SCIENCE & GLOBAL SECURITY
2019, VOL. 27, NO. 1, 3–14
https://doi.org/10.1080/08929882.2019.1573483

Routledge
Taylor & Francis Group

A Cryptographic Escrow for Treaty Declarations and Step-by-Step Verification

Sébastien Philippe [a,b], Alexander Glaser [c], and Edward W. Felten[d]

[a]International Security Program and Project on Managing the Atom, Belfer Center for Science and International Affairs, John F. Kennedy School of Government, Harvard University, Cambridge, MA, USA; [b]Nuclear Knowledge Program, Center for International Studies (CERI), Sciences-Po, Paris, France; [c]Program on Science and Global Security, Princeton University, Princeton, NJ, USA; [d]Center for Information Technology Policy, Princeton University, Princeton, NJ, USA

ABSTRACT

The verification of arms-control and disarmament agreements requires states to provide declarations, including information on sensitive military sites and assets. There are important cases, however, in which negotiations of these agreements are impeded because states are reluctant to provide any such data, because of concerns about prematurely handing over militarily significant information. To address this challenge, we present a cryptographic escrow that allows a state to make a complete declaration of sites and assets at the outset and commit to its content, but only reveal the sensitive information therein sequentially. Combined with an inspection regime, our escrow allows for step-by-step verification of the correctness and completeness of the initial declaration so that the information release and inspections keep pace with parallel diplomatic and political processes. We apply this approach to the possible denuclearization of North Korea. Such approach can be applied, however, to any agreement requiring the sharing of sensitive information.

ARTICLE HISTORY
Received 21 September 2018
Accepted 25 November 2018

Introduction

Ever since the Strategic Arms Limitation Talks between the United States and the Soviet Union, nuclear arms-control treaties have included transparency measures and the exchange of information.[1] Negotiating deeper cuts in U.S. and Russian nuclear arsenals would require unprecedented disclosures, however. In 1997, a National Academy of Sciences study proposed transparency measures such as,

> "the current location, type, and status of all nuclear explosive devices and the history of every nuclear explosive device manufactured, including the dates of assembly and dismantling or destruction in explosive tests; a description of facilities at which nuclear explosives have been designed, assembled, tested, stored, deployed, maintained, and

CONTACT Sébastien Philippe sebastien_philippe@hks.harvard.edu Belfer Center for Science and International Affairs, Harvard Kennedy School, 79 John F. Kennedy St, Mailbox 134, Cambridge, MA, 02138, USA

dismantled, and which produced or fabricated key weapon components and nuclear materials; and the relevant operating records of these facilities."[2]

Such disclosures are difficult to undertake, because they could provide an adversary with militarily significant information at an early stage. In a 2005 study, the National Academy of Sciences' Committee on International Security & Arms Control suggested that cryptography could help address this problem.[3]

A similar challenge is arising today in the context of United States–Democratic People's Republic of Korea (DPRK) talks on the denuclearization of the Korean Peninsula because denuclearization shares many of the problems associated with deeper nuclear reductions in an acute way. As part of an agreement, the DPRK would likely be required to provide data and disclose activities related to its nuclear and ballistic-missile programs, as well as submit to observation and onsite inspections by the international community.[4]

Prior verification plans proposed by the United States asked the DPRK to make substantial and detailed baseline declarations including: the current location, type, and status of all nuclear weapons and associated components; a description of facilities at which nuclear materials and weapons have been produced, designed, assembled, tested, stored, and deployed; and data on the quantities and characteristics of declared nuclear material.[5] From the DPRK point of view, agreeing to such demands may be too risky; it would provide the United States with a potentially comprehensive map of its military and nuclear weapons-related assets at a very early stage in the diplomatic process, which could become an important security threat if negotiations collapsed. But given the strong U.S. public commitment to verifiable denuclearization, it is difficult to conceive a successful diplomatic outcome in which the DPRK would not provide any kind of useful declaration.[6]

Here, we address this negotiation challenge by presenting an approach to declarations that provides a secure information-sharing mechanism for a state to sequentially reveal relevant sensitive information to another state while requiring the country making declarations to commit to the correctness and completeness of their initial declaration at the outset, potentially even before negotiations start. This cryptographic escrow scheme enables the release of partial information for verification at later stages, as opposed to engaging in the full disclosure of all data at once (Figure 1). This allows data exchanges to keep pace with confidence-building measures.

Our escrow leverages cryptographic primitives in particular commitment schemes.[7] Such schemes allow a party to commit to a particular piece of information, or value, while keeping it hidden from others. The committing party can release the value at a later stage while ensuring other parties it was not altered.

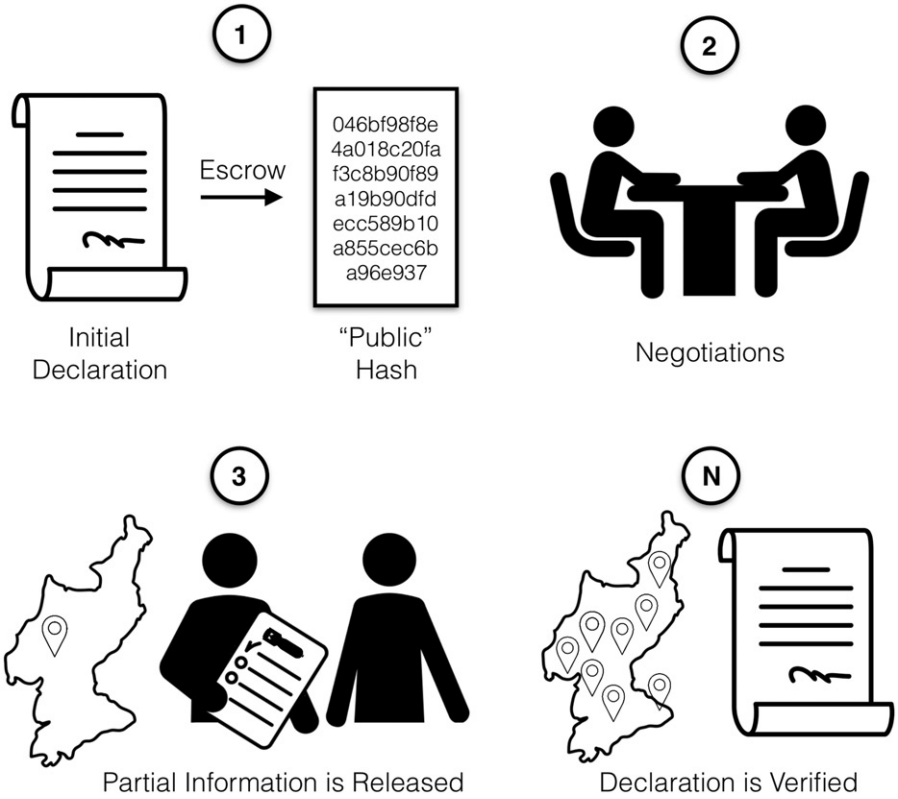

Figure 1. Using a cryptographic escrow in an inspection regime. (1) A detailed initial declaration is produced by the inspected party and placed in an escrow. A cryptographic commitment to this declaration is made public. (2) The negotiations are ongoing. The escrow is built such that it is possible to reveal only partial information at a time. (3) Prior to an on-site inspection, partial information about a site (location, status, and items) is revealed to the inspecting party. The inspections eventually confirm the correctness of this information. (N) As negotiations move forward, information is released incrementally until the complete declaration is revealed. Only then does the inspecting party have a complete picture of the inspected party assets.

While verifying the denuclearization of North Korea is a particularly relevant application for our approach, similar escrow scheme could be used in other international agreements including the exchange of secure declaration as part of future U.S.–Russia arms-reduction efforts,[3] or the declaration of sensitive information (e.g., identification and location of pollution emitters) in environmental agreements.[8]

Escrow construction

The most basic construction for our escrow could be a cryptographic commitment of the entire declaration. One way to implement a commitment scheme is through the application of cryptographic hash functions. In

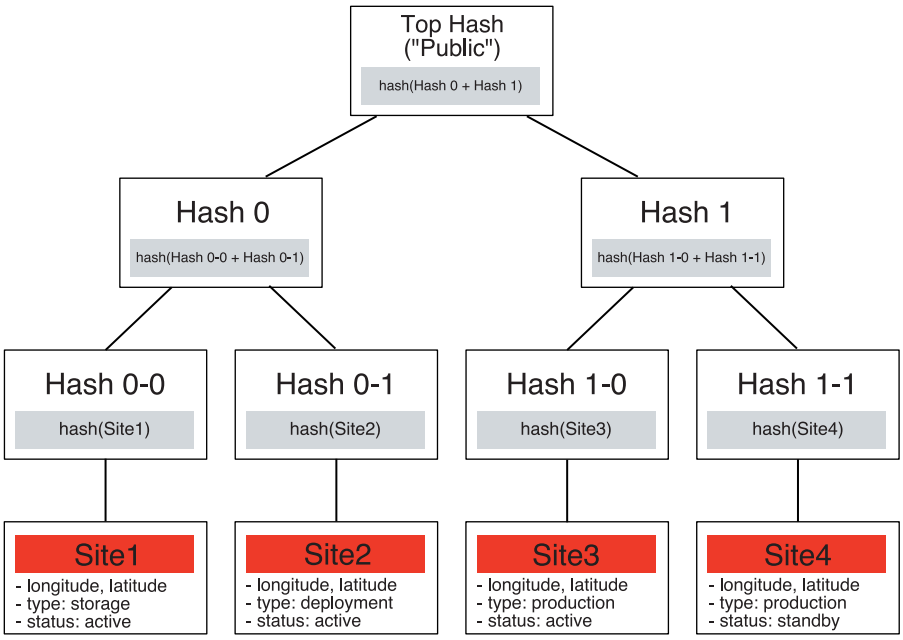

Figure 2. Sites declaration using a Merkle Tree structure. Each leaf of the tree contains information on individual sites (referred as Site 1–4). The information of each data block is hashed individually, the upper nodes hashes "0" and "1" are obtained by hashing the concatenation of the two lower hashes up to the Top Hash, also called the root of the tree. To later demonstrate that the information of Site 4 was part of the declaration, the prover needs to supply the clear text of Site 4, the hash of Site 4 (Hash 1–1), the hash 1–0, the Hash 1 and the Top Hash. The process does not reveal information about any other sites.

general, the hash of a message is much shorter than the message itself, and the underlying cryptographic hash function is designed such that it is infeasible to find a valid message for a given hash (assuming the values being hashed are drawn from a random distribution with high entropy) and infeasible to construct two different messages that produce the same hash (a property called collision resistance). In principle, multiple hash functions can be combined, using robust multi-property combiners, so that each of the necessary cryptographic properties holds for the combination if it holds for at least one of the hash functions being combined.[9] This could be used, for example, to allow each party in our scheme to propose a hash function he or she trusts, and to use a combined hash function that has the desired security if either one of the parties' chosen hash functions is secure.

Simply committing to the complete declaration would not provide for any flexibility on how much and what information can be revealed at a time, however. To address this issue, we turn our escrow into a binary Merkle tree (see Figure 2).[10] The tree is constructed as follows: every leaf (or childless node) can store any string (defined as a finite sequence of characters), for example, a cryptographic commitment to a data block with

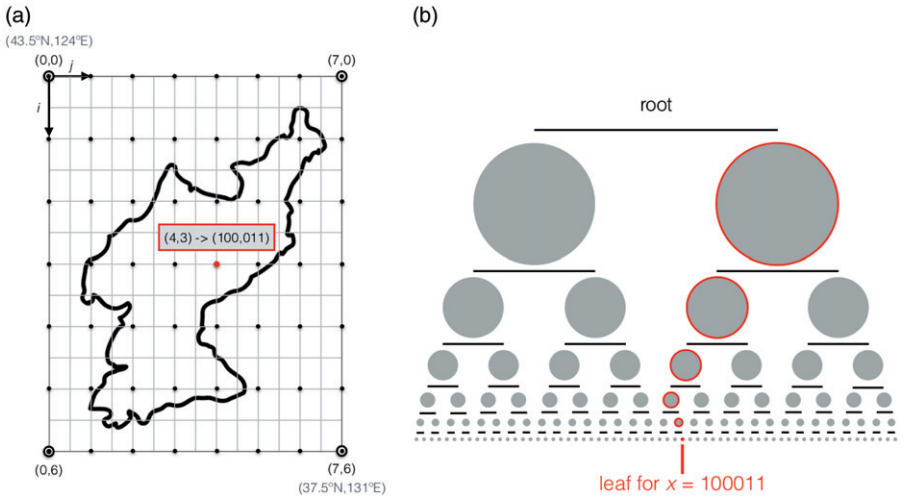

Figure 3. Mapping coordinates of a site to a Merkle tree leaf. Local coordinates (*i,j*) placed over the DPRK (A) can identify uniquely each location in the country with a given precision. The numbers *i* and *j* are then converted in their binary equivalent and concatenated into a string *x* corresponding to the binary path from the root to the corresponding leaf in our Merkle tree (B). Each leaf of the tree either stores a commitment to information about an existing site or to nothing if no site is present at this location.

information related to a specific site, including, in the case of North Korea, the denuclearization-relevant items stored at the site. Any non-leaf node must store the value *Hash*(*L*,*R*) whenever its left child stores *L* and right child stores *R*. The root of the tree then represents a commitment to the entire declaration and would be the only piece of information made public at the beginning of the diplomatic process.

Furthermore, we build the tree such that a pair of geographic coordinates in the country corresponds to a unique leaf in the tree. To do so, we superpose a grid over the map of the country (see Figure 3). For North Korea, the grid is bounded by latitudes 37.5° N and 43.5° N, and longitudes 124.0° E and 131.0° E. We then construct a local coordinate system (*i,j*) with the point (43.5° N, 124.0° E) as the origin. The number of grid points depends on the chosen resolution in latitude and longitude. For a resolution of one minute in both latitude and longitude (corresponding to ∼1.15 miles N-S and ∼0.89 miles E-W), as called for in existing arms-control agreements requiring the sharing of coordinates information,[11] there will be $7*6*60^2 = 151200$ points. Each point is numbered using its local coordinates (*i,j*). The numbers are converted in base 2 and concatenated to obtain the corresponding binary key *x*. For example, the point of coordinates (7*60, 6*60) on the one-minute resolution grid corresponds to the key $x = 110100100101101000$ of length $l = |x| = 18$ bits.

Because the number of grid points is not too large, we opt for a simplified construction in which there is a leaf node for every grid point. In

Figure 4. Example of message digest for a storage site. Data are encapsulated in multiple levels, which can be revealed at different points in time. The level 0 message contains a random number generated by the host, a random number provided by inspectors to guarantee freshness of the commitment, clear text data, and the hash of the level 1 message. The level 1 message contains additional data (here the number of warheads, missiles, amounts of uranium and plutonium, and their isotopics), which may be released at a later stage, for example prior to inspection. The message digests (generated with SHA3-256) are for illustration purpose only.

scenarios where the number of grid points would make the tree too large to be practical, other cryptographic data structures could provide the necessary properties without requiring the declaring party to store data for every empty grid point.[12]

In a tree of depth l, each leaf is then reached uniquely from the root via the path defined by x (see Figure 3) and contains a commitment on whether there is a site at this location and any other relevant information about the site, if it exists. In the case of North Korea, the number of sites to be declared is expected to be approximately 100–200,[13] much smaller than the number of leaves, 2^l. This construction allows information about any grid point, or any subset of grid points, to be revealed without conveying information about any other grid points.

Step-by-step verification in a freeze scenario

Figure 4 presents a cryptographic commitment (using a hash function) to information about a hypothetical nuclear weapons storage site, which would be stored in the leaf in the tree corresponding to the site location. The commitment is obtained by hashing a message "m_0" containing different pieces: a random number generated by the committing party, another random number provided by an outside party to guarantee freshness of the information (i.e., that the commitment was produced after the outside party's random number was generated),[14] information regarding

the entry including, for example, the type of facility, coordinates, and status, and finally additional information "m_1", which represent commitments to additional data that may be shared at a later stage, for example prior to the inspection of the site in question.

As a preliminary step in a verified denuclearization process following a phased approach,[15] the DPRK could agree on freezing the production of fissile materials and components for weapons as well as on monitored storage of existing weapons. Under this framework, the DPRK would produce a complete escrow of all production, storage, and deployment sites of nuclear weapons, missiles, and associated components. It would then commit to the inventories at each site and agree not to move assets between sites. (Movement patterns between sites could be monitored with satellites.)

To verify correctness of the declaration, the DPRK would invite the United States to perform onsite inspections and verify that the assets and information declared in the escrow are present and valid. During these inspections, accountable items could be tagged with unique identifiers, the United States would become more confident that a freeze is indeed in effect, and that the rest of the declaration, which has yet to be revealed, is correct.[16]

Confidence from the U.S. point of view would increase if sites could be picked at random,[17] although the DPRK may prefer to reveal the location and inventories at each site in the order it decides, for example, starting with sites that are already known or considered less sensitive. Because each site can be revealed without compromising others, the pace of inspections can be adapted to the political process, making this approach well suited for an "action for action" negotiating process, in which both sides would make incremental concessions working towards an ultimate settlement.

Combining the properties of the escrow and the possibility to perform challenge inspections would facilitate the process of establishing completeness of the declaration. If the United States believes it has detected proscribed activities at an undeclared site, it could provide North Korea with the site coordinates, corresponding to a specific key x. The DPRK could then prove whether it has included this specific site in the escrow. If the site is in the escrow, both parties would wait and plan for a future inspection to confirm the correctness of the declaration. If the site is not in the escrow, a special inspection would have to take place to demonstrate that no proscribed activities are taking place at the site. Given this risk of exposure, it would be in the interest of the DPRK to produce a complete declaration from the beginning.

Beyond verifying freeze scenarios, this escrow scheme could be adapted to make commitments about items, bulk materials, and sites on a periodic basis. For each period, the party making the declaration would

cryptographically sign the commitment, such that it cannot repudiate it later. If this signature also covers a hash of the party's signed commitment from the previous period, the result will be a cryptographic block-chain that binds the party to its entire history of commitments.

Security of the escrow

Overall, for the viability of our approach, it is imperative that the message length, the message content, and the implementation of the commitment protocol are robust against all relevant types of cryptographic attacks. When using hash functions, it is important to be mindful of the potential for preimage and collision attacks.[18] Preimage attacks would allow finding a message corresponding to a given hash produced with a particular hash function. These would compromise the secrecy of the information committed in our escrow. Finding collisions and preimages is computationally very difficult, however, assuming a secure hash is used. For example, if the length of a hash is n bits, where n is typically 256 or 512 for modern hash functions, a brute force attack would require $\sim 2^{\frac{n}{2}}$ evaluations of the hash function for finding a collision between two messages, and $\sim 2^n$ evaluations for finding preimages and second preimages. So far, there have been no known successful preimage attacks on NIST recommended hash functions.[19]

What is more typical for older and now considered vulnerable hash functions, however, has been the discovery of collision attacks, which would challenge the binding property of the commitment scheme.[20] A recent practical example is the discovery of the first collision for SHA-1 using a method to produce two PDF documents producing the same hash.[21] Discovery of an SHA-1 collision was anticipated for many years before it occurred, however. In our case, new collision attacks could affect the security of past declarations if they also allow to conduct secondary preimage attacks. These risks could be mitigated by using hash function combiners, allowing multiple hash functions to be combined in such a way that the combination is collision-free if at least one of the constituent hash functions is collision-free.[22] If doubts should arise about the continued collision-freedom of the hash functions being used, commitments could be regenerated to include a new combination of hash functions to provide additional insurance against collisions.

Conclusion

North Korean diplomats could walk to the negotiation table to meet their U.S. counterparts with a 256-bit or 512-bit message on a piece of paper. Using the escrow scheme developed in this paper, this simple message

could represent a commitment to a database containing every single bit of information about their nuclear and ballistic missile programs. Doing so would fulfill a U.S. demand to provide a comprehensive declaration of sites and assets. It would also prevent the United States from walking away with this information, a potentially unacceptable security threat for North Korea.

We showed how to combine our escrow with an inspection regime to verify the correctness and completeness of a declaration of nuclear and other relevant sites in a step-by-step approach. While not all information is available upfront to inspectors, confidence in the validity of the overall declaration grows with each successful inspection. Our approach also allows the inspected party to commit to additional information documenting weapon design, production records, and movement of assets through the weapons complex.

The approach presented in this paper has the potential to resolve a long-standing diplomatic deadlock: The United States wants a correct and complete declaration from the DPRK, which in return does not want to provide a target list that could enable a preventive military attack. Our proposal resolves this tension by allowing the DPRK to commit to such a declaration, which is gradually revealed as the diplomatic process proceeds. In the longer term, the case of North Korea could serve as an important precedent for using modern cryptographic techniques to support nuclear arms-control and disarmament.

Acknowledgements

The authors thank B. Barak, R. J. Goldston, F. von Hippel, and Z. Mian for their comments and feedback.

Notes

1. J. Newhouse, *Cold dawn: The story of SALT* (Washington, DC: Pergamon, 1989).
2. National Academy of Sciences, *The Future of U.S. Nuclear Weapons Policy* (Washington, DC: National Academies Press, 1997).
3. Committee on International Security and Arms Control, *Monitoring Nuclear Weapons and Nuclear-Explosive Materials: An Assessment of Methods and Capabilities* (Washington, DC: National Academies Press, 2005).
4. A. Glaser and Z. Mian, "Denuclearizing North Korea: A verified, phased approach," *Science,* 361 (2018): 981–983; R. S. Kemp, "North Korean disarmament: build technology and trust," *Nature,* 558 (2018): 367–369; N. E. Busch and J. F. Pilat, *The Politics of Weapons Inspections: Assessing WMD Monitoring and Verification Regimes* (Stanford, CA: Stanford University Press, 2017).
5. International Panel on Fissile Materials, "U.S. proposal for verification of North Korea's denuclearization," In Global Fissile Material Report 2009: A Path to Nuclear Disarmament (Princeton, IPFM, 2009), http://fissilematerials.org/library/gov08a.pdf .

6. V. P. Crawford, "A theory of disagreement in bargaining," *Econometrica: Journal of the Econometric Society,* 50 (1982): 607–637; B. Levenotoğlu and A. Tarar, "Prenegotiation commitment in domestic and international bargaining," *American Political Science Review* 99 (2005): 419–433.
7. See Appendix A for a glossary of cryptographic terms used in the manuscript. The mathematical definitions of most terms can be found in O. Goldreich, *Foundations of Cryptography* (New York, NY: Cambridge University Press, 2009).
8. S. Barrett, *Environment and Statecraft: The Strategy of Environmental Treaty-Making* (Oxford: Oxford University Press, 2003); National Research Council, *Verifying Greenhouse Gas Emissions: Methods to Support International Climate Agreements* (National Academies Press, 2010).
9. M. Fischlin, A. Lehmann, and K. Pietrzak, "Robust Multi-Property Combiners for Hash Functions," *Journal of Cryptology,* 27 (2014): 397–428.
10. R. C. Merkle, "A Digital Signature Based on a Conventional Encryption Function," *Lecture Notes in Computer Science,* 293 (1988): 369–378.
11. United States of America, Treaty between the United States of America and the Russian Federation on Measures for the Further Reduction and Limitation of Strategic Offensive Arms (2011).
12. S. Micali, M. Rabin, and J. Kilian, "Zero-Knowledge Sets," In Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, 11–14 October 2003, (Cambridge MA, 2003), 80–91.
13. M.J. Mazarr, G. Gentile, D. Madden, S. L. Pettyjohn, and Y. K. Crane, The Korean Peninsula: Three Dangerous Scenarios (RAND Corporation, 2018) https://www.rand.org/pubs/perspectives/PE262.html.
14. S. Haber and W. Stornetta, "How to Time-Stamp a Digital Document," Advances in Cryptology-CRYPT0'90 Proceedings, Advances in Cryptology (1991): 437–455.
15. A. Glaser and Z. Mian, "Denuclearizing North Korea: A Verified, Phased Approach," 981.
16. S. Fetter and T. Garwin, "Using tags to monitor numerical limits in arms control agreements," In Blechman B.M. (ed.), *Technology and the Limitation of International Conflict*, (Washington, D.C.: Foreign Policy Institute, School of Advanced International Studies, Johns Hopkins University, 1989), 33–54.
17. D. M. Kilgour, "Site selection for on-site inspection in arms control," *Contemporary Security Policy* 13 (1992): 439–462.
18. Preimage resistance means that for any output y of the hash function h, it is computationally hard to find any input x that hashes to that output, i.e., given y, it is difficult to find x such that $h(x) = y$. Collision resistance (or second-preimage resistance) means that for any input x, it is computationally hard to find any other input x' that hashes to the same value, i.e., given x, it is difficult to find $x' \neq x$ such that $h(x) = h(x')$. See: P. Rogaway and T. Shrimpton, "Cryptographic hash-function basics: definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance," In *Fast software encryption*, (Berlin/Heidelberg: Springer, 2004), 371–388.
19. M. J. Dworkin, "SHA-3 standard: Permutation-based hash and extendable-output functions," NIST, Federal Information Processing Standards, (NIST FIPS-202) (2015); Q. H. Dang, "Secure hash standard," NIST, Federal Information Processing Standards (NIST FIPS-180-4) (2015).
20. X. Wang, and H. Yu, "How to break MD5 and other hash functions," In EUROCRYPT *Lecture Notes in Computer Science* 3494 (2005): 19–35; S. Caskey, T.

Draelos, R. Schroeppel, and K. Tolk, "Impacts of collisions within hashing algorithms and safeguards data," In Proceedings of the 47th Institute of Nuclear Materials Management Annual Meeting, 16–20 July 2006 Nashville, TN (2006).
21. M. Stevens, E. Bursztein, P. Karpman, A. Albertini, and Y. Markov, "The first collision for full SHA-1," *Lecture Notes in Computer Science* 10401 (2017): 570–596.
22. Fischlin et al., "Robust multi-property combiners for hash functions," 399.

Appendix A:

Glossary of cryptographic terms

This appendix summarizes the definitions of important cryptography concepts introduced and discussed in the main paper. The definitions are presented in an order facilitating logical connections between them.

String: A string x is a finite sequence of characters expressed over a given alphabet. If the alphabet is the set of binary characters $\{0, 1\}$, then the string over $\{0, 1\}$ is a finite sequence of bits, e.g., 01010 ... 11010. The length of x, also written as $|x|$, represent the numbers of characters in x. For example, if $x = 010101$, $|x| = 6$.

Cryptographic primitive: A cryptographic primitive is a well-established algorithm that is used as a building block for designing higher-level cryptographic protocols. Examples of cryptographic primitives include one-way functions, authentication, encryption and decryption, commitments, and digital signatures.

Commitment scheme: A commitment scheme is a cryptographic primitive that allows a party to commit to a piece of information, or value, while keeping it hidden from others (also known as the *hiding* property). The committing party can release the committed value at a later stage while ensuring other parties it was not altered (also known as the *binding* property).

Cryptographic hash function: A cryptographic hash function (often shortened to "hash function") is an algorithm that maps a message of arbitrary length to a string of fixed length, also known as the *hash* or *digest* of a message. Hash functions are deterministic: the same message always results in the same hash. A small change in the message results, however, in large changes in the resulting hash (a property known as the *avalanche effect*). Hash functions must be easy to compute—for any message m, it must be easy to calculate $x = \mathrm{H}(m)$ – but extremely hard to invert—given a hash x, it must be infeasible to find a message m such that $\mathrm{H}(m) = x$ (a property known as *preimage resistance*). Hash functions must also be *collision resistant*—for any input m, it must be computationally hard to find any other input m' that hashes to the same value, i.e., given m, it is difficult to find $m' \neq m$ such that $\mathrm{H}(m) = \mathrm{H}(m')$. Given their properties, hash functions can be used to implement commitment schemes.

Trees: In computer science, a tree is a data type that represents a hierarchical tree structure, with a *root* node linked to multiple levels of *children* nodes such that no child node can have more than one *parent* node (See Figure A1). The link structure must be acyclic, so that no node is a parent, or grandparent, or other ancestor, of itself. Each node in the tree can be represented as a data structure consisting of a value, together with a list of pointers to children nodes. A tree is said to be *binary* if every node has either two children or none.

Leaf: The *leaves* of a *tree* are the nodes of the tree that have no children. They are also the nodes that are the furthest away from the root of the tree.

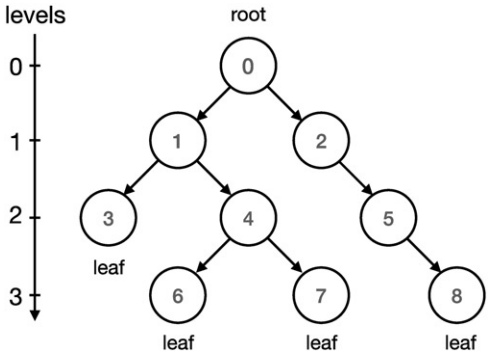

Figure A1. Example of a tree data structure. The node labeled 0 is the root of the tree. The parent of node 5 is node 2. Node 4 has two children, nodes 6 and 7. The nodes 3, 6, 7 and 8 have no children and are called the leaves. The node level corresponds to its distance from the root.

Hash trees: A hash tree (or Merkle tree) is a tree in which every leaf node contains the hash of a data block and every non-leaf node has a hash that is obtained by hashing the concatenation of the hashes of its child nodes up to the top hash in the root of the tree (see Figure 2 of the main manuscript).

ORCID

Sébastien Philippe http://orcid.org/0000-0002-7282-7520
Alexander Glaser http://orcid.org/0000-0001-5960-1239