_i)$.
- The behavior of the eletrical PUF varies unpredictably for the above discrete temperature levels $t_1, \ldots, t_k$. Knowing many outputs $R_j^i = F_{\mathsf{PUF}}(C_j, t_i)$ for various challenges C_j and temperatures t_i does not allow to predict unmeasured PUF-responses R_r^s for new temperatures $t_r \neq t_i$ or new challenges $C_s \neq C_j$.

Set-Up Phase:

- The verifier prepares an electrical, temperature-dependent Strong PUF with the above properties.
- He determines a private CRP-list $\mathcal{L}$ for this PUFs as follows:
  - For all considered temperature levels $t_1, \ldots, t_k$, he iterates the following procedure:
    - He puts the PUF at temperature t_i.
    - For $j = 1, \ldots, m$, he randomly chooses challenges C_j^i and applies it to the PUF (at temperature t_i). He measures the resulting response R_j^i.
  - The list $\mathcal{L}$ is then defined as $\mathcal{L} = (C_j^i, R_j^i, t_i)$ for $i = 1, \ldots, k$ and $j = 1, \ldots, m$.
- The verifier privately stores $\mathcal{L}$ and transfers the PUF to the prover.

Virtual Proof:

1) The prover claims to the verifier that the PUF is at a temperature $T \in \{t_1, \ldots, t_k\}$.
2) For $v = 1, \ldots, n$, the verifier randomly selects a tuple (C_v, R_v, T) from the list $\mathcal{L}$, and sends the value C_v to the prover.
3) For $v = 1, \ldots, n$, the prover applies the challenge C_v to the PUF, measures the response R_v^*, and sends this response to the verifier.
4) For $v = 1, \ldots, n$, the verifier compares the received value R_v^* to the values R_v in his list $\mathcal{L}$. If all values match [2], he accepts the virtual proof. Otherwise, he rejects.
5) For $v = 1, \ldots, n$, the verifier erases the tuple (C_v, R_v, T) from the list $\mathcal{L}$.

Discussion: The parameter n determines the security of the scheme. Assuming that the VP shall be executed w times, the value m should be set to $m = wn$, resulting in a list $\mathcal{L}$ of size $\Theta(wnk)$. Even though $\mathcal{L}$ may be relatively large in practice, it is still linear in the involved parameters.

The above technique allows proving the temperature in discrete levels of a certain stepwidth. Too small temperature changes will not affect the challenge-response behavior of the PUF in a detectable and stable manner. This puts an obvious lower bound on the resolution by which the temperature can be proven. A VP of a truly continuous variable in the limit would require an infinitely small stepwidth, leading to a list $\mathcal{L}$ of infinite size. In general, smaller stepwidths have to be paid for by larger lists $\mathcal{L}$ and by a more careful design of the underlying Strong PUF.

It is also interesting to ask what the above scheme exactly proves. Actually, the verifier can conclude that within the period between the times at which the last value C_v has been sent away by him in Step 2 and the last value R_v^* has been received by him in Step 3 of the protocol, the witness object (with very high probability) was at temperature T at k points in time. In addition, the sensor/witness object by which this measurement was made is uniquely identified in the protocol. The verifier can conclude that the responses R_v^* have been obtained from this very sensor within the above time period. Finally, if the system S_1 is assumed to be closed, then also the location of the sensor is proven to lie within S_1.

B. Extensions

The above approach generalizes easily to other simple, one-dimensional physical variables Φ, provided that Strong PUFs can be designed whose output $R_i = F_{\mathsf{PUF}}(C_i, \Phi)$ depends on Φ in a suitable manner. In these cases, Protocol 1 applies with only minor modifications. Along these lines, VPs of the current or voltage at an electrical component seem possible, or VPs of altitude, humidity, pressure, etc., provided that suitable witness objects can found. The design of such WOs appears as an interesting future research task.

A second relevant topic are VPs for more complex sensors, such as cameras or microphones. This task appears problematic at first sight: Consider a camera with p pixels, each of which can be at s states. Such a camera has p^s possible images as inputs. If we apply our above strategy of generating a CRP-list $\mathcal{L}$ (see Protocol III-A) in this context, we end up with a list of exponential length. The list would have length $\Theta(wnp^s)$, with n being the security parameter and w denoting the number of times the VP shall be repeated.

One potential solution are *public* virtual proofs, which are discussed in detail in Section VI, in which lists like $\mathcal{L}$ are unnecessary. A perhaps simpler option is to lead VPs for every single pixel of the camera. The witness list $\mathcal{L}$ then would have to include CRPs for the behavior of every single pixel under different local inputs of the pixel. In this case, the above "witness list" $\mathcal{L}$ becomes quite large, but still linear in the number of pixels p and their states s: It is of order $\Theta(wnps)$. The same approach could in principle also be applied to microphones: Individual VPs could be led for each of many discretized frequencies $f_1, \ldots, f_k$ of a complex sound signal, similar to the discretized approach of Protocol 1. Working out the details of such VPs constitutes a worthwhile topic for future reseach activities.

C. Possible Implementation

In the literature on PUFs, the temperature dependence of these structures is usually regarded as a problem. In our context, it can be turned into an advantage, and be used for something meaningful. From the many currently existing Strong PUF implementations (the above scheme needs a Strong PUF, since the used PUF must have many possible challenges), one good candidate seems the Bistable Ring PUF [11], [12]. It is known that a significant fraction of its CRPs (between 5.81% and 9.9% [11], [12], i.e., still an exponentially large absolute number of CRPs) are dependent on the external temperature, but are relatively stable upon multiple measurements at one fixed temperature level [11], [12]. Usually, these CRPs are discarded in standard PUF protocols, but in our VPs of temperature, it is exactly this fraction of CRPs that we would focus on. Said CRPs also vary steadily over a large temperature span (see Fig. 5 of [11] and Fig. 6 of [12]), and are furthermore among the unique CRPs for a given PUF instance and temperature [11], [12]. This indicates that current implementations of the BR PUF as in [11], [12] could directly be used in VPs of temperature over a wide temperature range, even though we did not implement this yet in this paper.

A second conceivable example are k-XOR Arbiter PUFs [67], [58] that operate exactly at the stability levels of these architectures, for example for $k = 8$, 12, or 16. Multiple measurement of the same CRP and subsequent majority

[2] Alternatively, the verifier may accept if more values match than given by a previously specified error bound.

voting at one and the same temperature level could stabilize the outputs with respect to voltage variations; this strategy has been applied sucessfully in recent, ultra-high exactness modeling attacks on silicon PUFs [60]. At the same time, the structure would be highly temperature dependent for such large values of k (compare again [60]). This promises usability in our VPs of temperature.

A full proof of concept is beyond the scope of this paper, and is left to future work.

IV. Virtual Proofs of Location

The prover's goal in a so-called *VP of location* is to show statements about the position of one or more physical objects in his system S_1 to the verifier. Several variants are conceivable: In the most general case, the prover may try to show the absolute coordinates (within S_1) of some arbitrary objects to the prover. More special scenarios are that the prover tries to show the absolute coordinates of some special witness objects; the relative location (or distance) between arbitrary objects, or between special witness objects and arbitrary objects; or, finally, the relative location (or distance) of two special witness objects. The latter is obviously the simplest scenario. Below, we present a technique which resolves it, at least for relatively small distances between the two witness objects. It is based on scattering phenomena in disordered optical systems, similar to Pappu's optical PUF [39], [40].

A. Virtual Proofs of Distance

It is well-known that the scattering process in Pappu et al.'s optical PUF [39], [40] and the resulting interference pattern are highly dependent on the exact relative position of the PUF, the laser source, and the recording CCD camera. While this is a curse for the inexpensive practical implementation of this PUF type, it is a blessing in our context: In can be used to prove the relative distance of two optical PUFs, which act as witness objects, to the verifier.

The following protocol gives the details. Our proof makes the assumption that the prover wants to show small distances D to the verifier, and that the interval of possible distance has been suitably discretized.

Protocol 2: Optical VP of Distance

Assumptions:

- We assume that the verifier wants to show the distance of two specific witness objects, in this case two optical PUFs à la Pappu et al. [40], [39], within a distance range $\mathbf{I_D}$. We further assume that this distance range is partitioned equally at a certain stepwidth, with k resulting discretized distances $d_1, \ldots, d_k \in \mathbf{I_D}$.

Set-Up Phase:

- The verifier prepares a first and a second optical PUF à la Pappu et al. [40], [39].
- He determines a private CRP-list $\mathcal{L}$ for these two PUFs as follows:
  - For all considered distances $d_1, \ldots, d_k$, he iterates the following procedure:
    - He places the first and the second PUF at distance d_i to each other, as in the set-up depicted in Figure 2. [3]
    - For $j = 1, \ldots, m$, he randomly chooses challenges $C_j^i = (p_j^i, \Theta_j^i)$, where p_j^i is a coordinate on the first PUF and Θ_j^i a spatial angle.
    - For $j = 1, \ldots, m$, he directs a laser beam at coordinate p_j^i and under angles Θ_j^i at the first PUF, and measures the resulting optical responses R_j^i behind the second PUF. [4]
  - The list $\mathcal{L}$ is defined as $\mathcal{L} = (C_j^i, R_j^i, d_i)$ for $i = 1, \ldots, k$ and $j = 1, \ldots, m$.
- The verifier privately stores the list $\mathcal{L}$ and transfers the two PUFs to the prover.

Virtual Proof:

- The prover claims to the verifier that the first and second PUF are at a distance $D \in \{d_1, \ldots, d_k\}$ in the set-up of Figure 2.
- For $v = 1, \ldots, n$, the verifier randomly selects tuples (C_v, R_v, D) from the list $\mathcal{L}$, and sends the values $C_v = (p_v, \Theta_v)$ to the prover.
- For $v = 1, \ldots, n$, the prover directs a laser beam at coordinate p_v and angle Θ_v to the first PUF in the set-up of Figure 2, measures the resulting optical response R_v^* behind the second PUF, and sends this response to the verifier.
- The verifier compares the received n values R_v^* to the values R_v in his list $\mathcal{L}$. If they match, he accepts the virtual proof, otherwise, he rejects. He erases the n used tuples (C_v, R_v, D) from the list $\mathcal{L}$.

Discussion: The above scheme is, in principle, suited to allow relatively small resolutions, down to the order of the wavelength of the employed laser light. Still, it also has its limitations: Distance changes much smaller than that wavelength can no longer be resolved, as the optical signal will not notably change for such small differences. Furthermore, large distances between the two witness objects (such as meters or larger) also cannot be proven.

[3] To make this yet more precise: The two cuboid-shaped optical PUFs are positioned in such a way that their geometrical centers are on a line that is perpendicular to their largest two surfaces, and that the distance between their nearest neighbouring surfaces is d_i.

[4] Again to be precise, these responses will usually not be the raw interference patterns, but the result of an image transformation that is applied to these patterns, for example the Gabor transformation [39], [40] or other suitable transformations [55].

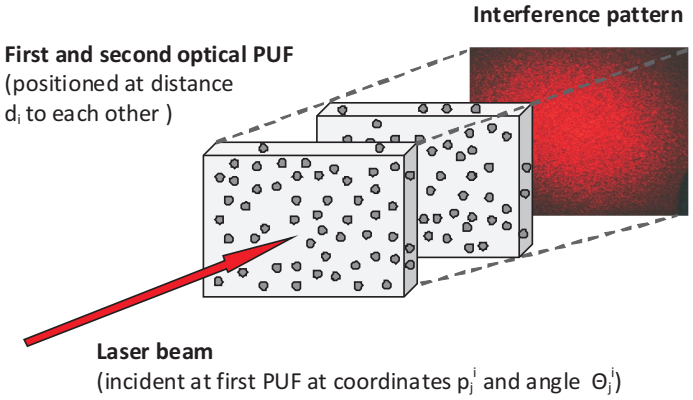

Figure 2. The basic set-up of an optical virtual proof of relative distance. Both optical PUFs participate in the interference process.

One important subcase of VPs of distance are *VPs of co-locality*, where the prover wants to show that two objects are in direct neighbourhood to each other. The above schemes easily can be used for such an approach: It can prove that the two witness objects have a distance smaller than the resolution of the VP. In any case, this suffices to show that the WOs are closer than the wavelength of the employed laser light.

Also integrated circuits (ICs) could be used for VPs of distance or co-locality, even though we did not follow this route in this paper. Analog circuits would be natural candidates, but also digital ICs could be suitable. The basic idea would be to design two communicating ICs, which execute some form of "joint computation". The outcome of the computation should depend on their distance; in particular, it should either be impossible or at least computationally laborious to emulate the outcome of the computation if the circuits are further apart than claimed.

In this context, it is important to see that the information exchange between the two ICs is limited by the speed of light. At the same time, the ICs could operate at GHz frequencies, i.e., one clock cycle every nanosecond. Within this small time period, light travels only 30cm. This would give hope to achieve VPs of distance at resolutions of 30cm. At the least, it might be sufficient to distinguish between the case where the two circuits are in direct proximity and the case where they are at a larger distances on the order of kilometers. This would already suffice for very basic VPs of co-locality and simple security applications.

B. Possible Implementation

By using existing technology on optical PUFs, it seems definitely plausible to implement the above type of VP. In more detail, Pappu et al. [39], [40] have described how an optical scattering token can be re-placed multiple times within a measurement unit in such a way that the resulting speckle pattern is reliably reproduced. They used kinematic mounts and other, standard mechanical technology to this end. If one of the optical tokens of our VP of distance is fixed within the measurement unit, exactly the same technology would likely allow accomplishment of our VP of distance. Again, a full experimental implementation is not within the scope of this paper, but is planned to be approached in future, dedicated implementation experiments.

V. Virtual Proofs of Destruction

Let us now turn to the last VPs treated in this paper, so-called VPs of destruction. Their existence is somewhat counterintuitive: How should one prove that a certain object has been destroyed? Given a pile of ashes, say, how should the prover argue about the features of the original object before it was vaporized? How could such a proof be led for arbitrary items, not just for specially designed witness objects? Some quick thought illustrates that such a general form of VPs of destruction is extremely difficult to achieve, if not straightforwardly impossible.

There are certain subforms that are simpler to accomplish, however. For example, one might design a VP of destruction in the following manner:

- The prover shows that a first object O_1 is in his possession.
- The prover *"destroys"* or *"irreversibly modifies"* this object to obtain a second object O_2. The nature of the second object O_2 should be such that it is unambiguously clear that O_2 can *only* have been obtained from O_1 by irreversibly modifying O_1.
- The prover shows that the second object O_2 is in his possession.

The challenge here is to choose a suitable object O_1 and a suitable physical modification on O_1. It should allow a proof that O_2 unambiguously originates from $O1$. In the rest of this section, we present two constructions to this end, one optical and one quantum mechanical. The schemes work only for a special form of "destruction", in which after the

process of destruction enough structure is left to identify the remaining object, and to establish a link with the original. Subject to personal taste, they could also be called VPs of (irreversible) modification for this reason.

A. Optical Virtual Proofs of Destruction

The following scheme realizes a VP of destruction for an optical system that is reminiscent of Pappu's optical PUF. The idea is to design the system in two stages, with an inner and an outer layer, and to later prove when the outer layer has been removed. The following protocol has the details.

Protocol 3: OPTICAL VP OF DESTRUCTION

Assumptions:

- We assume that the prover wants to show that a certain object has been irreversibly modified or changed.

Set-Up Phase:

- The verifier prepares a first optical PUF, for example of cuboid or spherical shape.
- The verifier collects a challenge-response list $\mathcal{L}_1$ for this first PUF. I.e., he directs a laser beam under a number of randomly chosen points and angles of incidence at the first PUF and records the optical responses.
- The verifier fully encapsulates this first PUF within a second optical PUF (see Figure 3), forming a *larger, composed optical PUF*.
  He may use a different material for forming the second PUF, for example one with a different melting point or chemical solubility than the first PUF.
- The verifier collects a challenge-response list $\mathcal{L}_C$ for the composed PUF. I.e., he directs a laser beam under a number of randomly chosen points and angles of incidence at the composed PUF and records the optical responses.
- The verifier transfers the composed PUF to the prover.

Virtual Proof:

1) The prover shows to the verifier that he is still in possession of the composed PUF. To this end, the following steps are executed:
   a) For $v = 1, \ldots, n$, the verifier randomly selects a tuple (C_v, R_v, T) from the list $\mathcal{L}_C$, and sends the value C_v to the prover.
   b) For $v = 1, \ldots, n$, the prover applies the challenge C_v to the composed PUF, measures the response R_v^*, and sends this response to the verifier.
   c) The verifier compares the received n values R_v^* to the values R_v in his list $\mathcal{L}_C$. If they match, he accepts the virtual proof, otherwise, he rejects.
2) The prover removes the encapsulating second PUF from the composed PUF, setting free the first PUF. He can do so, for example, by exploiting the different melting point or solubility of the second PUF.
3) The prover shows to the verifier that he has removed the encapsulating second PUF and revealed the first PUF. This shows that he has irreversibly modified the composed PUF. To this end, the following steps are executed:
   a) For $v = 1, \ldots, n$, the verifier randomly selects a tuple (C_v, R_v, T) from the list $\mathcal{L}_1$, and sends the value C_v to the prover.
   b) For $v = 1, \ldots, n$, the prover applies the challenge C_v to the first PUF, measures the response R_v^*, and sends this response to the verifier.
   c) The verifier compares the received n values R_v^* to the values R_v in his list $\mathcal{L}_1$. If they match, he accepts the virtual proof, otherwise, he rejects.

Discussion: What does the above VP actually prove? It shows that an irreversible modification of a specific object, namely the composed PUF, has occured in the time period between the events of the verifier sending away the first challenge C_v in Step 1a and the verifier receiving the last value R_v^* in Step 3b of the protocol. The involved objects (the first PUF and the composed PUF) are uniquely identified in this process. Finally, under the additional assumption that the prover's system is closed, the verifier can conclude that the modification has taken place in S_1.

The protocol combines two standard PUF-like challenge-response protocols with several specific hardware features of the witness objects, i.e., of the composed PUF and the first PUF. These security-relevant hardware features are:

- The composed PUF must have a large number of challenges. Otherwise, a fraudulent prover could read out all possible CRP and falsely prove possession of the composed PUF in Step 1, while in fact the PUF has already been modified.
- The composed PUF must be unclonable. Otherwise, a fraudulent prover could clone it and modify or destroy the clone instead of the original composed PUF, i.e., the unamnbiguous identification of the involved objects is then no longer maintained.
- Physically removing the second PUF from the composed PUF must be a practically irreversible process, i.e., it must be impossible to restore the composed PUF in its original form after the removal.
- Given the composed PUF, it must be impossible to obtain challenge response pairs from the first PUF by any other method (such as special physical measurements or numerical simulations) than removing the second PUF.

If the proof is executed only once in practice and surely will never be re-started (for example due to channel malfunctions or similar practical issues), then the last steps of

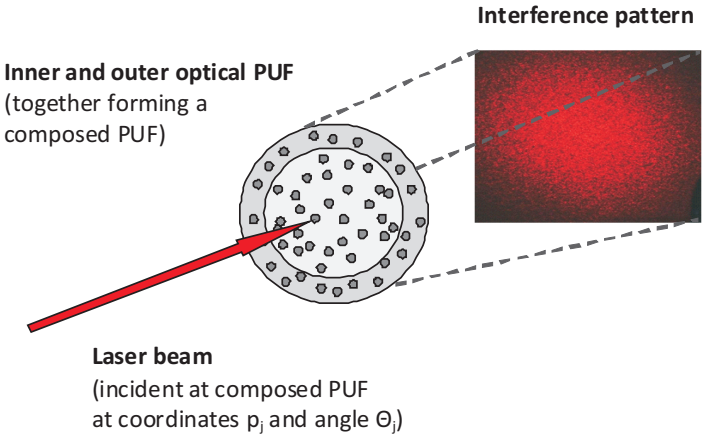

Figure 3. A system composed of two optical PUFs is used in our VPs of destruction: A first inner PUF, and a second outer PUF, which encapsulates the first PUF. Both influence the speckle pattern. The outer PUF cannot be removed without being irreversibly destroyed.

erasing the used CRPs from the lists $\mathcal{L}_1$ and $\mathcal{L}_C$ can be left away, and the lists can be made very short.

Finally, let us remark that by using several onion-like layers around the first PUF, multiple irreversible modifications of an object can be proven in a row. Of course, such a repeated proof must assume that it is impossible to remove, clone or simulate any of these layers (compare our above list of security-relevant features).

B. Quantum Virtual Proofs of Destruction

Let us now illustrate how a second well-known alternative security technology can be exploited in VPs, namely quantum systems. The key observation in our context is that quantum systems in unknown states cannot be measured without disturbing their state. In other words: As long as the original state remains unknown, the original state cannot be rebuilt after measurement. In this sense, quantum measurements bring about some form of "irreversibly destruction" in certain situations.

It is well-known that this effect can be exploited cryptographically, for example in quantum key exchange protocols. An adversary who measures the quantum systems (e.g., polarized photons) in transmission betweeen Alice and Bob will irreversibly disturb them. This can be noticed by Alice and Bob, who may then abort transmission.

A closer look reveals that the same effect can be used to obtain a VP of destruction by quantum technology. The quantum systems (e.g., polarized photons) act as witness objects in this case. The following protocol has the details; it is very similar to the Bennett-Brassard key exchange protocol [4], and assumes some familiarity with this protocol.

Protocol 4: QUANTUM VP OF DESTRUCTION

Assumptions:

- The prover wants to show to the verifier that he has measured (and thus irreversibly altered in their state) some quantum systems $p_1, \ldots, p_k$. These quantum systems act as witness objects in the VP.
- We implicitly assume that the prover has some technology at his disposal that allows him to store the quantum systems he receives, at least for the time frames that are relevant in the context of our VP.

Set-Up Phase:

The verifier prepares k polarized photons $p_1, \ldots, p_k$ in the following fashion (compare [4]):

- He fixes two orthogonal bases $\mathcal{B}_0, \mathcal{B}_1$, for example $\mathcal{B}_0 = \{0°, 90°\}$ and $\mathcal{B}_1 = \{45°, 135°\}$
- He chooses two tuples $B = (b_1, \ldots, b_k) \in \{0,1\}^k$ (the *"bases-tuple"*) and $V = (v_1, \ldots, v_k) \in \{0,1\}^k$ (the *"value-tuple"*) at random.
- For $i = 1, \ldots, k$, he encodes the value v_i in the basis B_{b_i} in the photon p_i. He does so by suitably polarizing the photon p_i in the basis B_{b_i}, as described in [4]. For example, if he wants to encode the value "1" in the basis $\mathcal{B}_1$, he polarizes the respective photon in an angle of $90°$.

The verifier sends the photons $p_1, \ldots, p_k$ to the prover.

Virtual Proof:

In order to allow the prover to show that he measures the photons (and thus irreversibly destroys their state), the prover and verifier jointly execute the following protocol:

1) The verifier chooses a tuple $T = (t_1, \ldots, t_k) \in \{0,1\}^k$ (the *"test-tuple"*) at random, and sends it to the prover.
2) For $i = 1, \ldots, k$, the prover measures the photon p_i in the basis B_{t_i}, and returns the measured value $r_i \in \{0,1\}$ to the verifier.

3) Let now $I \subseteq \{1, \ldots, k\}$ be the index set for which $t_i = b_i$. The verifier checks for all $i \in I$ that $r_i = v_i$. If this is the case, he accepts the VP, otherwise he aborts and rejects the VP.

Discussion: The above VP is described by the example of photons, but can be carried out by other quantum systems in an analog fashion. It allows the conclusion that the measurement has taken place in the time frame between step 1, in which the verifier sends away the bistring T, and step 2, in which the values r_i are returned to the verifier.

The VP's security directly follows from the security of the Bennett-Brassard key exchange protocol [4]. The role of the external adversary in Bennett-Brassard is played by the prover in our protocol: He cannot know or derive the values encoded in the photons without knowing the bases in which they were encoded. Any measurement without knowledge of the correct bases (for example, random measurements) will both lead to wrong measured values and to a notable disturbance of the state of the photons. For example, if the measurement of the photons had taken place before the VP (i.e., in the wrong bases), the values r_i would be altered and incorrect. This would be noticed by the verifier, who would reject the VP. Furthermore, if the prover tried to present the correct answers r_i without measurement, he would fail exactly for the same reasons as an external adversary fails to derive the exchanged key in the Bennett-Brassard protocol. Again, this would be noticed in step 3, and the verifier would reject the VP. In fact, the prover's chance of measuring l photons ahead of time without being caught decrease exponentially in l (compare [4]).

Regarding our assumption of a quantum memory, we remark this very assumption implicitly underlies many quantum protocols and quantum computing proposals, without diminishing the scientific reception of these proposals. It is currently under heavy research (see [6] and references therein). The time frame for which quantum storage is required depends very strongly on the application of our protocol, and should be revisited when real applications become a topic. Our aim in this paper is different: It lies on introducing VPs, and on plausibilizing that they can be realized by various technologies.

Let us have a final word on variants of the protocol. In principle, it would be possible that the prover chooses the "test-tuple" $T = (t_1, \ldots, t_k)$ by himself, and measures the quantum systems in the bases stipulated (by himself!) in the test tuple. This method saves one round of communication. However, it would only prove that the prover has measured the photons *before* a certain point in time. It would not allow the conclusion that the prover has executed the measurement *after* a certain point. In fact, he could have made the measurement a very long time ago, and simply kept the results. In this sense, Protocol 4 is a more exact method, one that allows a very close determination of the point of the measurement, i.e., of the destruction.

C. Possible Implementation

Regarding optical VPs of destruction, we again refer to the prototypical technology introduced by Pappu et al. [39], [40], which solves the positioning problem in the VP (compare Section IV-B). With respect to removing the outer PUF, we conjecture that this could easily be accomplished by employing to PUF materials with different chemical solubility, and by using one particular and dedicated solvent for the outer PUF. Also materials with different melting points could be utilized.

With respect to quantum VPs of destruction, the basic technology for polarizing, transmitting and measuring photons has been verified multiple times in experiments on quantum key exchange (QKE), even over very long distances, and requires no further justification in the course of this paper [32]. Only the storage of quantum system is currently under heavy research [6]. We stress, however, that such technology questions are not — and cannot be — the main topic of this paper. As quantum technology advances, so will the possibility for practical implementations of our VPs. This can be observed well by the example of QKE, which step by step got closer to practical implementations, now being a commercially available product [28].

VI. Public Virtual Proofs

Private VPs have two potential downsides: Firstly, they require a secure set-up phase and the storage of private information at the verifier. Secondly, they involve a physical transfer of the PUF to the prover. Thirdly, some of the used WOs contain information whose exposure to an adversary compromises the security of the scheme. This security critical information is not present in the form of a classical, digital key stored in NVM, but it still exists. For example, if an adversary knows the exact parameters which influence the temperature-dependent CRP behavior of the BR PUF, he may be able to numerically simulate its CRP behavior, without the WO/BR PUF being at the claimed temperature. This allows cheating in a VP of temperature. Similar considerations may hold for some other electric WOs. [5]

Whether the above aspects are considered serious disadvantages depends on the exact application scenario of the VP. In a typical client-server setting, for example bank cards/ATMs/bank headquarters, the secure set-up phases and private information storage at the verifier tyically pose no problems. On the other hand, if a VP shall be executed

[5] At the same time, they do not hold for all WOs, not even in private VPs. As an example, consider optical WOs and their uses in various VPs: Even if the adversary knows the exact position of all scatterers, he will not be able to exactly simulate the CRP behavior, simply for reasons of computational complexity.

between arbitrary parties in a www-like communication infrastructure, they do. Likewise, the security critical information in the PUF can be no problem in certain settings, but may be relevant in others. This motivates an investigation whether the above three aspects could be overcome by an alternative approach named *"public virtual proofs"* (public VPs). Our aim in this section is to merely sketch public VPs, and to motivate further research in this direction. We do not present working implementations, nor do we execute experimental proofs of concept.

The basic idea behind public VPs is to use WOs that are similar to SIMPL systems [44], [50], [13], [46], [48] or public PUFs [2], [37], [41]. SIMPL systems are PUF-like structures, which are also unclonable and exhibit a certain challenge-response behavior. Furthermore, similar to Strong PUFs, their CRP interface is publicly accessible, and they possess so many CRPs that an adversary cannot read out all of them. In opposition to Strong PUFs, however, SIMPLs have a public simulation algorithm by which everyone, including the adversary, can simulate their challenge response behavior. The simulation comes at a time loss, though: It shall be impossible for the adversary to determine the response to a randomly chosen challenge *by simulation* as quickly as one could determine this response *by actual measurement* on the SIMPL system. It shall not be possible to optimize the public simulation code of the SIMPL, or to run in on a quick computer, in order to bring down the simulation time to the response speed of the real, physical SIMPL system. This means that a party holding this unique SIMPL system can always be distinguished from a party merely simulating the responses, simply by measuring and comparing their response times. Furthermore, it is part of the attack model on SIMPLs that the adversary could learn all details about their internal configuration without compromising security. This point has been elaborated in all detail in [46].

Given these features of SIMPL systems, it seems tempting to construct SIMPL-like WOs and use them in public VPs. Let us illustrate our idea by virtue of an example, namely potential *public VPs of temperature*. We stress that our protocol is yet hypothetic, since implementations of temperature-sensitive SIMPLs have not yet been published.

Protocol 5: PUBLIC ELECTRICAL VPS OF TEMPERATURE

Assumptions:

- The prover wants to show the temperature of one specific witness object, in this an electrical SIMPL system, within a temperature range $\mathbf{R_T}$. This temperature range is discretized, possessing k discrete temperature levels $t_1, \ldots, t_k \in \mathbf{R_T}$.
- The SIMPL is temperature dependent. Its responses R_j^i are a function not only of the applied challenges C_j, but also of the current temperature $t_i \in \mathbf{R_T}$ of the SIMPL: $R_j^i = F_{\mathsf{SIMPL}}(C_j, t_i)$.
- As for any SIMPL system, there is a numerical simulation algorithm Sim by which the responses can be simulated accurately: For challenges C_j and temperatures $t_i \in \mathbf{R_T}$, it shall hold that $R_j^i = \mathsf{Sim}(C_j, t_i)$.
- Numerically simulating a response R_i^j shall be notably slower than physically measuring it. Concretely, we assume for any temperature level t_i and for a randomly chosen challenge C_j:
  - No adversary can present the correct responses R_i^j quicker than in time T_{Adv}, even if he knows (or has simulated) many other CRPs of the SIMPL system.
  - The honest users can simulate the response R_i^j on their hardware in time T_{UserSim}.
  - The honest users can obtain the response by physical measurement on the SIMPL in time T_{UserMeas}.
  - It shall hold that $T_{\mathsf{UserSim}} > T_{\mathsf{Adv}} > T_{\mathsf{UserMeas}}$.
- We assume that the occuring communication delays between the prover and the verifier are small compared to the difference $T_{\mathsf{Adv}} - T_{\mathsf{UserMeas}}$.

Public Virtual Proof:

1) The prover claims to the verifier that the SIMPL is at a temperature $T \in \{t_1, \ldots, t_k\}$.
2) For $v = 1, \ldots, n$, the verifier randomly selects a challenge C_v.
3) For $v = 1, \ldots, n$, the verifier and the prover execute the following procedure:
   - The verifier sends the challenge C_v to the prover.
   - The prover applies the challenge C_v to the SIMPL, and immediately sends the measured response of the SIMPL to the verifier.
   - The verifier stores the obtained response R_v. He checks if the time that passed between sending away C_v and receiving R_v is smaller than T_{Adv}. If not, he aborts.
4) For $v = 1, \ldots, n$, the verifier checks by simulation if $R_v = \mathsf{Sim}(C_j, t_i)$. If not, he aborts.

Discussion: The above protocol requires no private set-up phase at the verifier and no CRP-lists. It thus enables VPs between arbitrary parties, which is a huge asset in www-like communication scenarios. Furthermore, if standard, temperature-sensitive SIMPL systems are used, then no security-critical information at all is present on the side of the prover. For a detailed discussion on this topic, we refer the reader to [46].

We stress that the practically efficient implementation of SIMPL systems is currently an ongoing research topic. In particular, no temperature sensitive SIMPLs have been identified up to this point. Our main motivation to include the above example was to show that VPs offer potential

beyond private VPs. Whether this potential can be unleashed in practice, and whether the associated problems can all be overcome, remains subject to future research; it is a typical high risk, high impact situation. Still, we felt that the paper would be incomplete without public VPs.

VII. Summary

We introduced a new security concept in this paper, so-called *"virtual proofs of reality"* (VPs). Figure 4 illustrates their idea: Physical systems or processes shall be converted into digital data in a way that enables a later proof that the digital data is "correct" and "authentic", i.e., that it adequately describes some features of a really existing physical system or process. The prover may be assisted by so-called *"witness objects"* (WOs) in the proof, which help him transforming physical reality into digital data in a provably correct manner. The prover's system shall not contain any classical secret keys; in particular, the WOs shall not contain such keys, nor be assumed tamper resistant in the usual sense. The use of WOs is not as restrictive as it may seem at first glance: Some electronic device will necessarily have to be used by the prover to transfer digital data to the verifier. Instead of an arbitrary device, the prover may as well employ witness objects.

In this very first paper on the topic, our aim was not — and could not have been — to realize every conceivable aspect of VPs. Rather, our focus was on introducing the novel concept, describing several example protocols, and plausibilizing that these protocols can be implemented with existing techniques. Our treatment covered different technologies, including electrical PUFs, optical PUFs, and quantum systems.

Our first VP described how to prove the temperature of a witness object in the prover's system under the above circumstances. It could be based, so we argued, on special electrical PUFs that have a high fraction of temperature sensitive CRPs (like the Bistable Ring PUF [11], [12]). While these CRPs are unwanted in classical PUF applications, they serve well in our context and enable VPs of temperature. A second possibility is the use of more stable electrical PUFs, which are operated at their stability limit, for example XOR Arbiter PUFs with 8 or 16 XORs [67], [58], [60]. The influences of voltage fluctuations could be eradicated by multiple measurements and majority voting at one fixed temperature, similar to the process carried out in [60] for silicon modeling attacks with very high exactness.

Our second protocol dealt with VPs of distance, and proves the mutual distance of two optical witness objects reminiscent of Pappu's optical PUF. The joint optical speckle pattern which these two objects generate strongly depends on their relative position. Pappu et al. [39], [40] describe in their work how one optical scattering system can be repeatedly positioned inside a measurement apparatus such that approximately the same speckle pattern occurs. The same technology could be used in our case, assuming that one of the scattering tokens is fixed inside the measurement apparatus, and the other token is movable and is being repositioned.

Finally, we turned to the most complicated VPs treated in this paper, VPs of destruction. Their goal is to show that a certain physical object has been destroyed or irreversibly modified. These proofs seems counterintuitive at first sight; one necessary assumption seems that the remaining object after destruction or modification is not atomized or entirely reduced to ashes. In this sense, it seems likely that VPs of destruction can only be led for for certain objects and under certain circumstances. For example, it must be possible to establish an unambigious link between the state of the object before destruction and after destruction.

We suggested two different protocols for VPs of destruction. Our first technique utilizes the idea of placing a PUF inside a PUF. Two optical PUFs à la Pappu et al. [40], [39] are used to form a composed PUF. The structure is designed in such a way that a removal of its outer layer (i.e., its outer PUF) is irreversible. In the fabrication process, some CRPs of the inner PUF structure can be measured before the outer PUF is added. Once the outer PUF has been removed, these CRPs can be used to re-identify the inner structure; but as long as the outer structure is present, the CRPs of the inner PUF obviously cannot be measured. This allows a VP of destruction, in which even the point in time when the outer PUF has been removed can be determined with quite high exactness. Again, a relization seems plausible by existing technology, in this case the techniques from earlier works on optical PUFs [39], [40]. These techniques can be combined with materials that have different solubility or melting points, which should enable an easy removal of the outer PUF.

A second realization was based on quantum systems. We made use of the established principles of Bennett-Brassard quantum key exchange (QKE) [4], but with a small twist: The prover illustrates his measurement (and the destruction of the state of the quantum systems) to the prover by showing that he has correctly learned some of the secrets stored in the quantum systems. A cheating prover would be in a similar situation as an external eavesdropper in Bennett-Brassard, so we argued; therefore the security of our protocol is directly inherited from the Bennett-Brassard protocol.

Towards the end of the paper, we described in theory one potential extension of our method, namely so-called public VPs. Their main potential advantages over private VPs are that they function without secure set-up phases at the verifier, without a physical transfer of witness objects from the verifier to the prover prior to the proof, and potentially also allow that the prover himself (or another third party not trusted by the verifier) fabricates the witness objects. In this sense, the advantages of public VPs over private VPs are reminiscent of the upsides of public key over private key

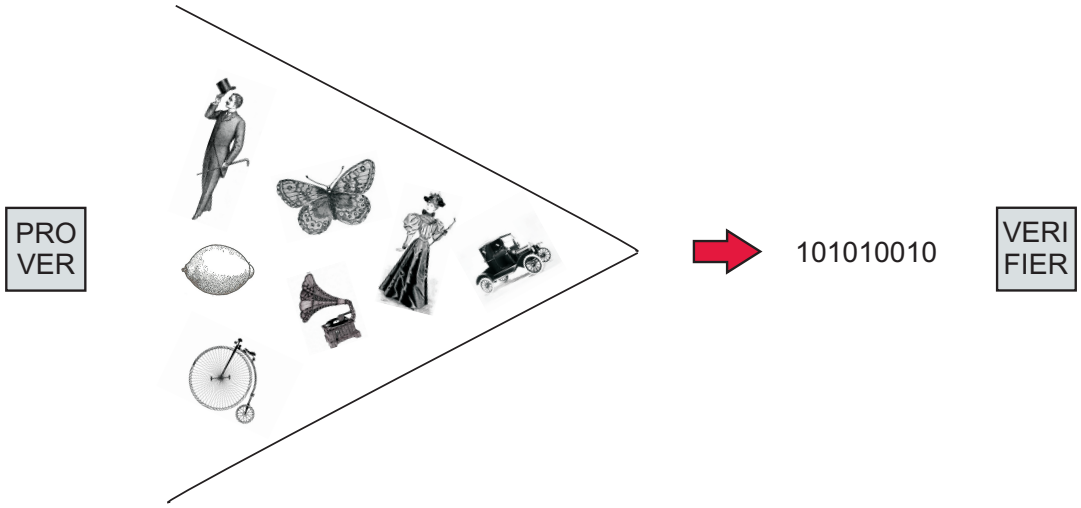

Figure 4. The idea behind virtual proofs in its most general form: Complex physical systems are converted into digital data in a way that allows proving that the digital data is "correct" and "authentic", i.e., that it corresponds to a real, actual physical system or physical process with the claimed properties. This conversion is accomplished via so-called *"witness objects"* (red arrow), without using classical secret keys or tamper proof hardware.

cryptography. We sketched one public VP of temperature in order to illustrate our idea and to complete the paper.

Applications: Our main focus in this work were not potential practical applications of virtual proofs. Rather, we focused on the introduction and plausibilization of this novel security concept. Still, several such applications lie at hand, and we will briefly discuss them below for completeness.

To start with, VPs of distance can be used to prove that one or more objects were at a particular place in a particular point in time. Conceivable applications lie in the context of bank cards and automated teller machines (ATMs); missile tagging or weapons inspections; or in any situation where a joint authentication by two different items is necessary, one classical example being the two security tokens of the US president and vice president required to launch an atomic weapon. Another application of the joint interference patterns arising from two different scattering objects could be (i) encryption and decryption schemes that depend on the cooperation of two parties holding the two objects, or (ii) location-dependent encryption and decryption, especially in the case that one of the scattering objects has been immobilized and is bound to one particular place (for example a terminal or ATM).

VPs of destruction, on the other hand, have immediate applications to the digital rights management problem: The rights to play a certain content may be linked to the existence of a certain object; if the customer no longer wants to maintain these rights (and no longer wants to pay for them), the object is destroyed by the customer. He could prove that he did so to the company granting the rights. Another appliance of VPs of destruction could lie in the field of secure data deletion. Finally, VPs of temperature and, more generally, sensor data appear to have straightforward applications to the security sector, for example in sensor networks. Many other examples are conceivable, and are left to the readers.

Future Work: An obvious next step is the full experimental implementation of our concepts, and the exploration of associated practical questions, such as: Which temperature or distance resolutions can be achieved in practical implementations? How can these resolutions be optimized? How can we, for example, design Bistable Ring PUFs (or other electrical PUFs) with a particularly high temperature sensitivity, which would allow very finegrained VPs of temperature? A related topic is the design of entirely new VPs. Which other types of VPs exist, and how would the corresponding witness objects have to look like? For example: Can disordered, unclonable cameras be designed in order to prove the authenticity of pictures or movies via virtual proofs?

A second possible strand of future research concerns the logic and computational complexity aspects behind our new concept. Is there a "universal" VP, to which any other VP can be reduced, similar to the existence of universal Turing machines? Is there a "hierarchy" of physical statements that can be proven by VPs with different computational resources, communication complexities, or numbers of witness objects? Given the relationship of VPs to interactive proof systems, it seems natural to consider such issues. An extension of the Turing machine model, some sort of "physical Turing machines", could be necessary to address them with full formal rigor. Some first steps to this end have already been made in [47].

Interestingly, the above theoretical questions may not be resolvable by mathematics alone, but overlap with physics. They could be seen in alignment with recent efforts of

linking information theory, physics and computation. Among the many works relevant to this emerging area, we would like to exemplarily mention the arguments of G. 't Hooft [27], L. Susskind [68], R. Bousso [7], [8] and others on the limited entropy or information storage capacity of physical systems (see Bekenstein [3] for an easily accessible summary). These works originated in the area of physics, but also have immediate consequences for the areas of physical compuation and information theory. For example, the authors of [27], [68], [7], [8] establish a theoretical upper bound on the information that can be stored in a given spatial volume, i.e., they show that it is impossibile to store more information in bits in a physical system than given by a low-degree bound in the system's volume. It seems highly interesting to extend these *physical* impossibility arguments to further *computational* and *information-theoretic* questions. VPs and the open theoretical questions associated with them seem to naturally fit into this emerging area.

ACKNOWLEDGEMENTS

We thank Dima Kononchuk and Peter van Emde-Boas for suggesting the concept of using a "PUF inside a PUF" in our VPs of Destruction after a talk of the author in the RISC seminar at CWI, Amsterdam, in 2012. We are also grateful to Stefan Wolf for proposing the general idea of using quantum technology in our VPs of Destruction after a talk at SOFSEM 2011.

REFERENCES

[1] R. J. Anderson: *Security Engineering: A guide to building dependable distributed systems*. Wiley, 2010.

[2] N. Beckmann, M. Potkonjak: *Hardware-Based Public-Key Cryptography with Public Physically Unclonable Functions*. Information Hiding 2009.

[3] J.D. Bekenstein: *How does the entropy/information bound work?* Foundations of Physics 35.11 (2005): 1805-1823.

[4] C.H. Bennett, G. Brassard: *Quantum cryptography: Public key distribution and coin tossing.* IEEE International Conference on Computers, Systems and Signal Processing, 1984.

[5] Buhrman, H., Chandran, N., Fehr, S., Gelles, R., Goyal, V., Ostrovsky, R., and Schaffner, C. *Position-based quantum cryptography: Impossibility and constructions*. CRYPTO 2011, pp. 429-446, Springer.

[6] Khodjasteh, K., Sastrawan, J., Hayes, D., Green, T. J., Biercuk, M. J., and Viola, L. Designing a practical high-fidelity long-time quantum memory. Nature Communications, 4, 2013

[7] R. Bousso: *A covariant entropy conjecture.* Journal of High Energy Physics 1999.07 (1999): 004.

[8] R. Bousso: *The holographic principle.* Reviews of Modern Physics 74.3 (2002): 825.

[9] C. Brzuska, M. Fischlin, H. Schröder, S. Katzenbeisser: *Physical Unclonable Functions in the Universal Composition Framework.* CRYPTO 2011.

[10] J. Buchanan, R. Cowburn, A. Jausovec, D. Petit, P. Seem, G. Xiong, D. Atkinson, K. Fenton, D. Allwood, and M. Bryan: *Forgery: Fingerprinting documents and packaging.* Nature, vol. 436, no. 7050, p. 475, 2005.

[11] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, U. Rührmair: *The Bistable Ring PUF: A New Architecture for Strong Physical Unclonable Functions*. HOST 2011.

[12] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, U. Rührmair: *Characterization of the Bistable Ring PUF*. DATE 2012.

[13] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, M. Stutzmann, U. Rührmair: *Circuit-based Approaches to SIMPL Systems*. Journal of Circuits, Systems and Computers, 2011.

[14] G. Csaba, X. Ju, Z. Ma, Q. Chen, W. Porod, J. Schmidhuber, U. Schlichtmann, P. Lugli, U. Rührmair: *Application of Mismatched Cellular Nonlinear Networks for Physical Cryptography*. IEEE CNNA - 12th International Workshop on Cellular Nonlinear Networks and their Applications, 2010.

[15] M. van Dijk, U. Rührmair: *Physical Unclonable Functions in Cryptographic Protocols: Security Proofs and Impossibility Results.* Cryptology ePrint Archive, Report 228/2012, 2012.

[16] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, M. T. Manzuri Shalmani: *On the Power of Power Analysis in the Real World: A Complete Break of the KeeLoqCode Hopping Scheme*. CRYPTO 2008, pp. 203-220, 2008.

[17] W. Clarkson, T. Weyrich, A. Finkelstein, N. Heninger, J. Halderman, E. Felten: *Fingerprinting blank paper using commodity scanners.* IEEE Symposium on Security and Privacy (Oakland'09), pp. 301-314, 2009.

[18] B. Gassend, *Physical Random Functions*, MSc Thesis, MIT, 2003.

[19] B. Gassend, D. E. Clarke, M. van Dijk, S. Devadas: *Silicon physical random functions*. ACM Conference on Computer and Communications Security 2002, pp. 148-160, 2002

[20] B. Gassend, D. Lim, D. Clarke, M. van Dijk, S. Devadas: *Identification and authentication of integrated circuits.* Concurrency and Computation: Practice & Experience, Vol. 16(11), pp. 1077 - 1098, 2004.

[21] O. Goldreich, S. Micali, A. Wigderson: *Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems.* Journal of the ACM (JACM) 38.3 (1991): 690-728.

[22] S. Goldwasser, S. Micali, C. Rackoff: *The knowledge complexity of interactive proof systems.* SIAM Journal on computing 18.1 (1989): 186-208.

[23] J. Guajardo, S. S. Kumar, G. J. Schrijen, P. Tuyls: *FPGA Intrinsic PUFs and Their Use for IP Protection*. CHES 2007: 63-80

[24] G. Hammouri, A. Dana, B. Sunar: *CDs have fingerprints too.* CHES 2009: 348-362.

[25] C. Hilgers: *Praktische Realisierung von Verfahren aus der Physikalischen Kryptographie*. Master Thesis, Technische Universität München, 2009.

[26] D. E. Holcomb, W. P. Burleson, K. Fu: *Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Numbers*. IEEE Trans. Computers 58(9): 1198-1210, 2009.

[27] G. 't Hooft: *Dimensional reduction in quantum gravity.* Arxiv preprint gr-qc/9310026, 1993.

[28] http://swissquantum.idquantique.com/?-Quantum-Cryptography-

[29] C. Jaeger, M. Algasiner, U. Rührmair, G. Csaba, M. Stutzmann: *Random pn-junctions for physical cryptography*. Applied Physics Letter 96, 172103, 2010.

[30] T. Kasper, M. Silbermann, C. Paar: *All You Can Eat or Breaking a Real-World Contactless Payment System*. Financial Cryptography 2010, pp. 343-350, 2010.

[31] S. S. Kumar, J. Guajardo, R. Maes, G. J. Schrijen, P. Tuyls: *The Butterfly PUF: Protecting IP on every FPGA*. HOST 2008: 67-70

[32] Kurtsiefer, C., Zarda, P., Halder, M., Weinfurter, H., Gorman, P. M., Tapster, P. R., and Rarity, J. G.. Quantum cryptography: A step towards global key distribution. Nature, 419(6906), 450-450.

[33] J.-W. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas. *A technique to build a secret key in integrated circuits with identification and authentication applications.* In Proceedings of the IEEE VLSI Circuits Symposium, June 2004.

[34] M. Majzoobi, A. Elnably, F. Koushanfar: *FPGA time-bounded unclonable authentication.* Information Hiding 2010.

[35] M. Majzoobi, F. Koushanfar: *Time-bounded authentication of FPGAs.* IEEE Transactions on Information Forensics and Security, 2011.

[36] M. Majzoobi, F. Koushanfar, M. Potkonjak: *Lightweight Secure PUFs.* IC-CAD 2008: 607-673.

[37] S. Meguerdichian, M. Potkonjak: *Matched public PUF: ultra low energy security platform*. ISLPED 2011: 45-50

[38] R. Ostrovsky, A. Scafuro, I. Visconti, A. Wadia: *Universally Composable Secure Computation with (Malicious) Physically Uncloneable Functions*. IACR Cryptology ePrint Archive 2012:143, 2012.

[39] R. Pappu: *Physical One-Way Functions*. PhD Thesis, Massachusetts Institute of Technology, 2001.

[40] R. Pappu, B. Recht, J. Taylor, N. Gershenfeld: *Physical One-Way Functions*, Science, vol. 297, pp. 2026-2030, 20 September 2002.

[41] M. Potkonjak, S. Meguerdichian, A. Nahapetian, S. Wei: *Differential public physically unclonable functions: architecture and applications*. DAC 2011: 242-247

[42] R. Rivest: *Illegitimi non carborundum*. Invited keynote talk, CRYPTO 2011.

[43] K. Rosenfeld, E. Gavas, R. Karri: *Sensor Physical Unclonable Functions*. HOST 2010, pp. 112-117, 2010.

[44] U. Rührmair: *SIMPL Systems: On a Public Key Variant of Physical Unclonable Functions*. Cryptology ePrint Archive, Report 2009/255, 2009.

[45] U. Rührmair: *Oblivious Transfer based on Physical Unclonable Functions (Extended Abstract)*. TRUST 2010.

[46] U. Rührmair: *SIMPL Systems, Or: Can We Design Cryptographic Hardware without Secret Key Information?* SOFSEM 2011.

[47] U. Rührmair: *Physical Turing Machines and the Formalization of Physical Cryptography*. Cryptology ePrint Archive, Report 2011/188, 2011.

[48] U. Rührmair: *SIMPL Systems as a Keyless Cryptographic and Security Primitive*. In: D. Naccache (Ed.), Festschrift of J.-J. Quisquater, LNCS Vol. 6805, Springer, 2012.

[49] U. Rührmair, H. Busch, S. Katzenbeisser: *Strong PUFs: Models, Constructions and Security Proofs.* In A.-R. Sadeghi, P. Tuyls (Editors): Towards Hardware Intrinsic Security: Foundation and Practice. Springer, 2010.

[50] U. Rührmair, Q. Chen, M. Stutzmann, P. Lugli, U. Schlichtmann, G. Csaba: *Towards Electrical, Integrated Implementations of SIMPL Systems*. WISTP 2010.

[51] U. Rührmair, C. Jaeger, M. Bator, M. Stutzmann, P. Lugli, and G. Csaba: *Applications of high-capacity crossbar memories in cryptography*. IEEE Transactions on Nanotechnology, 2011.

[52] U. Rührmair, M. van Dijk: *Practical Security Analysis of PUF-based Two-Player Protocols*. CHES 2012.

[53] U. Rührmair, M. van Dijk: *On the Practical Use of Physical Unclonable Functions in Oblivious Transfer and Bit Commitment Protocols*. Journal of Cryptographic Engineering, 2013.

[54] U. Rührmair, M. van Dijk: *PUFs in Security Protocols: Attack Models and Security Evaluations*. IEEE Symposium on Security and Privacy (Oakland'13), 2013.

[55] U. Rührmair, C. Hilgers, S. Urban, A. Weiershäuser, E. Dinter, B. Forster, C. Jirauschek: *Optical PUFs Reloaded*. IACR Cryptology ePrint Archive, Report 2013/215, 2013.

[56] U. Rührmair, C. Jaeger, C. Hilgers, M. Algasinger, G. Csaba, and M. Stutzmann: *Security applications of diodes with unique current-voltage characteristics.* Financial Cryptography 2010, 2010.

[57] U. Rührmair, S. Devadas, F. Koushanfar: *Security based on Physical Unclonability and Disorder.* In M. Tehranipoor and C. Wang (Editors): Introduction to Hardware Security and Trust. Springer, 2011

[58] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, J. Schmidhuber: *Modeling Attacks on Physical Unclonable Functions*. ACM Conference on Computer and Communications Security, 2010.

[59] U. Rührmair, J. Sölter, F. Sehnke: *On the Foundations of Physical Unclonable Functions.* IACR Cryptology ePrint Archive, Report 2009/277, 2009.

[60] U. Rührmair, J. Sölter, F. Sehnke, X. Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Burleson, S. Devadas: PUF Modeling Attacks on Simulated and Silicon Data. IEEE T-IFS, 2013.

[61] U. Rührmair, M. Stutzmann, G. Csaba, U. Schlichtmann, P. Lugli: *Method for Security Purposes*. Patent Application, WO2010105994 A2, PCT/EP2010/053234. Publication Date: 2010. Priority Date (i.e., first submission of patent): 2009. See http://www.google.com/patents/WO2010105994A2?cl=en

[62] U. Rührmair: *Physical Cryptography, Or: How to Realize Cryptography and Security by Random, Disordered, and Unclonable Physical Structures*. Invited talk, SOFSEM 2011. See http://kedrigern.dcs.fmph.uniba.sk/kralovic/sofsem2011/index.php?param=204

[63] U. Rührmair: *Cryptographic Protocols based on Physical Unclonable Functions and Related Structures*. Invited talk, RISC Seminar, CWI, Amsterdam. See http://projects.cwi.nl/crypto/risc2012.html

[64] P. Simons, E. van der Sluis, V. van der Leest: *Buskeeper PUFs, a promising alternative to D Flip-Flop PUFs.* HOST 2012: 7-12.

[65] A. Sharma, L. Subramanian, E. A. Brewer: *PaperSpeckle: microscopic fingerprinting of paper.* Proceedings of the 18th ACM conference on Computer and communications security (CCS'18), 2011.

[66] J. R. Smith, A. V. Sutherland: *Microstructure-Based Indicia.* Proceedings of the Second Workshop on Automatic Identification Advanced Technologies, Morristown, NJ, pp. 79-83, 1999.

[67] G. E. Suh, S. Devadas: *Physical Unclonable Functions for Device Authentication and Secret Key Generation.* DAC 2007: 9-14

[68] L. Susskind: *The World as a Hologram.* J. Math. Phys. 36, pp. 6377-6396, 1995.

[69] P. Tuyls, G. J. Schrijen, B. Skoric, J. van Geloven, N. Verhaegh, R. Wolters *Read-Proof Hardware from Protective Coatings.* CHES 2006: 369-383.