NIST comments on the initial ZKProof documentation

April 6, 2019

Luís Brandão, René Peralta, Angela Robinson

NIST, Gaithersburg USA

The Privacy Enhancing Cryptography (PEC) team at the National Institute of Standards and Technology (NIST) is interested in the development and dissemination of cryptographic technology capable of enabling privacy. This includes zero-knowledge proofs (ZKPs) and secure multi-party computation (SMPC).

In early 2019, members of ZKProof.org — an open initiative promoting an effort towards the standardization of ZKPs — invited us (at the NIST PEC team) to participate in the 2nd ZKProof workshop (Berkeley, April 2019), and to provide feedback on the ZKProof documentation. That documentation includes the proceedings of the 1st ZKProof workshop (Boston, May 2018), and is publicly available online and open to collaborative editing at https://zkproof.org/documents.html.

We appreciate the character of openness and inclusivity expressed in the ZKProof charter, and welcome the initiative. We find that the ongoing ZKProof documentation has great potential to develop into a valuable reference for secure, practical, and interoperable zero-knowledge proof technology.

While we see the ZKProof effort as aligned with the goal of promoting the development of useful reference material, this does not imply on our part any official position about standardization of the material being developed.

We intend to contribute constructively. This document includes a few editorial suggestions and content-related comments. We are glad to be able to participate in the upcoming workshop and we anticipate more detailed discussion will take place therein and thereafter.

1. Selected comments

1.1. Editorial — adjustments and indexation

C1. **Reference document.** Consider merging the set of proceedings, and ongoing contributions across workshops, into one consolidated reference document on ZK proofs, in a manner that promotes consistent style and notation across all sections, and enables future contributions from multiple sources. For this purpose, consider:

- defining the development of the proceedings of each workshop to be limited in time (having a closure, acknowledging its possible informal content, identifying all participants);
- building the reference document as a coherent combination of the relevant content produced across time, taking into account all contributions.

C2. **Recommendations and requirements.** To highlight suggested and essential practices, consider enhancing the identifiability and organization (e.g., indexation) of "recommendations" and "requirements" throughout the document.

C3. **Scope of the Creative Commons license.** The current CC BY 4.0 International license expressed in the Charter is focused on "content issued from the ZKProof Standards Workshop" (notice the singular). Consider widening this to cover the collaborative edits performed and to be performed within the editable documentation made available for community collaboration.

C4. **Glossary.** Consider adding a comprehensive glossary, listing all technical terms and providing corresponding links to where each term is defined, exemplified, and used in the document.

1.2. Editorial — producing new content

C5. **Executive summary.** Consider adding an executive summary, describing at a high level the structure and content of the overall reference documentation. Consider doing the same for each chapter or track.

C6. **Examples.** To enhance accessibility to a broader audience, consider enhancing the document with indexed examples that illustrate concepts that may be unfamiliar to some target audience. Each example can be highlighted with a caption (e.g., "Example 5: ZK proof setup with a CRS with trapdoor"), an explanation (possibly an illustration) within a boxed environment, and a footnote identifying the included concepts (e.g., "setup, trapdoor, CRS, prover and verifier").

1.3. Track 1: security/theory

C7. **Proofs of knowledge.** Consider making a clearer distinction of ZK proofs of membership vs. ZK proofs of knowledge, including by means of examples and definitions. Consider clarifying how the formalism can adequately model proofs of knowledge. A definition of an "extractability" property/game may be useful.

C8. **Concurrency.** Aspects of concurrency could be addressed more explicitly. Do the prover and verifier know in which session they are interacting? Consider mentioning the need for session ids.

C9. **Transferability.** The concept of transferability could benefit from more attention. For example, in an interactive protocol over the Internet, how do regular authenticated channels vs. "ideally" authenticated channels affect transferability? Would a non-transferable protocol become transferable when the prover signs all sent messages and the verifier uses the output of a cryptographic hash function to select random challenges?

C10. **Circuits vs. R1CS.** The first track ("security" / "theory"?) mentions Boolean circuits but not R1CS. The third track ("implementation") focuses on R1CS without explaining why/when it is preferable to a circuit representation. Consider explaining better (in the "security" track) what is R1CS. Consider introducing and exemplifying a circuit-to-R1CS translation and/or vice-versa. Consider clarifying better in the "implementation" track why the focus is on R1CS, for example compared with circuits.

C11. **Common vs. public.** Consider clarifying the distinction between common knowledge (between prover and verifier) and public knowledge. The lack of distinction is noticed in several parts when trying to think of a comparison between transferable vs. non-transferable cases. CRS is being defined as public, although in practice it could be obtained as common to the intervening parties, private to a particular interaction.

1.4. Track 2: Applications

C12. **Motivation.** Section "2.1 Introduction and motivation" could benefit from more motivation about the three application use-cases that will be discussed. Consider providing a short intuitive explanation about each one.

C13. **Gadgets.** The enumeration (table) of gadgets is very useful. Consider completing the table.

C14. **Interactivity vs. transferability.** In Section 2.2, consider revising the assertion in item 1: "Only non-interactive ZK (NIZK) can actually hold this property" [being publicly verifiable / transferable?]. If transferability is a design goal, then there are settings where it is possible to design interactive protocols for which the view (transcript) of the original verifier (interacting with the original prover) can later serve as a transferable proof for other verifiers.

C15. **Implicit scope of use-cases.** The last paragraph in section 2.2 says "digital money based applications belong to the first model" [public verifiable as a requirement]. This assertion appears implicitly scoped in a too narrow subset of conceivable applications about digital money. Conversely, one could consider a scenario where Alice wants to convince Bob, in a non-transferable way, that Alice bought something from Charlie. Consider clarifying better the scope of examples vs. the scope of areas of application.

C16. **References.** Consider adding references when mentioning "while adapting to the existing Identity standards". Same comment of adding references applies (across sections) to other cases where specific prior results, definitions, claims, etc. are mentioned but not referenced. (This does not intend to suggest that the document becomes a survey, but simply that what is mentioned in concrete be supported with corresponding references that the reader can lookup for fact-checking and further reading.)

1.5. Track 3: Implementation

C17. **Backend choice NIZK-R1CS.** Consider providing more rationale for the choice on NIZK and R1CS. Section 3.2 could benefit from a comparative overview of the various low-level backend options for representing relations. Comparing the advantages and disadvantages of interactive vs. non-interactive, and of several representations (e.g., including arithmetic circuits), may open more room for future document contributions on the cases that have not yet been explored in the existing documentation.

C18. **Computational security parameter.** Consider providing rationale for the recommendation of 120 bits of computational security.

C19. **Statistical security.** Consider discussing various examples of acceptable values of statistical security parameter. It can be useful to explore how interactive to non-interactive transformations may affect the requirements on the statistical security parameter, e.g., making it become a computational parameter when applying Fiat-Shamir.

C20. **Side-channels.** Consider exemplifying conceivable cases where side-channels are problematic.

C21. **Validation.** Consider including some discussion on testing and validation of implementations.

C22. **Intellectual property.** Consider discussing possible guidance regarding intellectual property.

2. Towards a reference document

2.1. Setting expectations and orientation

Consider setting, during the 2nd workshop, some expectations for the development of a ZKProof reference document. Such expectations may be useful for receiving more targeted feedback from the community. Concepts to agree upon might include: title, purpose, aim, scope, and editorial methodology.

D1. **Title.** Something that identifies the "reference" aspect, e.g., "ZKProof community reference on zero-knowledge proofs".

D2. **Purpose.** For example: *The purpose of developing the ZKProof reference document is to provide, within the principles laid out by the ZKProof charter, a reference for the development of zero-knowledge-proof technology that is secure, practical and interoperable.*

D3. **Aim.** For example: *The aim of the document is to consolidate the reference material developed in collaborative processes during the ZKProof workshops. The document intends to be accessible to a large audience, which includes the general public, the media, the industry, developers and cryptographers.*

D4. **Scope.** For example: *The document intends to cover material relevant for the development of secure, practical and interoperable technology, as identified in the* purpose*. The document will also elaborate on introductory concepts or works, as a way to enable an easier understanding of more advanced techniques. When a focus is chosen from several alternative options, the document should try to include a rationale describing, if possible, comparative advantages, disadvantages and applicability. However, the document does not intend to be a thorough survey about ZKPs, and does not need to cover every conceivable scenario.*

D5. **Format.** For example: *To achieve its accessibility goal, and considering its wide scope, the document favors the inclusion of: a well defined structure (e.g., chapters, sections, ...), executive summaries (one general and one per chapter); illustrative examples covering the main concepts; enumerated recommendations and requirements; summarizing tables; glossary of technical terms; appropriate references for presented claims, results.*

D6. **Editorial methodology.** For example: *The primary direction for the development of reference material arises in connection with the ZKProof workshops. Each workshop produces proceedings that provide a scope of content; then, a process ensues for integration of material into a consolidated reference that can evolve across several workshops. Assuming a 12-months gap between workshops, here is a possible 11-months editorial process to evolve the reference document:*

1. *(Session chairs) 2 months to produce workshop proceedings (informal, focused on discussed content and suggestions).*
2. *(Editors) 2 months to integrate the new proceedings into the evolving reference document. This step culminates with the release of: a new "beta" version of the reference; a "diff" highlighting the changes since the last version; as needed, a description of editorial decisions; and a call for comments, indicating areas that may require specialized contribution.*
3. *(General public) 2 months open to submit public feedback.*
4. *(Editors) 1.5 months to integrate the feedback, essentially repeating step 2, but also publishing, in the diff, a table with all received comments and cross-referencing them to all changes made.*

5. *Repeat step 3.*

6. *Repeat step 4, and let the output be called the new "stable" version.*

At each step of the process, the ZKProof steering committee supervises the progress and issues recommendations or formal opinions, as needed.

2.2. A possible starting point

D7. **Initial compilation.** Consider enhancing the indexation and labeling features of the reference document, to ensure that any part of the content can be easily referenced in a direct manner when making comments or proposing adjustments. As an initial contribution in this direction, we ported to LaTeX the content of the six original documents available online, and compiled it into a single document, making several editorial adjustments therein. We make this compilation and its source-code available for appreciation by the ZKProof team. A summary of the editorial adjustments:

- indexed all lines, pages, sections, subsections, tables, figures;
- uniformized the numbering and style of sections and subsections;
- upgraded some paragraph headers to subsections;
- added captions to the figure and the tables; split some tables;
- added table of contents, lists of tables and figures, PDF bookmarks and hyperlinks;
- added bibtex source for references and generated the corresponding tags and hyperlinks;
- used math font for math symbols;
- moved around and merged some repeated/related material (e.g., change log, references, external resources, glossaries);
- initiated a list of acronyms;
- added some pop-up annotations with minor suggestions.