SoK: Towards a Unified Approach of Applied Replicability for Computer Security

Abstract

Reproducibility has been an increasingly important focus within the Security Community over the past decade. While showing great promise for increasing the quantity and quality of available artifacts, reproducibility alone only addresses some of the challenges to establishing experimental validity in scientific research and is not enough to move forward our discipline. Instead, replicability is required to test the bounds of a hypothesis and ultimately show consistent evidence to a scientific theory. Although there are clear benefits to replicability, it remains imprecisely defined, and a formal framework to reason about and conduct replicability experiments is lacking. In this work, we systematize over 30 years of research and recommendations on the topics of reproducibility, replicability, and validity, and argue that their definitions have had limited practical application within Computer Security. We address these issues by providing a framework for reasoning about replicability, known as the Tree of Validity (ToV). Using the ToV framework, we evaluate two papers with Distinguished Artifact Awards and demonstrate that true reproducibility is often unattainable; however, meaningful comparisons are still attainable by replicability. We expand our analysis of two recent SoK papers, themselves replicability studies, and demonstrate how these papers recreate multiple paths through their respective ToVs. In so doing, we are the first to provide a practical framework of replicability with broad applications for, and beyond, the Security research community.

1 Introduction

Prompted by growing concerns of a reproducibility crisis across all of science and engineering, the Security Community has recently enacted Artifact Evaluation Committees (AECs) in many of its most important venues. The promises of AECs appear to be many, from creating an opportunity for current authors to demonstrate that their claims are computationally reproducible to providing future researchers with artifacts to more easily compare their results to prior efforts. While the measured benefits of reproducibility have not yet been fully demonstrated [26], the widespread practice of making artifacts available and reproducible is likely to improve the value of results published by this community.

Even with the potentially substantial increase in the number of artifacts available to the community, significant challenges remain. For instance, while the term *computationally reproducible* is rigidly defined to mean a different team using the same code and data to achieve the same result [25], there are few meaningful definitions that assist in reasoning about the differences between two efforts when reproducibility of a study is not possible, such as when the underlying data or the experimental setup are unavailable or the environmental conditions are unattainable.

In such cases, *replicability* tests the limits of a hypothesis under new conditions, providing a new understanding of the experiments that reproducibility alone cannot achieve. Replicability advances a new understanding of areas and shows a trend across multiple independent studies. Other areas of science (e.g., Medicine and Psychology) primarily focus efforts on replicability. As reproducibility and replicability control for different outcomes, the lack of reproducibility does not mean a study is not replicable and vice versa. While this community's efforts have primarily focused on reproducibility, we introduce formal frameworks and methods to discuss replicability. For the remainder of this paper, we will refer to the broad field of reproducibility/replicability as *Validity* to be precise in our definitions and avoid confusion.[1]

In this paper, we make the following contributions:

- **Systematization of Validity:** Previous work, not only in computer security but all of computational science, does not develop a comprehensive or directly applicable definition of replicability. We systematize 30 years of meta-science on this topic with the goal of providing a practical understanding of replicability.
- **Framework of Validity:** We provide a new approach to

[1] Validity has been used to describe reproducibility, but it has never to our knowledge been used to describe the field.

classifying reproducibility/replicability using a binary tree of available artifacts, the Tree of Validity (ToV). This framework unifies with previous definitions of validity and applies a structure for comparing validity studies. We show that this framework is not only pertinent to the Security Community, but to all of computational science.

- **Application to Case Studies:** We provide two case studies on individual papers that have recognized Distinguished Artifact awards [5,42], showing how each can be mapped to a ToV and how reproducibility for both is limited in spite of their award status. We then more broadly show the application of our framework to previous SoK papers [21, 36], which show how our ToV framework unifies multiple experimental efforts that move toward improved claims of validity.

The benefit of this work is that it not only conceptually systematizes the topics of replicability and validity, but that it also provides a practical framework and guidance allowing researchers to better contextualize their contributions to a research area and ideally facilitating a more rapid advancement of science.

The remainder of this paper is organized as follows: Section 2 provides definitions of terms; Section 3 systematizes this space; Section 4 introduces the ToV framework, its goals, and visualizations of multiple possible ToVs; Section 5 performs case studies on two award-winning papers and two SoKs to show the applicability of the ToV framework; Section 6 offers discussion and open problems; and Section 7 provides concluding remarks.

2 Definitions

We provide definitions of various parts of experimental methodologies to avoid confusion.

We refer to *validity* as the field of science that reproducibility and replicability are a subset of. A validator is a group or team that conducts experiments to gauge the validity of another author's work, through reproducing or replicating the experiments. A validity experiment is any such experiment that is in part based on the original authors' work.

For experiments, we define the *setting* as composed of a problem and domain. We define the *problem* as a scientific question that the experiments provide evidence to (e.g., deepfake detection). The *domain* is the environment of the experiment. It can include but is not limited to the population studied, time, software systems, hardware systems, etc. As an example of a setting, detecting malicious network traffic is the problem with the domain being a specific network.

The *process* is the experimental methodology conducted in a setting. A *method* is the approach to gather and/or manipulate data. The *data* is the collection of measurements or observations within the setting. Finally, the *analysis* is conducted on the data and provides a quantifiable measure. To follow the above example, the data is the network traffic (e.g., TCP connections). The method is how the malicious traffic is detected (e.g., a machine learning detector). The analysis is then a gauge of detecting the performance of the method (e.g., false positives).

3 Systematization

To motivate the development of our framework, we provide a systematization of the numerous proposals for defining reproducibility and replicability. While the discussions on reproducibility have resulted in a clear definition, replicability remains vague and largely unexamined. As such, our systematization shows that there remain open problems with the definitions of replicability. This systematization is not exhaustive of every available definition of reproducibility. Rather, it seeks to highlight research into reproducibility and replicability that aim to clarify misunderstandings of validity and build frameworks to address challenges within validity. Figure 1 shows the relative relation between all of the definitions systematized in this work.

3.1 Claerbout Terminology

The *Claerbout Terminology* [30] refers to some of the first papers to formally define computational reproducibility. We highlight the two papers associated with building the basis for these definitions: Claerbout et al. [7] and Peng [29].[2]

Claerbout et al. [7] - With the proliferation of computational capabilities, Claerbout et al. [7] identified that anyone should be able to validate computational results. This is due to the innate ability of the original authors to code experiments, requiring that the "validator" only runs the program. As such they structured their classes and research group to promote reproducible research and identified reproducibility as being able to run the same software on the same input and obtaining the same results. Claerbout et al. [7] proposed several goals for reproducible research: first, to "teach researchers how to ... reproduce their own research results a year or more later" (i.e., longevity); second, to "learn how to leave finished work in a condition where coworkers can reproduce the calculation including the final illustration by pressing a button" (i.e., iteration); and third, to "prepare a complete copy of ... a local software environment so that graduating students can take their work away... to other sites ... and reproduce their work." They achieve this goal by outlining a systematic way to prepare research artifacts, where documents are created using software that runs the code for each experiment in the compilation of the document.[3] This is one of the first definitions of reproducible research and self-contained experiments.

[2] Plesser [30] is the first to refer to this taxonomy as the Claerbout Terminology, but Barba [3] and Liberman [23] also group these papers together.

[3] They modernized this work into a formal framework called ReDoc in Schwab et al. [35].

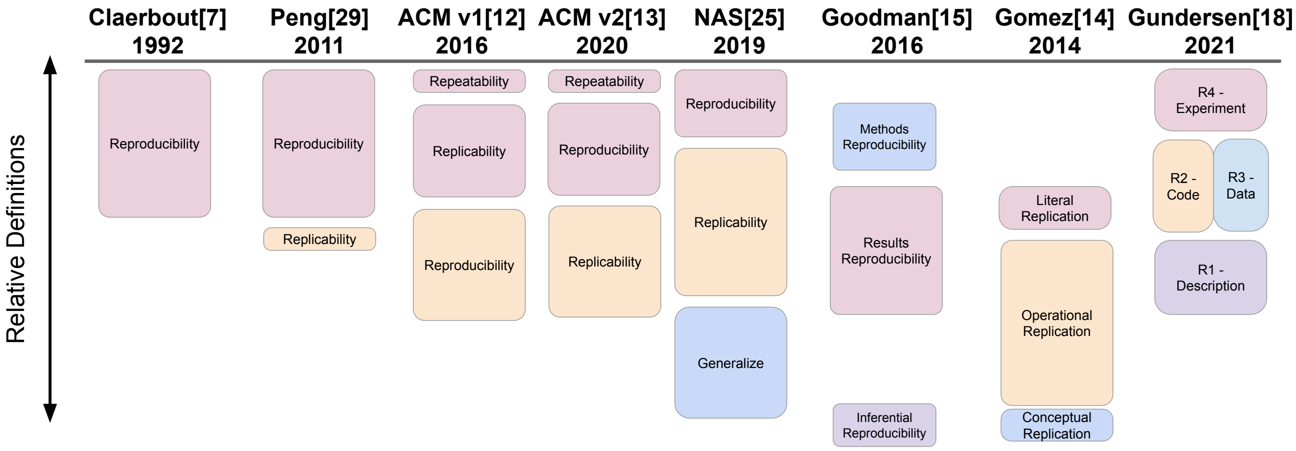


Figure 1: This figure shows the relative definitions from each paper surveyed in this systematization. A larger box shows that there is a conceptual spectrum surrounding the word and covers multiple meanings. Similar figures exist in [3, 14, 23, 30].

Peng [29] - In discussing the challenges faced by computational reproducibility, Peng argues that replicability is the ultimate standard. Replicability is where "independent investigators address a scientific hypothesis and build up evidence for or against it." This constitutes an independent verification of the results proposed by the research. He concedes that replication is often unavailable due to costs associated with fully recreating an experiment. In such cases, reproducibility can act as a minimum standard where the software and data can be re-analyzed.

Peng builds upon Claerbout et al. [7] by introducing reproducibility as a spectrum. He argues reproducibility exists in terms of the availability of the artifacts starting from not reproducible with nothing made available to code available to code and data available to linked and executable code and data. Full reproducibility, while not a substitute for "gold-standard replicability," can provide further proof for the validity of a claim. Thus, Peng defines replicability as a fully independent attempt to build further evidence for or against a scientific hypothesis and reproducibility as the re-analyzing of available code and data. While Claerbout et al. [7] provide a strict definition for reproducibility, Peng [29] provides a strict definition for replicability (e.g., fully independent experiments) and reproducibility as a spectrum. Counterintuitively, Peng's definitions put fully reproducible (i.e., only using the original artifacts) as the closest substitute to replicability. Yet replication experiments would exist at the opposite end of the reproducibility spectrum because Peng's replicability is fully independent of the original authors. Thus, the closer one's experiment to full reproducibility, the farther from the replicability standard. This spectrum does not provide context for reproducing results by using only part of the available artifacts.

Takeaway *The Claerbout Terminology provides the foundational understanding of reproducibility, yet fails to define replicability. Fundamentally,* **reproducibility and replicability are interchangeably treated as both properties and actions.**

3.2 Definitions by Organizations

The growing concerns for reproducibility in the mid-2010s prompted organizations to undertake investigations into defining reproducibility. Although numerous organizations have addressed and proposed reproducibility definitions, we discuss the two most prominent organizations that influence the Security community's understanding of reproducibility, the Association for Computing Machinery (ACM) and the National Academies of Science (NAS).

ACM - The ACM released their first definitions for reproducibility derived from the International Vocabulary of Metrology in 2016. In contention with the Claerbout terminology, they define *Repeatability* as "same team, same experimental setup", *Replicability* as "different team, same experimental set up" and *Reproduciblity* as "different team, different experimental setup" [12]. After identifying the contentions between the ACM's definitions and broader computational sciences, this version was later unified with the Claerbout terminology in 2020 with *Reproducibility* as "different team, same experimental setup" and *Replicability* as "different team, different experimental setup" [13].[4] The ACM measures these properties, which results in badges awarded to published papers.

The ACM provides three badges, *Artifacts Evaluated*, *Artifacts Available*, and *Results Validated*, each with levels of the property contained within the badge [13]. The Artifacts Eval-

[4] Liberman [23] identifies that the opposing definitions were introduced to computational science by Drummond [10].

uated badge contains two levels, Functional and Reusable. A functional award is "documented, consistent, complete, and exercisable." The associated repository must contain an inventory of the artifacts included and a sufficient description to enable the artifacts to be exercised (i.e., documented). A consistent artifact is relevant to the associated paper, contributes to the main results of a paper, and is complete if "all components relevant to the paper in question are included" [13]. Finally, a functional artifact is exercisable if all included software runs and generates results. The ACM defines a Reusable artifact as "significantly exceeds minimal functionality." With no additional properties beyond Artifact Evaluated - Functional, the Reusable badge contains "carefully documented and well-structured to the extent that reuse and repurposing is facilitated" [13] artifacts. An Artifact Available badge is awarded when the authors make the artifacts available through a publicly accessible archival repository.

The Results Validated badge contains two levels, Results Reproduced and Results Replicated [13]. Results Reproduced is awarded when a subsequent study by an independent person or team obtained the main results "using, in part, artifacts provided by the author" [13]. Results Replicated is awarded when an independent person or team obtains the main results "without the use of author supplied artifacts" [13]. We note that the badging does not fully reflect their definitions. The ACM provides a strict definition of reproducibility, but the badges do not follow this strict definition. For example, the Results - Reproduced badge only requires using "in part, artifacts provided by the authors" [13], but the revised definition of reproducibility "means that an independent group can obtain the same result using the author's own artifacts" [13]. Thus, these definitions do not fully align.

Takeaway *The practice of reproducibility and replicability does not always align with the conceptual definitions, leading to inconsistent application and miscommunication.*

NAS - As the growing concerns for reproducibility became more public, in 2017 the United States Congress entreated the National Science Foundation (NSF) to engage the NAS to investigate reproducibility and replicability and provide recommendations to improve the field. The NAS released a book on reproducibility and replicability in 2019 [25]. This body of work represents a synthesis of over two years of exploratory committees to define reproducibility and replicability. The NAS explored numerous areas of research (e.g., Biology, Biomedical, Economics, Psychology, and Medicine) to understand current undertakings of reproducibility and aims to unify definitions across disciplines. This report highly focuses on meta-science research and analyzes the applications of reproducibility and replicability, providing extensive recommendations for fields of science as well as guidelines for conducting reproducibility studies.

To differentiate between reproducibility and replicability, the NAS first focuses on the underlying concepts. They pose the following iterative questions related to reproducibility and replicability:

(1) "Are the data and analysis laid out with sufficient transparency and clarity that the results can be checked?"

(2) "If checked, do the data and analysis offered in support of the result in fact support that result?"

(3) "If the data and analysis are shown to support the original result, can the result reported be found again in the specific study context investigated?"

(4) "Finally, can the result reported or the inference drawn be found again in a broader set of study contexts?"

From these questions, the NAS conceptually defines the demarcation between reproducibility and replicability between questions (2) and (3). Thus, reproducibility is focused on verifying the existing claims of a paper by checking the available artifacts of a study. Replicability is defined as testing the limits of the study and broader implications. This introduces two new properties of reproducibility and replicability, indirect and direct. Direct studies imply that there was an experimental procedure that was conducted, whether it was the exact same artifacts of the authors (i.e., reproducibility) or new data and experiments (i.e., replicability). Olszewski et al. [26] performed an indirect and direct reproducibility study on ML Security.

We note that the usage of indirect is not the same between reproducibility and replicability. While an indirect study of reproducibility assesses "the extent of the availability of computational information" [25], an indirect replicability study is not defined. Upon examining the studies the report classifies as indirect replicability studies, these studies appear to focus on assessing the validity of the methodology used across several studies (e.g., "49.6% of the articles with null hypothesis statistical test (NHST) results contained at least one inconsistency" [25]). Further, the NAS report claims that direct reproducibility is rarer than indirect reproducibility due to the difficulty of conducting a study, showing that 9/13 surveyed reproducibility studies are indirect. Although no formal definition for indirect replicability is provided, they highlight 19/22 direct replicability and 3/22 indirect replicability studies.

The NAS definitions motivate two ideas. First, it provides a term for a property implicitly identified by Peng, indirect; there is a limit to the reproducibility of a paper due to the availability of artifacts. Second, it motivates the difference between replicability and reproducibility by the purpose of conducting the study. For example, reproducibility is checking that the artifacts support the main results (e.g., Question 2), while replicability is testing the finding in new contexts (e.g., Questions 3 and 4).

Takeaway *The current taxonomies do not have a unified convention for desired properties. There is a missing connection between the purpose, outcome, and realization of reproducibility and replicability.*

3.3 Expanded Definitions

The Claerbout Terminology provided the foundation of computational reproducibility, and organizations adopted versions of it to promote reproducible research. As noted above, there are several limitations within this theoretical understanding of reproducibility and its adaptation to computational science. In this section, we discuss works that build upon the previous work by relying on different concepts to derive a taxonomy.

Gomez et al. [14] - In synthesis of a survey of 20 replication classifications, Gomez et al. derive a framework for replicability motivated by the purpose of the replication. They define the operationals (i.e., parts) of an experiment as the protocol, the operationalization, the population, and the experimenter. The protocol is the experimental methodology followed to perform the experiment. The operationalization is the control and response and how this effect is measured. The population is the studied population, and the experimenters are the ones performing the experiment. Gomez et al. designed the framework for the function of the replication experiment. By modifying the operationals, the function of the experiment changes. They list six functions of replication: (1) to control for sampling error; (2) to control protocol independence; (3) to understand operationalization limits; (4) to understand population limits; (5) to control experimenter independence; and (6) to validate hypotheses. As an example, modifying the experimenters of the replication fulfills the function (5), to control for experimenters' independence. From these modifications, Gomez et al. [14] define literal replication as "new runs of the experiment on another sample of the same population", operational replication as a broader category of replications that modify operations of the experiment, and conceptual replication as changing every operation of the experiment. Operational replication can further be denoted by the changes in the experiment (e.g., changing the experimenters would be Operational Experimenter Replication).

Of particular interest in this framework is the function of the replication. Gomez et al. [14] primarily derive these functions from Schmidt [34]. The purpose of running a replication experiment can be defined before conducting the experiment. Function (1) controlling for sampling errors is to determine that the results do not happen by chance. This function is achieved through literal replication (i.e., running the same experiment on a new collection of data from the same population). Function (2) identifies that changing the protocol does not modify the results. Schmidt [34] formulates this function as "controlling for artefactual results (internal validity)." Thus, the function ensures that the result is not due to a specific tool or methodology (e.g., a faulty thermometer). Function (3) controls the operationalizations of the experiment and ensures that the result is not due to how the response is measured. For example, Function (4) assesses the extent the result is unique to the studied population. This is formulated as generalizability by the NAS [25]. Function (5) determines whether the result is independent of the original researchers. Schmidt [34] calls this a control for fraud. Function (6) seeks to validate the hypothesis. Gomez et al. [14] define conceptual replication to meet this function.

This framework focuses on the function of replication to derive their definitions of replicability. We note that this framework does not consider Claerbout's reproducibility. As such, the functions of replicability are not comprehensive to validity. Although designed for Software Engineering, this framework does not intuitively map to computational experiments. As Peng points out, some methodologies rely on massive datasets that additional studies cannot recreate. Further, this framework focuses on complete replicability that does not use any experimental artifacts from the original authors. As such, although they map literal replication to repetition and conceptual replication to reproduction, these definitions are not consistent with the Claerbout terminology.

Takeaway *The purpose of a validity study is inherently connected to the changes made within the experiment, yet no taxonomy currently unifies the function, the availability, and the actualization of the experiment.*

Goodman et al. [15] - Goodman et al. define three types of reproducibility: method reproducibility, results reproducibility, and inferential reproducibility. Method reproducibility is the provision of "enough detail about study procedures and data so the same procedures could, in theory or in actuality, be exactly repeated." [15] Although they focus their discussion on the theoretical ability to reproduce experiments, they capture a similar property as the NAS, that there exists a theoretical ability to be methods reproducible (i.e., indirect) and an actualization of method reproducibility (i.e., direct). They describe results reproducibility as their version of replicability. It is "obtaining the same results from the conduct of an independent study whose procedures are as closely matched to the original experiment as possible" [15].

They briefly mention robustness (i.e., the stability of results across variations of the experiment) and generalizability (i.e., "the persistence of the effect outside of the experimental framework" [15]). Inferential reproducibility refers to the interpretation of the results after a study is conducted (i.e., a paper is inferential reproducible if an independent team comes to the same conclusions regardless of methodology). Therefore, while an independent team may run similar experiments and achieve results reproducibility, they may fail to be inferential reproducible if the independent team draws different conclusions. They frame this discussion around Bayesian statistics. "If a finding can be reliably repeated, it is likely to be true, and if it cannot be, its truth is in question" [15]. Finally, they identify problems that result in why an experiment may not be reproducible. This implicitly points to Gomez's functions. Further, they discuss that the results of reproducibility studies often cause disagreements between the original authors and the reproducer. This is caused by the misconception of terminology and misrepresented methodology choices. The demarcation between these three definitions appears to be in what part of speech they are. Method reproducibility is described as a property of published research, results reproducibility is an action taken by other researchers, and the unclear terminology can result in disagreements about the reproducibility study.

Takeaway *The outcome of validity experiments builds evidence towards a scientific theory. The current terminology does not reflect that replicability is iterative and cannot compare between two independent validity studies.*

Gundersen [18] - While the previously discussed frameworks focused on properties or functions to derive functions of reproducibility, Gundersen [18] derives their framework for AI/ML reproducibility from the scientific method. They motivate the design of their framework by breaking the MNIST dataset [22] into the step-by-step tasks needed to re-implement the methodology. For example, to collect handwritten digits, one would have to first gather writers, second, have them write the numbers, and finally digitize the numbers. Each other task within an experiment can be further broken down into sub-tasks (e.g., data processing can be broken into removing outliers and normalizing the data).

Gundersen defines three degrees of reproducibility: outcome reproducible, analysis reproducible, and interpretation reproducible. Each of these definitions is defined as the outcome of an experiment (i.e., research is not "X reproducible" until the experiment has been attempted). Outcome reproducible is defined as the same outcome from the original experiments. Analysis reproducible is when the outcome is not necessarily the same, but the same analysis leads to the same interpretation. Interpretation reproducible is when both the outcome and analysis are different but lead to the same interpretation.

The three degrees of reproducibility are affected by what is made available by the original authors. For this, Gundersen defines four reproducibility types: R1 - Description, where only the description of the experiment is used; R2 - Code, where the code and description are used to reproduce the experiment; R3 - Data, where the description and the data are used; and R4 - Experiment, where the documentation, data, and code are all used to reproduce the experiment. Gundersen is one of the first to provide definitions that reflect what is used from the original experiment (e.g., the same data is used in the reproduction experiments). The reproducibility type is iterative though and, as such, does not handle varied methodologies. This framework demonstrates that reproducibility and replicability are affected by what is used from the original experiments. While an improvement to the ACM's definition, the same data oversimplifies the rest of the experiment. Although it addresses several deficiencies in previous frameworks, it does not account for what the original paper can achieve. Thus, Gundersen treats reproducibility as an outcome of an experiment and fails to unify reproducibility and replicability as properties and actions.

Takeaway *Current frameworks are not robust to the varying implementations of the scientific process. Further, reproducibility as an outcome is affected by what artifacts are used from the original experiments.*

3.4 Lessons and Problems

This body of work summarizes over 30 years of work, yet several deficiencies remain with how validity is defined and formalized. In this systematization, we see that reproducibility is described as a property and as an action of an experiment. For example, an experiment could be described as reproducible because one *could* run the experiments or reproducible because one *has* run the experiments. We note that this is often part of the confusion one runs into when discussing validity. While Gundersen attempts to address this by explicitly stating that reproducibility is based on the conclusion of a validity experiment, this ignores that there are limits to the extent the experiments could be reproducible. Further, this does not allow comparison between subsequent validity studies. As such, *there is no way to reason or compare between two independent validity studies*. We address this limitation in our framework.

No taxonomy explicitly establishes that reproducibility and replicability are not mutually exclusive. An experiment could be reproducible but not replicable, replicable but not reproducible, or both reproducible and replicable. For example, if a paper has provided all artifacts, one could reproduce all of their experiments by running their code and ensuring that the artifacts run and output a similar result. One could also replicate the study by coding these experiments from scratch to demonstrate that it is replicable at some level. These taxonomies treat reproducibility and replicability as an ill-defined hierarchy when they control for different outcomes.

Finally, none of the previous work is meaningfully deployed, as they fail to provide a robust and actionable framework for assessing and comparing experimental validity across studies.

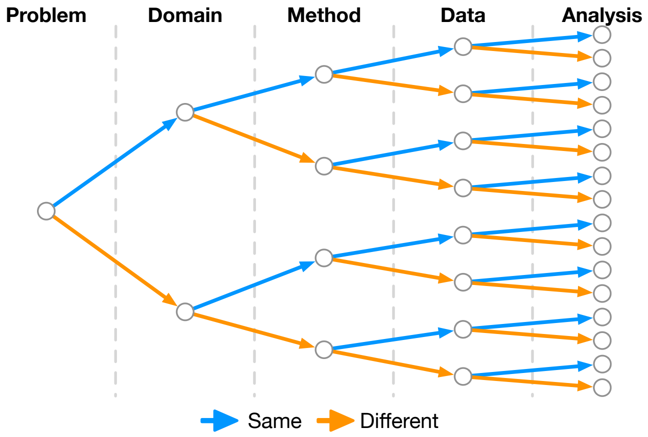


Figure 2: The Tree of Validity shows every possible validity experiment by using the same or different parts of an experiment. We emphasize that this tree can be modified to express different methodologies by swapping or creating new layers in the perfect binary tree.

4 Framework

From our systematization, we identify that there are several shortcomings within reproducibility and replicability frameworks. As such, the current definitions of reproducibility and replicability do not attain a unified framework. Our goal is not to provide new "words" or "adjectives" and provide a commentary on how researchers should favor one definition over the other. Instead, we introduce a new framework for characterizing validity. Our goal is to provide a framework that unifies the previous definitions and addresses the misunderstandings and innate properties of previous frameworks. In this section, we discuss the goals we aim to achieve and the properties this framework should have, formalize the framework, and show it unifies with previous understanding of reproducibility and replicability. We will then use this framework to map prior efforts in Section 5.

4.1 Goals

In Section 3, we outline the previous frameworks and their deficiencies. From the previous taxonomies, we define reproducibility as a form of validity of the physical manifestations of experiments (e.g., code or data). Replicability is designed to build evidence towards or against the initial hypothesis. To address the limitations in previous frameworks, our framework should meet the following goals:

- *Unify with previous frameworks.* The previous taxonomies and frameworks establish a foundation for the field of validity. As such, any proposed work should not oppose these definitions. Our framework should aim to consolidate these definitions into one structure.
- *Reconcile validity as a property and as an action.* Much of the confusion between the taxonomies is due to treating published work as reproducible in that it has been reproduced and as reproducible in that it could be reproduced again in the future. Our framework should address the potential for validity and reconcile the inherent limitations due to the available experimental artifacts.
- *Accommodate different fields and varied methodologies.* The Security community encompasses numerous subfields, and security research is growing in complexity. As such, the proposed framework should be accommodating to the field and able to handle complicated methodologies.
- *Quantify differences between validity studies.* When validity studies are conducted, they indicate the changes made by calling the study either reproducible or replicable. Mapping such complex tasks to two or three words is incomplete and does not allow for direct comparisons to be made. As such, our framework should succinctly express the differences between studies.

4.2 Tree of Validity

To motivate the construction of our framework, consider the perfect case for reproducibility where the published research provides all experimental artifacts. In this case, it would be possible to use any part of the original experiment to conduct a validity experiment. We broadly formulate our framework around the following experiment: an experiment addresses a *problem* within a *domain*. We refer to this as the *setting* as seen in Figure 2. A *method* is applied to create *data* and then *analysis* is conducted on the data, referred to as the *process*. Each of these parts of an experiment will create the layers of a perfect binary tree. The decision of the binary tree is whether the part remains the *same* or *different*. We call this the Tree of Validity (ToV) and visualize it in Figure 2. This Tree of Validity shows every iteration of possible validity experiments. For example, following the upper path through the ToV (i.e., keeping everything the same) would create a validity experiment that Claerbout would describe as reproducibility. We show how each definition from Section 3 maps to the Tree of Validity in Figure 3. Thus, the comparison between definitions is no longer relative but quantifiable.

The layers of the Tree of Validity are not static, although changes to the *setting* are most likely not to occur. There are numerous enumerations of using the same parts of the original experiment depending on how granular the tasks of the experiment are expressed. For example, the steps to the experiment could be a *methodology* that consists of a software and hardware component. Thus, the *method* layer can be split into a *software* and a *hardware* layer. In another example, an experiment may wish to split the *data* into a *train* and *test* layer for a machine learning problem. Further, the layers can

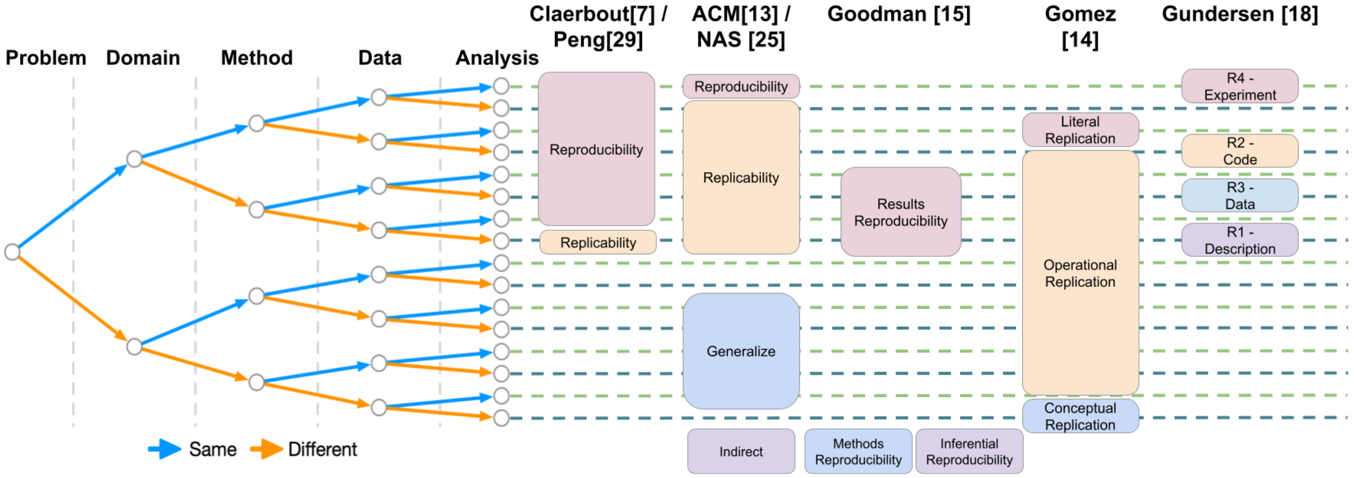


Figure 3: This figure shows where all of the previous definitions map within our framework. We can see that no framework covers every definition and that several definitions can mean similar things. For example, the NAS definition of replicability maps to several paths through the **Potential** Tree of Validity.

be swapped or removed. For example, in some experiments, it may make sense to move *data* before *method* or remove either layer. Thus, this formulation allows the ToV to be dynamic to the experiment it is describing. A ToV can be constructed for all experimental methodologies, and we show examples of applying the ToV in Section 5.

4.3 PEC

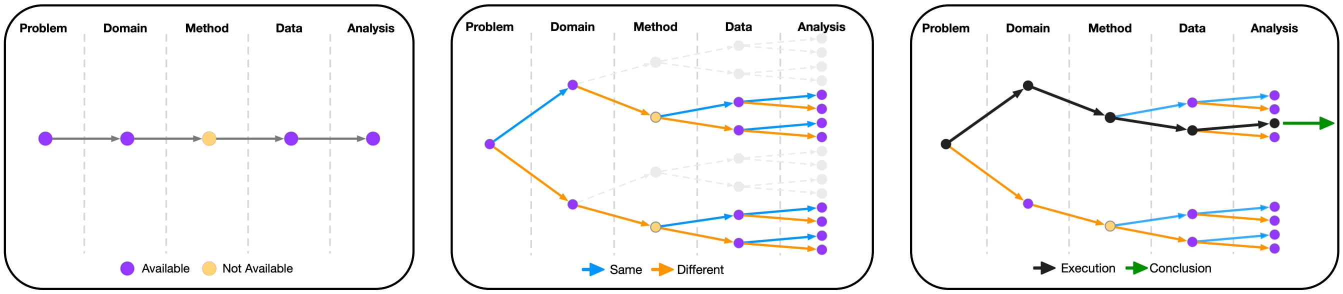
By formulating our framework around the Tree of Validity, we can now describe several properties of validity. We define the **Potential**, **Execution**, and **Conclusion** as aspects of the framework. **Potential** - Often reproducibility and replicability are inherently limited by what experimental artifacts are made available, a notion identified by Peng, the NAS, Goodman et al., and Gundersen. While some assign definitions or provide a hierarchy, they fail to provide a cohesive formulation that identifies the **Potential** for validity that each published paper inherently has. The Tree of Validity shows the **Potential** for each validity experiment. Figure 4 shows the construction of a Tree of Validity for an experiment. In practice, the authors may not make every experimental artifact available (e.g., a massive or private/sensitive data set). Figure 4a shows a shorthand way of describing the experimental artifacts available which we refer to as the Seed of Validity (SoV). We construct this SoV, but the SoV can be constructed by the original authors. We further discuss this in Section 6.

In Figure 4a, we see that the *method* is not available. This could be, for example, the system code to run an experiment. As it is not available, this limits the paths one could take through the tree. We show the **Potential** tree in Figure 4b which reflects the possible **Executions** a researcher could take to confirm validity. Since the *method* is not available, it is impossible to reproduce the experiment. This does not limit other forms of validity (e.g., replicability). For example, a validator could implement their own method on the same data and analysis.

Execution - The **Execution** of a validity experiment manifests as a path from the root node of the **Potential** tree to a leaf node. Thus, an **Execution** is the act of conducting a validity experiment. Figure 4c shows an execution of a validity experiment in black. We can now compare validity experiments as paths through the **Potential** tree. An execution that follows the path where every edge is the *same* as the original experiment is an execution of reproducibility. If two executions have the same path through the potential tree of validity and at least one edge of the path walks *different* experimental artifacts, then the executions are not inherently the same. For example, if two executions test the method and analysis in a *different* domain, the two domains do not have to be the same. Every execution describes an experimental methodology and, thus, can create its own Tree of Validity.

Conclusion - After executing a path through the **Potential** tree, the output results in a **Conclusion**. With varying experiments, this can result in a figure (e.g., a bar graph showing a comparison of results) or a number (e.g., relative differences in accuracies).

Our framework is conceptually simple, yet describes validity in a different way than previous definitions. As such, we can express how the availability of experimental artifacts affects the potential validity studies. Furthermore, we can describe previous definitions in terms of our ToV. An experiment's ToV describes the potential. We can modify the ToV such that it fits any experimental process. This framework provides a unified communication strategy between validity studies.



(a) Seed of Validity **(b)** **(c)**

Figure 4: This figure shows an example of constructing our framework around an experiment. (a) We see what is made available (e.g., problem, domain, data, and analysis) and what is not available (e.g., method). This shortened figure is referred to as the Seed of Validity (SoV). (b) We can construct the **potential** tree from (a). Since the method (e.g., experimental code) is not available it limits what end nodes we can reach. (c) An **execution** through the **potential** tree is a validity experiment that results in an outcome that we can draw **conclusions** from.

5 Case Studies

To demonstrate our framework, we provide several case studies from the Security Community. We present two types of case studies. First, we select two case studies from USENIX Artifact Award winners between 2022 and 2024. These case studies show how individual experiments can be turned into Trees of Validity. Second, we select two Systematizations of Knowledge (SoKs) from USENIX 2024 that replicate experiments from different papers in the same domain. We demonstrate that these experiments can be turned into multiple executions through a Tree of Validity.

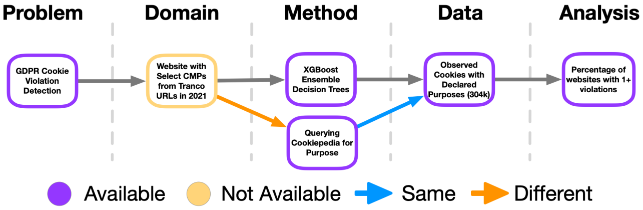

Problem
Domain
Method
Data
Analysis
GDPR Cookie Violation Detection
Website with Select CMPs from Tranco URLs in 2021
XGBoost Ensemble Decision Trees
Querying Cookiepedia for Purpose
Observed Cookies with Declared Purposes (304k)
Percentage of websites with 1+ violations
Available
Not Available
Same
Different

Figure 5: This figure shows the seed of validity for Bollinger et al.'s [5] Automating Cookie Consent and GDPR Violation Detection. It also contains a reduced representation of the baseline model as an execution through the Tree of Validity.

5.1 Artifact Awards

USENIX Security implemented an AEC in 2020 [6]. In 2022, USENIX Security awarded its first two distinguished artifact awards [5, 33], four awards in 2023 [19, 20, 24, 42], and five awards in 2024 [4, 8, 9, 11, 39]. These eleven papers represent the best-case scenarios for building complete Trees of Validity because all experimental artifacts are made available. Thus, for the first case studies we implement the Trees of Validity for each of these papers. We discuss two papers that were awarded distinguished artifact awards that provide interesting applications of our framework and provide all of the remaining Trees of Validity in Appendix A.

5.1.1 GDPR Cookie Violations

Bollinger et al. [5] automate GDPR violation detection of cookies. This work was awarded the 2022 Distinguished Artifact award and demonstrates several interesting aspects of our framework. We can see the seed of the tree of validity in Figure 5. Their methodology consists of collecting the websites from their domain and training an ensemble decision tree on the data. From this data, they map each cookie to a purpose and identify potential violations from the various policies (e.g., Google Analytics cookies not being labeled for analytics). In addition to their original experiment, they create a baseline model, which acts as an execution through the Tree of Validity. They construct a baseline model by querying Cookiepedia [27], which contains labels from manual analysis on the purpose of cookies. The execution of the baseline model only differs in the method.

One of the interesting aspects of this case study is the domain. The domain considers websites taken from the Tranco [31] ranking of May 5th, 2021 that use a consent management platform (CMP). As this paper considers the state of the internet in 2021, time is a part of the domain. The authors dutifully note that reproducing these experiments from scratch is infeasible, because of changes in the state of the internet. We run the experimental artifacts and find that 1,032 of the 6,940 (14.9%) of the URLs no longer resolve.[5] This is most likely due to websites no longer being hosted at the listed URL, if they even exist at all.

Several of the artifacts within the artifact repository demon-

[5] This is not out of six million as they limited the number of crawled websites in the available artifact to 6,940 to demonstrate the efficacy of the crawler.

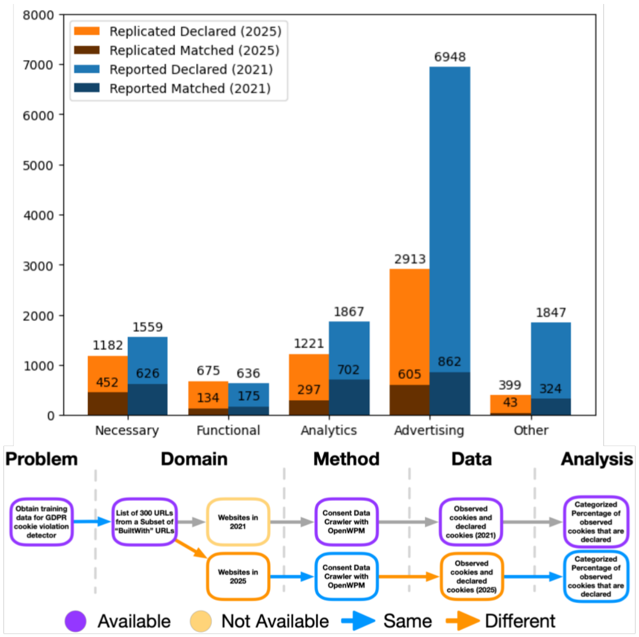


Figure 6: After running Bollinger et al.'s artifacts, we obtain less observed and declared cookies than their reported results due to updates to Cookie policies and websites no longer active. Beneath the graph is our walk through the ToV of this experiment.

strate the web crawler and collection on a sample. They provide the declared and matched cookie results for 300 URLs from 2021. We show the replicated data collection in January of 2025 in Figure 6. Of interest, are the functional and advertising cookies. The advertising cookies in our run show only 42% of the original declared cookies. In functional, we see a slight increase in this number. We show the path through Bollinger et al.'s SoV in Figure 6 underneath the graph. As such, reproducibility experiments conducted on their artifacts will only result in execution paths that start at an internal node. For example, we can train their model with the processed training data, but we cannot recollect the complete dataset.

While the potential ToV for this paper is theoretically possible, one will never be able to collect the data from this study again (i.e., practically impossible). The internet is a complex ecosystem that changes over time. The experimental process in Bollinger et al. cannot be recreated from start to finish as this changes. They discuss this in their experimental artifact and provide intuition on how the data collection will change over time.

Takeaway *Even when* **all** *experimental artifacts are made available, reproducibility is not always possible to execute. However, authors can proactively address where experimental methodologies will vary over confounding factors.*

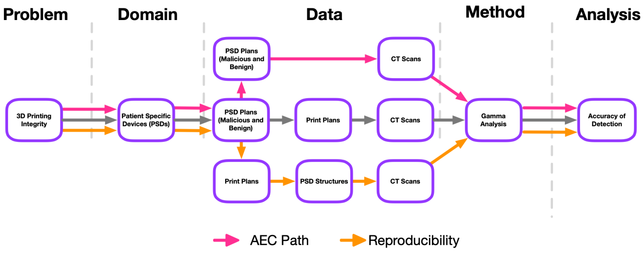


Figure 7: This figure shows the SoV for XCHECK by Yu et al. [42] and the execution path the AEC took to reproduce the results.

5.1.2 XCHECK

Yu et al. [42] proposed XCHECK which verifies the integrity of the 3-D printed patient-specific devices (PSD)s. They verify the integrity of the PSDs using various methods (e.g., gamma analysis). In their experimental artifacts, they provide all of the PSD plans and CT scans of the PSDs used in the study. Thus, a validator can run Yu et al.'s gamma analysis on the CT scans and PSDs to obtain their experimental results, but this is not full reproducibility. As seen in Figure 7, a validator would need to walk an execution through the ToV: using the PSD plans, print the PSDs; then scan the PSDs using a CT scanner, which results in usable CT scans; then the validator can apply the gamma analysis to identify malicious attacks. XCHECK was awarded a Results - Reproduced badge, but the execution path does not use the same collection methodology. Instead, it relies on the prior conducted data collection pathway. Thus, the execution of a validity experiment for the AEC starts at an internal node of the ToV.

The operation of the data collection methodology may affect the final results. This is confirmed in Yu et al.'s discussion section. For example, prior work [17] shows that there are optimal object orientations for CT scanning. Although XCHECK is cost-efficient compared to the cost of printing organs (e.g., millions of dollars), validating the full experimental process would be expensive for a validator that does not have access to the equipment. The validator would need to buy or rent a 3D printer and a CT scanner to run all of the associated experiments. Specialized testbeds (e.g., SPHERE [1]) do not currently have this expensive hardware.

Takeaway *The Security community's understanding of reproducibility is sometimes only a partial execution of a reproducibility pathway starting at an internal node of the ToV.*

5.2 Systematizations

Section 5.1 demonstrated how Trees of Validity are built from existing papers and their experimental artifacts. To show how our framework allows for comparison between replicability studies, we rely on SoKs that conducted several experiments around a central problem. USENIX Security introduced SoKs in 2024 [2] and published eight SoKs. We selected two SoKs that demonstrate replicating several . First, we consider deepfake datasets from Layton et al. [21] that iteratively run experiments to identify problems with the construction of deepfake datasets. Second, we demonstrate our framework on Stafeev et al. [36], where they systematize web crawlers. In their experiments, they run the largest experimental evaluation of web crawlers. Both of these papers demonstrate several executions through a tree of validity.

5.2.1 Deepfake Datasets

Layton et al. [21] conduct several experiments to identify existing problems in deepfake datasets. Although their systematization focuses on deepfakes in general, their experiments are targeted at audio deepfake detection. They present three research questions around audio deepfake datasets that are answered through experiments. First, are models built to detect audio deepfakes reproducible? Second, are the metrics representing the performance behavior of the models sufficient? Third, how does the current construction of the dataset affect the model performance? The third research question is answered in two parts: (1) by changing the training set and evaluating the performance and (2) by changing the evaluation set to a new domain and evaluating performance. This experimental setup can be modeled by a ToV and shows that each of these research questions is targeted at a subset of the ToV. Within their work, they highlight seven models across five experiments.

For succinctness, we highlight and construct the Tree of Validity from one of the audio deepfake detectors they use in the paper, RawNet2 [37]. Then, we define the executions of the three experiments using one of the other models, wav2vec [38]. There are five other models used in the paper, but these models would appear as the same execution path through the ToV as wav2vec. In Figure 8, we see the executions through the Tree of Validity that map to the research questions presented in their paper. The Problem is detecting audio deepfakes in the Domain of the ASVspoof2021 dataset [41]. As we build the Tree of Validity from the perspective of the RawNet2 model, the Method is the model of detection of the RawNet2 Model. The data it is trained on is the ASVspoof Train set and the ASVspoof Eval set to generate the Analysis metrics of EER, FPR, and TPR.

RQ1 They reproduce the RawNet2 model using the same experimental artifacts and walk the path where all of the experimental methodology remains the same. Thus, this execution walks the path of the upper-most, as visualized in Figure 8. When reproducing wav2vec, they walk the path where the only difference is the method. The other models that are reproduced would also take the same path as wav2vec.

RQ2 In this research question, they are assessing the performance of the models with a new analysis technique. Layton et al. compute the similarity between the models, shown as the execution from the root node to the leaf nodes of similarity.

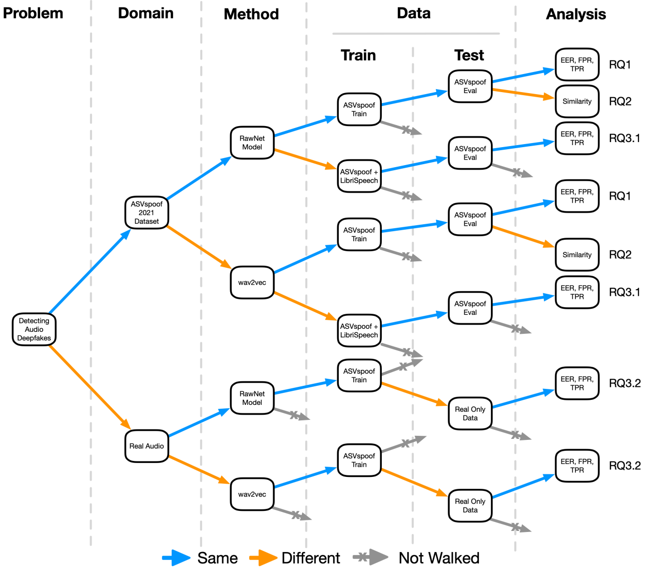


Figure 8: This figure shows the executions of the experiments Layton et al. [21] conduct. We model RawNet2 as the original experiment and wav2vec as the second model. The grey arrows represent sub-trees that are not a part of any execution conducted in the paper.

RQ3 The third research question tests the limitations within the datasets. As such, the two experiments that show this are (1) changing the training set and (2) changing the test set. To do this, they modify the train set with more real audio from Librispeech [28] for (1). This path manifests at two leaf nodes. The first is with the RawNet2 model on a different train set but the same test and analysis. The second is with wav2vec on a different train set but the same test and analysis. For (2), they show that the models are biased towards predicting deepfakes by only giving the models real audio. This changes the domain with which the experiment is conducted, and the two paths through the different domains are the executions for wav2vec and RawNet2 on the new domains.

We can apply our framework to these experiments and identify how the individual research questions differ experimentally. In previous frameworks identified in Section 3, RQ3.1 and RQ3.2 fall under replicability with no meaningful difference. While these frameworks would label RQ1 as reproducibility, they cannot differentiate between RQ2 and RQ3. In our framework, we visually see that there are differences between the research questions and can define the differences by the path taken from the root node to the desired leaf node.

SoKs map a space and provide validity studies. They show where Security research has been conducted. Therefore, by applying our framework to Layton et al., we can denote the

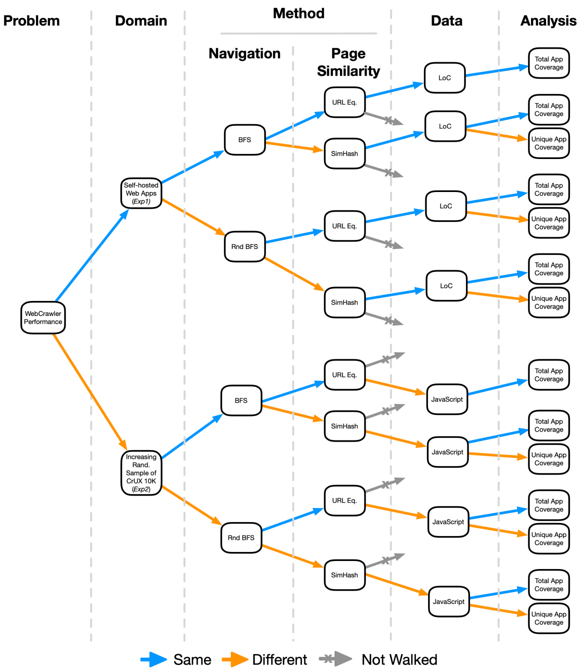


Figure 9: This figure shows the executions of the experiments in Stafeev et al. [36]. We model the baseline navigation and page similarity algorithm configuration (e.g., BFS and URL Eq.) as the Potential Tree of Validity and show the executions of the possible combinations with Rnd BFS and SimHash.

differences in the research questions and highlight how the different experimental methodologies affect the replicability of the results.

Takeaway *Our framework provides a clear distinction between the research questions and shows the potential for further replicability studies to improve the understanding of the problem area.*

5.2.2 Web Crawlers

Stafeev et al. [36] compare web crawlers that perform page navigation with a page similarity algorithm. The final tool, *Arachnarium*, implements seventeen-page similarity algorithms and six navigation strategies. There are over 100 possible combinations of similarity algorithms and navigation strategies. To reduce the load on real-world websites, they down-select by performing *Exp1* on self-hosted web applications and taking the ten best page similarity algorithms to perform *Exp2* on an increasingly large random samples from disjoint CrUX Top 10k [16] buckets. While each experiment could be mapped to this Tree of Validity, we demonstrate the baseline of Breadth-First Search (BFS) and URL Equal (URL Eq.) path navigation as the potential ToV. We show the combinations of BFS and URL eq. with Random BFS (Rnd BFS) and the SimHash navigation algorithm as shown in Figure 9.

Exp1 The first experiment implements the over 100 combinations of the seventeen-page similarity algorithms and the six navigation strategies. In this experiment, the data collected is the lines of code, and the code coverage is considered for the analysis. In the top path through the ToV, we see that the domain is self-hosted web applications, the method is broken up into the page navigation algorithm, BFS, and the page similarity algorithm, URL Eq. The total app coverage is considered for each combination. For every combination, a unique app coverage is calculated as a different metric to compare against.

Exp2 Once the top ten page similarity algorithms are identified, the second experiment runs the web crawlers on the increasingly-large random samples from disjoint CrUX Top 10k buckets. This experiment collects the unique JavaScript lines run by crawling the website and represents the bottom sub-tree of the root node. We note that more algorithms could be implemented within this ToV, but they would implement in the same paths as the *different* pathways.

From this case study, we see the numerous executions Stafeev et al. conducted in this work. Further, the analysis can be augmented to create new executions through the ToV to increase the coverage through the ToV. By proposing the framework in this way, we can reason not only about singular validity studies, but any number of validity studies.

Takeaway *Our framework allows a comparable analysis of multiple validity studies by providing a standardized approach.*

6 Discussion

6.1 Current Progress and Adoption

Our framework provides a communication pathway between authors and validators. The Security community introduced AECs to address growing concerns about reproducibility. Within our framework, this would be walking the top path of the Tree of Validity. Although USENIX is requiring the publication of experimental artifacts with the paper, this does not address the broader concerns of Validity. Reproducibility does not provide the validity of the experiments; merely, it shows that the results presented in published research were obtained within the associated artifacts. Due to complex domains or limitations in cost, validators cannot always execute the full reproducibility path and instead rely on the collected artifacts (e.g., measured data). Even in the best-case scenario, code does not always work. Systems are complex, and we need a language to discuss how our experiments can be affected. Our framework provides a communication path between original authors and validators. It presents a consistent way to communicate experimental methodologies.

Furthermore, our framework focuses on applying replicability standards to the Security community. While in complex systems correctness is difficult to show, we can provide evidence towards a conclusion. Our framework promotes exploration of the underlying hypothesis by multiple experimenters, incorporating a variety of differences (e.g., methods, data). By expressing experiments as ToVs, comparisons to prior work can be more precisely made and greater evidence towards the validity of a hypothesis can be achieved. In so doing, multiple efforts in an area can more naturally demonstrate progress in that space, and more clearly delineate novelty.

We recognize that the nodes within the SoV are subjective to the authors and validators. Fundamentally, it does not matter whether the choices made are subjective. Rather, if the original authors know the determinative factors in the experimental process and the validators agree, then *they are communicating in a consistent fashion*. If disagreements occur, this can be expressed by constructing new SoVs.

We encourage authors to include a SoV in the Appendix of their work. Modifying a pipeline figure, something most authors could include, can result in a detailed SoV. Further, they should identify areas that would cause variation in their results. For example, identifying time-dependent domains or sources of randomness. We expect that the SoV diagram can be expanded to include these variations. This expresses where the sources of variation occur. Further, conferences can include our framework in the publication process, and AECs can include this as a part of the artifact appendix. The additional figure presents limited difficulty for authors and a chance to argue for both the contribution (e.g., independent confirmation of another study) and novelty (i.e., a new contribution such as method or data) of their work.

6.2 Applications Beyond This Study

We applied our framework to a diverse set of Security papers, including audio deepfakes, cookie analysis, medical device integrity, and web crawlers. However, the framework is flexible and can provide benefits to areas outside of Security. Human-computer interaction can use our framework to express changes to experimental methodologies or specific populations involved in a user study. Machine Learning researchers can integrate this framework into their research to compare to similar work, especially in cases where randomness present within algorithms/hardware creates variable outcomes.

This framework is also applicable beyond computer science. For example, biological studies are often complex and not deterministic (e.g., population sampling). One could use this framework to model the experimental process and discuss where their SoV introduces variability and how that can affect future replicability studies. Similarly, areas such as psychology could account for factors not included in prior studies (e.g., stress, fatigue, motivation) using this framework.

6.3 Open Problems

Our work proposes a framework for replicability that can be adopted throughout the Security community and acts as a foundation for the field of meta-science within the Security community. As such, we identify several future work areas.

We define the conclusion as an outcome of an execution through the ToV, but we do not address what quantifies as a "successful" execution. While prior work [26, 32] assigns arbitrary thresholds (e.g., within 10% of the results), this is subjective and falls into fallacies identified within null-hypothesis statistical testing.[6] For example, there can be numerous reasons for results falling outside of 10% (e.g., time). Gundersen [18] proposes that if the interpretation of the results remains the same, then it is successful. This measure still remains subjective. Interpreting the results of validity experiments is an open and *difficult* problem, and future research should be conducted to identify these measures.

Security research often operates in controlled environments to limit disruption of real-world settings. An avenue for future meta-science research is quantifying the variability within from lab-controlled. Of particular interest is identifying avenues for Security research to transition into practice.

Finally, we see potential in identifying avenues for automating replicability analysis and studies. This approach is made easier by testbeds (e.g., SPHERE [1]), but broader implications exist within how these processes generalize to different environments. For example, one may be interested in testing environments outside the domain of their original experiments. The development of tools and techniques to automate this process can help create better mechanisms for researchers to make their work more transparent.

7 Conclusion

Reproducibility is a growing concern within the Security community and as such, conferences and authors are starting to address this. Yet reproducibility is only a small part of the broader field of validity. Replicability is the testing of the underlying hypothesis. In this work, we provide a systematization of the field of computational reproducibility and replicability. We then provide the first framework to unify definitions and address the limitations in prior work, specifically in how replicability is ill-defined. Our framework is flexible and conceptually simple, built around a binary tree of all of the available experimental processes, the Tree of Validity. This formulation allows consistent communication of experimental methodologies and comparisons of reproducibility and replicability studies, which we demonstrate through several case studies of award-winning artifacts and systematizations of knowledge from USENIX Security. We encourage future authors to adopt our framework in their research to promote open science goals.

[6] See the ASA's statement on p-values [40].

8 Ethics Consideration

We did not make any ethical considerations for our work, but there are ethical ramifications of our work. Security research often occurs in sensitive areas, such as privacy, surveillance, and data protection. Replicating a study can inherently affect the privacy and security of the studied area. For instance, a sensitive dataset may be kept from the public, making a reproducibility study impossible. This framework gives future authors a means of comparing their work in the absence of such a dataset.

9 Open Science

We provide all of the ToVs produced as part of this paper. Experimentally, we conducted one main experiment of replicability (e.g., analyzing cookie traffic). We will provide the code to conduct this with the submission, but note that it primarily relies on previous artifacts of submission. To calculate our end results we provide a script to run the collection and generate Figure 6.

References

[1] Security and privacy heterogeneous environment for reproducible experimentation. https://sphere-project.net/.

[2] Davide Balzarotti and Wenyuan Xu. Message from the USENIX Security'24 program co-chairs. In *33rd USENIX Security Symposium, USENIX Security 2024*, 2024.

[3] Lorena A Barba. Terminologies for reproducible research. *arXiv preprint arXiv:1802.03311*, 2018.

[4] Fabian Bäumer, Marcus Brinkmann, and Jörg Schwenk. Terrapin attack: Breaking SSH channel integrity by sequence number manipulation. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 7463–7480, 2024.

[5] Dino Bollinger, Karel Kubicek, Carlos Cotrini, and David Basin. Automating cookie consent and GDPR violation detection. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 2893–2910, 2022.

[6] Srdjan Čapkun and Franziska Roesner. Message from the USENIX Security'20 program co-chairs. In *29th USENIX Security Symposium, USENIX Security 2020*, 2020.

[7] Jon F Claerbout and Martin Karrenbach. Electronic documents give reproducible research a new meaning. In *SEG technical program expanded abstracts 1992*, pages 601–604. Society of Exploration Geophysicists, 1992.

[8] Giulio De Pasquale, Ilya Grishchenko, Riccardo Iesari, Gabriel Pizarro, Lorenzo Cavallaro, Christopher Kruegel, and Giovanni Vigna. ChainReactor: Automated privilege escalation chain discovery via AI planning. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 5913–5929, 2024.

[9] Gelei Deng, Yi Liu, Víctor Mayoral-Vilches, Peng Liu, Yuekang Li, Yuan Xu, Tianwei Zhang, Yang Liu, Martin Pinzger, and Stefan Rass. PentestGPT: Evaluating and harnessing large language models for automated penetration testing. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 847–864, 2024.

[10] Chris Drummond. Replicability is not reproducibility: nor is it good science. In *Proceedings of the Evaluation Methods for Machine Learning Workshop at the 26th ICML*, volume 1. National Research Council of Canada Montreal, Canada, 2009.

[11] Victor Duta, Mitchel Josephus Aloserij, and Cristiano Giuffrida. SafeFetch: Practical Double-Fetch protection with Kernel-Fetch caching. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 1207–1224, 2024.

[12] Association for Computing Machinery. Artifact review and badging - version 1.0 (not current), 2016.

[13] Association for Computing Machinery. Artifact review and badging - current, Aug 2020.

[14] Omar S Gómez, Natalia Juristo, and Sira Vegas. Understanding replication of experiments in software engineering: A classification. *Information and Software Technology*, 56(8):1033–1048, 2014.

[15] Steven N Goodman, Daniele Fanelli, and John PA Ioannidis. What does research reproducibility mean? *Science translational medicine*, 8(341):341ps12–341ps12, 2016.

[16] Google. Overview of CrUX. https://developer.chrome.com/docs/crux.

[17] Thomas Gregory, Ulrich Hansen, Monica Khanna, Celine Mutchler, Saik Urien, Andrew A Amis, Bernard Augereau, and Roger Emery. A CT scan protocol for the detection of radiographic loosening of the glenoid component after total shoulder arthroplasty, 2014.

[18] Odd Erik Gundersen. The fundamental principles of reproducibility. *Philosophical Transactions of the Royal Society A*, 379(2197):20200210, 2021.

[19] Sven Hebrok, Simon Nachtigall, Marcel Maehren, Nurullah Erinola, Robert Merget, Juraj Somorovsky, and Jörg Schwenk. We really need to talk about session tickets: A Large-Scaleanalysis of cryptographic dangers with TLSsession tickets. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 4877–4894, 2023.

[20] Jakob Koschel, Pietro Borrello, Daniele Cono D'Elia, Herbert Bos, and Cristiano Giuffrida. Uncontained: uncovering container confusion in the linux kernel. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5055–5072, 2023.

[21] Seth Layton, Tyler Tucker, Daniel Olszewski, Kevin Warren, Kevin Butler, and Patrick Traynor. Sok: The good, the bad, and the unbalanced: Measuring structural limitations of deepfake media datasets. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 1027–1044, 2024.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[23] Mark Liberman. Replicability vs. reproducibility - or is it the other way around? *Language Log*, 2015.

[24] Vivek Nair and Dawn Song. Multi-Factor Key Derivation Function (MFKDF) for Fast, Flexible, Secure, & Practical Key Management. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2097–2114, 2023.

[25] National Academies of Sciences Engineering and Medicine and others. *Reproducibility and replicability in science*. National Academies Press, 2019.

[26] Daniel Olszewski, Allison Lu, Carson Stillman, Kevin Warren, Cole Kitroser, Alejandro Pascual, Divyajyoti Ukirde, Kevin Butler, and Patrick Traynor. " get in researchers; we're measuring reproducibility": A reproducibility study of machine learning papers in tier 1 security conferences. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 3433–3459, 2023.

[27] OneTrust. Cookiepedia. `https://cookiepedia.co.uk/`.

[28] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[29] Roger D Peng. Reproducible research in computational science. *Science*, 334(6060):1226–1227, 2011.

[30] Hans E Plesser. Reproducibility vs. replicability: a brief history of a confused terminology. *Frontiers in neuroinformatics*, 11:76, 2018.

[31] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. Tranco: A research-oriented top sites ranking hardened against manipulation. *arXiv preprint arXiv:1806.01156*, 2018.

[32] Edward Raff. A step toward quantifying independently reproducible machine learning research. *Advances in Neural Information Processing Systems*, 32, 2019.

[33] Tobias Scharnowski, Nils Bars, Moritz Schloegel, Eric Gustafson, Marius Muench, Giovanni Vigna, Christopher Kruegel, Thorsten Holz, and Ali Abbasi. Fuzzware: Using precise MMIO modeling for effective firmware fuzzing. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1239–1256, 2022.

[34] Stefan Schmidt. Shall we really do it again? the powerful concept of replication is neglected in the social sciences. *Review of general psychology*, 13(2):90–100, 2009.

[35] Matthias Schwab, N Karrenbach, and Jon Claerbout. Making scientific computations reproducible. *Computing in Science & Engineering*, 2(6):61–67, 2000.

[36] Aleksei Stafeev and Giancarlo Pellegrino. Sok: State of the krawlers-evaluating the effectiveness of crawling algorithms for web security measurements. 2024.

[37] Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher. End-to-end anti-spoofing with rawnet2. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6369–6373. IEEE, 2021.

[38] Hemlata Tak, Massimiliano Todisco, Xin Wang, Jeeweon Jung, Junichi Yamagishi, and Nicholas Evans. Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation. *arXiv preprint arXiv:2202.12233*, 2022.

[39] Rafael Uetz, Marco Herzog, Louis Hackländer, Simon Schwarz, and Martin Henze. You cannot escape me: Detecting evasions of SIEM rules in enterprise networks. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 5179–5196, 2024.

[40] Ronald L Wasserstein and Nicole A Lazar. The asa statement on p-values: context, process, and purpose, 2016.

[41] Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, et al. Asvspoof 2021: accelerating progress in spoofed and deepfake speech detection. In *ASVspoof 2021 Workshop-Automatic Speaker Verification and Spoofing Coutermeasures Challenge*, 2021.

[42] Zhiyuan Yu, Yuanhaur Chang, Shixuan Zhai, Nicholas Deily, Tao Ju, XiaoFeng Wang, Uday Jammalamadaka, and Ning Zhang. XCheck: Verifying integrity of 3d printed Patient-Specific devices via computing tomography. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2815–2832, 2023.

A Appendix

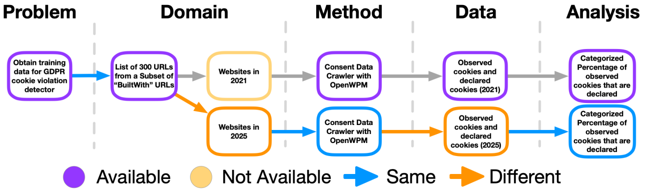

Problem
Domain
Method
Data
Analysis
Obtain training data for GDPR cookie violation detector
List of 300 URLs from a Subset of "BuiltWith" URLs
Websites in 2021
Websites in 2025
Consent Data Crawler with OpenWPM
Consent Data Crawler with OpenWPM
Observed cookies and declared cookies (2021)
Observed cookies and declared cookies (2025)
Categorized Percentage of observed cookies that are declared
Categorized Percentage of observed cookies that are declared
Available
Not Available
Same
Different

Figure 10: ToV for the paper "Automating Cookie Consent and GDPR Violation Detection," presented at the 2022 USENIX Security Symposium.

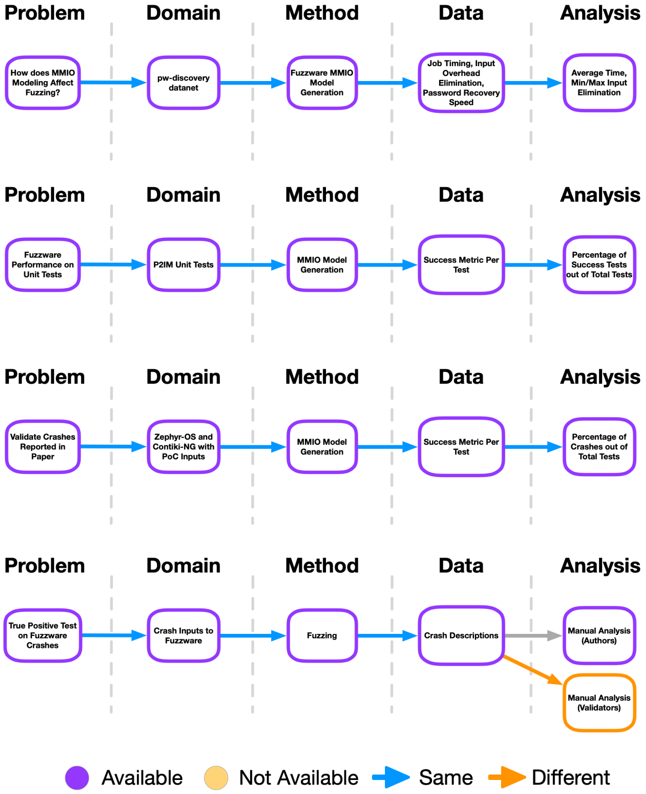

Problem
Domain
Method
Data
Analysis
How does MMIO Modeling Affect Fuzzing?
pw-discovery datanet
Fuzzware MMIO Model Generation
Job Timing, Input Overhead Elimination, Password Recovery Speed
Average Time, Min/Max Input Elimination
Problem
Domain
Method
Data
Analysis
Fuzzware Performance on Unit Tests
P2IM Unit Tests
MMIO Model Generation
Success Metric Per Test
Percentage of Success Tests out of Total Tests
Problem
Domain
Method
Data
Analysis
Validate Crashes Reported in Paper
Zephyr-OS and Contiki-NG with PoC Inputs
MMIO Model Generation
Success Metric Per Test
Percentage of Crashes out of Total Tests
Problem
Domain
Method
Data
Analysis
True Positive Test on Fuzzware Crashes
Crash Inputs to Fuzzware
Fuzzing
Crash Descriptions
Manual Analysis (Authors)
Manual Analysis (Validators)
Available
Not Available
Same
Different

Figure 11: ToV for the paper "Fuzzware: Using Precise MMIO Modeling for Effective Firmware Fuzzing," presented at the 2022 USENIX Security Symposium.

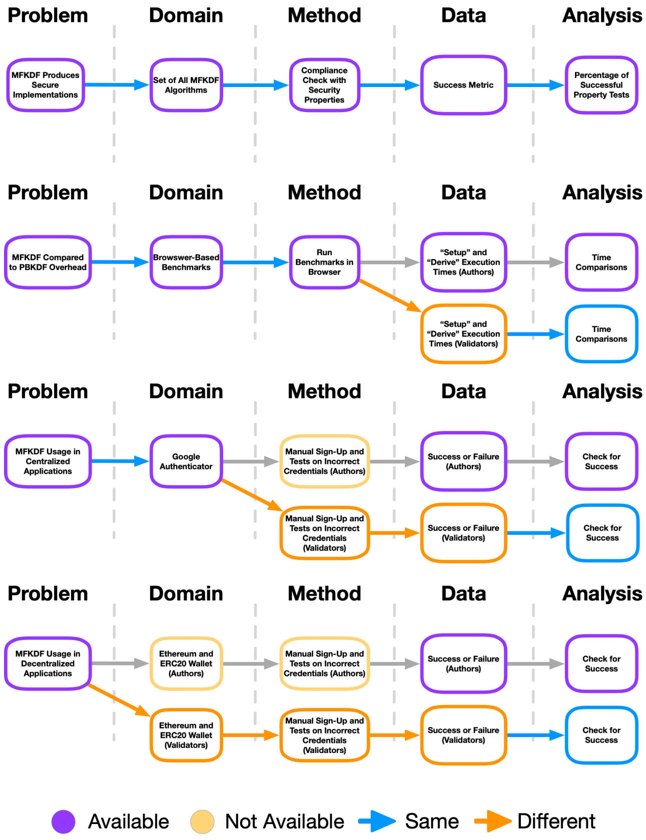

Problem
Domain
Method
Data
Analysis
MFKDF Produces Secure Implementations
Set of All MFKDF Algorithms
Compliance Check with Security Properties
Success Metric
Percentage of Successful Property Tests
Problem
Domain
Method
Data
Analysis
MFKDF Compared to PBKDF Overhead
Browswer-Based Benchmarks
Run Benchmarks in Browser
"Setup" and "Derive" Execution Times (Authors)
Time Comparisons
"Setup" and "Derive" Execution Times (Validators)
Time Comparisons
Problem
Domain
Method
Data
Analysis
MFKDF Usage in Centralized Applications
Google Authenticator
Manual Sign-Up and Tests on Incorrect Credentials (Authors)
Success or Failure (Authors)
Check for Success
Manual Sign-Up and Tests on Incorrect Credentials (Validators)
Success or Failure (Validators)
Check for Success
Problem
Domain
Method
Data
Analysis
MFKDF Usage in Decentralized Applications
Ethereum and ERC20 Wallet (Authors)
Manual Sign-Up and Tests on Incorrect Credentials (Authors)
Success or Failure (Authors)
Check for Success
Ethereum and ERC20 Wallet (Validators)
Manual Sign-Up and Tests on Incorrect Credentials (Validators)
Success or Failure (Validators)
Check for Success
● Available
● Not Available
Same
Different

Figure 12: ToV for the paper "Multi-Factor Key Derivation Function (MFKDF) for Fast, Flexible, Secure, & Practical Key Management," presented at the 2023 USENIX Security Symposium.

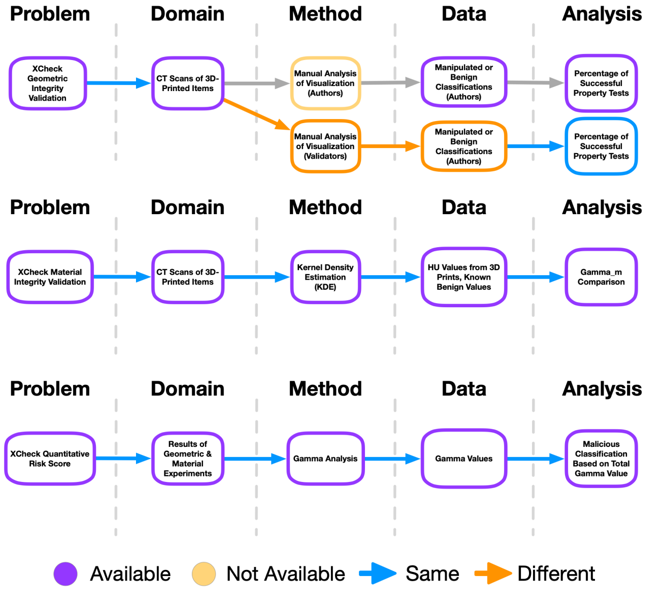

Problem
Domain
Method
Data
Analysis
XCheck Geometric Integrity Validation
CT Scans of 3D-Printed Items
Manual Analysis of Visualization (Authors)
Manipulated or Benign Classifications (Authors)
Percentage of Successful Property Tests
Manual Analysis of Visualization (Validators)
Manipulated or Benign Classifications (Authors)
Percentage of Successful Property Tests
Problem
Domain
Method
Data
Analysis
XCheck Material Integrity Validation
CT Scans of 3D-Printed Items
Kernel Density Estimation (KDE)
HU Values from 3D Prints, Known Benign Values
Gamma_m Comparison
Problem
Domain
Method
Data
Analysis
XCheck Quantitative Risk Score
Results of Geometric & Material Experiments
Gamma Analysis
Gamma Values
Malicious Classification Based on Total Gamma Value
● Available
● Not Available
Same
Different

Figure 13: ToV for the paper "XCheck: Verifying Integrity of 3D Printed Patient-Specific Devices via Computing Tomography," presented at the 2023 USENIX Security Symposium.

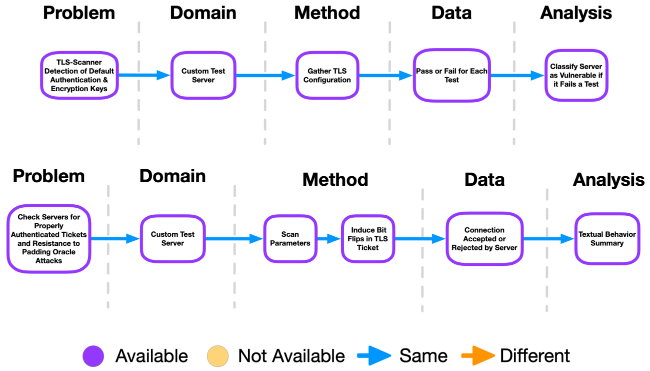

Problem
Domain
Method
Data
Analysis
TLS-Scanner Detection of Default Authentication & Encryption Keys
Custom Test Server
Gather TLS Configuration
Pass or Fail for Each Test
Classify Server as Vulnerable if it Fails a Test
Problem
Domain
Method
Data
Analysis
Check Servers for Properly Authenticated Tickets and Resistance to Padding Oracle Attacks
Custom Test Server
Scan Parameters
Induce Bit Flips in TLS Ticket
Connection Accepted or Rejected by Server
Textual Behavior Summary
● Available
● Not Available
Same
Different

Figure 14: ToV for the paper "We Really Need to Talk About Session Tickets: A Large-Scale Analysis of Cryptographic Dangers with TLS Session Tickets," presented at the 2023 USENIX Security Symposium.

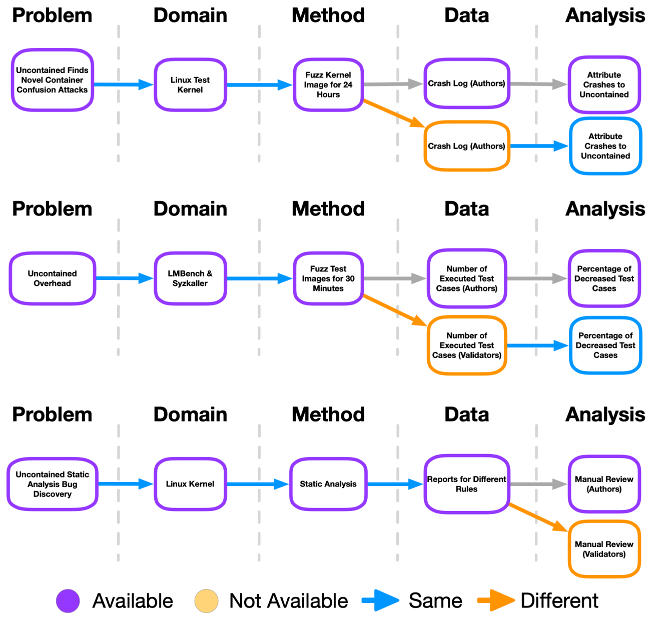

Problem
Domain
Method
Data
Analysis
Uncontained Finds Novel Container Confusion Attacks
Linux Test Kernel
Fuzz Kernel Image for 24 Hours
Crash Log (Authors)
Attribute Crashes to Uncontained
Crash Log (Authors)
Attribute Crashes to Uncontained
Problem
Domain
Method
Data
Analysis
Uncontained Overhead
LMBench & Syzkaller
Fuzz Test Images for 30 Minutes
Number of Executed Test Cases (Authors)
Percentage of Decreased Test Cases
Number of Executed Test Cases (Validators)
Percentage of Decreased Test Cases
Problem
Domain
Method
Data
Analysis
Uncontained Static Analysis Bug Discovery
Linux Kernel
Static Analysis
Reports for Different Rules
Manual Review (Authors)
Manual Review (Validators)
● Available
● Not Available
Same
Different

Figure 15: ToV for the paper "Uncontained: Uncovering Container Confusion in the Linux Kernel," presented at the 2023 USENIX Security Symposium.

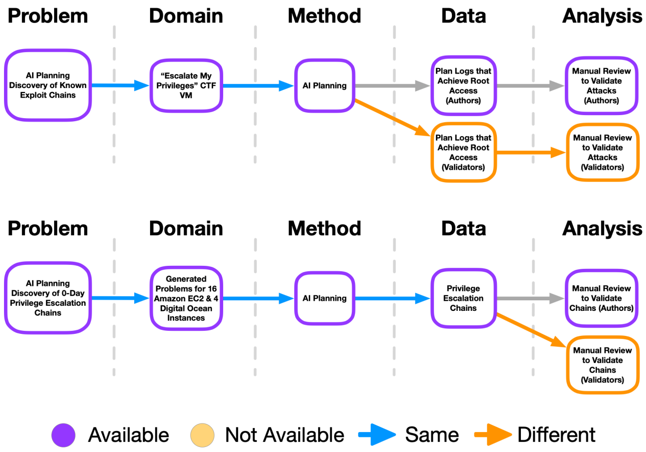

Problem
Domain
Method
Data
Analysis
AI Planning Discovery of Known Exploit Chains
"Escalate My Privileges" CTF VM
AI Planning
Plan Logs that Achieve Root Access (Authors)
Manual Review to Validate Attacks (Authors)
Plan Logs that Achieve Root Access (Validators)
Manual Review to Validate Attacks (Validators)
Problem
Domain
Method
Data
Analysis
AI Planning Discovery of 0-Day Privilege Escalation Chains
Generated Problems for 16 Amazon EC2 & 4 Digital Ocean Instances
AI Planning
Privilege Escalation Chains
Manual Review to Validate Chains (Authors)
Manual Review to Validate Chains (Validators)
Available
Not Available
Same
Different

Figure 16: ToV for the paper "ChainReactor: Automated Privilege Escalation Chain Discovery via AI Planning," presented at the 2024 USENIX Security Symposium.

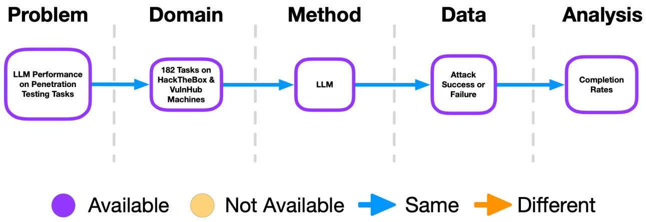

Problem
Domain
Method
Data
Analysis
LLM Performance on Penetration Testing Tasks
182 Tasks on HackTheBox & VulnHub Machines
LLM
Attack Success or Failure
Completion Rates
Available
Not Available
Same
Different

Figure 17: ToV for the paper "PentestGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing," presented at the 2024 USENIX Security Symposium.

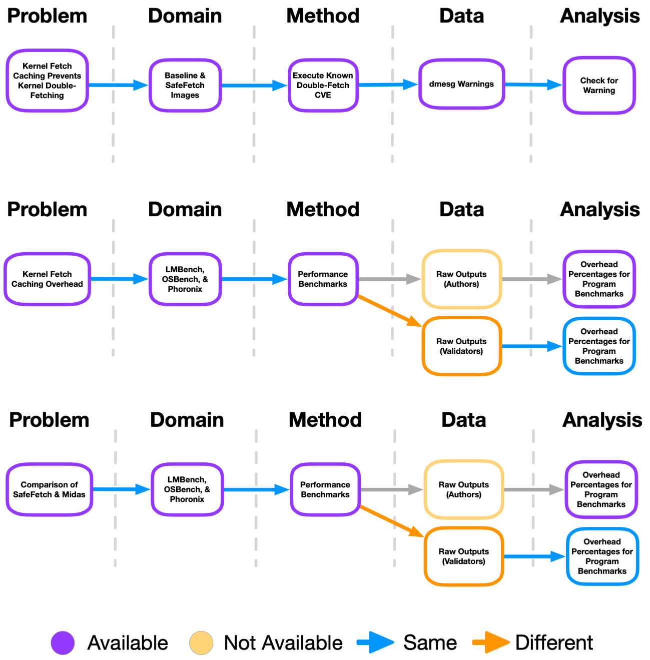

Problem
Domain
Method
Data
Analysis
Kernel Fetch Caching Prevents Kernel Double-Fetching
Baseline & SafeFetch Images
Execute Known Double-Fetch CVE
dmesg Warnings
Check for Warning
Problem
Domain
Method
Data
Analysis
Kernel Fetch Caching Overhead
LMBench, OSBench, & Phoronix
Performance Benchmarks
Raw Outputs (Authors)
Overhead Percentages for Program Benchmarks
Raw Outputs (Validators)
Overhead Percentages for Program Benchmarks
Problem
Domain
Method
Data
Analysis
Comparison of SafeFetch & Midas
LMBench, OSBench, & Phoronix
Performance Benchmarks
Raw Outputs (Authors)
Overhead Percentages for Program Benchmarks
Raw Outputs (Validators)
Overhead Percentages for Program Benchmarks
Available
Not Available
Same
Different

Figure 18: ToV for the paper "SafeFetch: Practical Double-Fetch Protection with Kernel-Fetch Caching," presented at the 2024 USENIX Security Symposium.

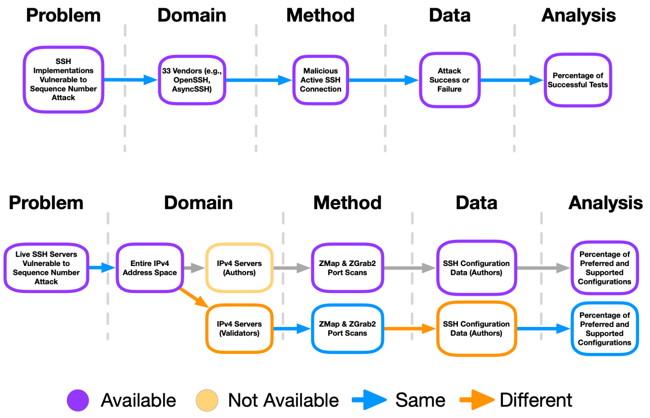

Problem
Domain
Method
Data
Analysis
SSH Implementations Vulnerable to Sequence Number Attack
33 Vendors (e.g., OpenSSH, AsyncSSH)
Malicious Active SSH Connection
Attack Success or Failure
Percentage of Successful Tests
Problem
Domain
Method
Data
Analysis
Live SSH Servers Vulnerable to Sequence Number Attack
Entire IPv4 Address Space
IPv4 Servers (Authors)
ZMap & ZGrab2 Port Scans
SSH Configuration Data (Authors)
Percentage of Preferred and Supported Configurations
IPv4 Servers (Validators)
ZMap & ZGrab2 Port Scans
SSH Configuration Data (Authors)
Percentage of Preferred and Supported Configurations
Available
Not Available
Same
Different

Figure 19: ToV for the paper "Terrapin Attack: Breaking SSH Channel Integrity By Sequence Number Manipulation," presented at the 2024 USENIX Security Symposium.

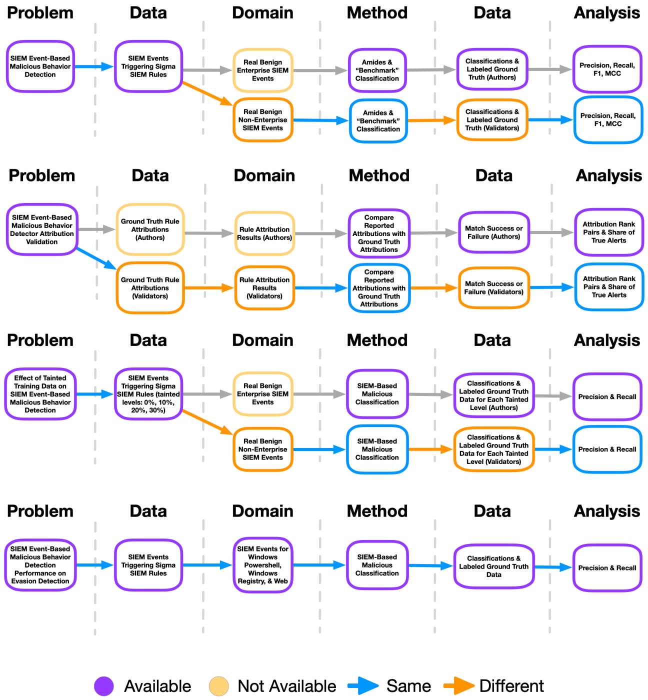

Problem
Data
Domain
Method
Data
Analysis
SIEM Event-Based Malicious Behavior Detection
SIEM Events Triggering Sigma SIEM Rules
Real Benign Enterprise SIEM Events
Amides & "Benchmark" Classification
Classifications & Labeled Ground Truth (Authors)
Precision, Recall, F1, MCC
Real Benign Non-Enterprise SIEM Events
Amides & "Benchmark" Classification
Classifications & Labeled Ground Truth (Validators)
Precision, Recall, F1, MCC
Problem
Data
Domain
Method
Data
Analysis
SIEM Event-Based Malicious Behavior Detector Attribution Validation
Ground Truth Rule Attributions (Authors)
Rule Attribution Results (Authors)
Compare Reported Attributions with Ground Truth Attributions
Match Success or Failure (Authors)
Attribution Rank Pairs & Share of True Alerts
Ground Truth Rule Attributions (Validators)
Rule Attribution Results (Validators)
Compare Reported Attributions with Ground Truth Attributions
Match Success or Failure (Validators)
Attribution Rank Pairs & Share of True Alerts
Problem
Data
Domain
Method
Data
Analysis
Effect of Tainted Training Data on SIEM Event-Based Malicious Behavior Detection
SIEM Events Triggering Sigma SIEM Rules (tainted levels: 0%, 10%, 20%, 30%)
Real Benign Enterprise SIEM Events
SIEM-Based Malicious Classification
Classifications & Labeled Ground Truth Data for Each Tainted Level (Authors)
Precision & Recall
Real Benign Non-Enterprise SIEM Events
SIEM-Based Malicious Classification
Classifications & Labeled Ground Truth Data for Each Tainted Level (Validators)
Precision & Recall
Problem
Data
Domain
Method
Data
Analysis
SIEM Event-Based Malicious Behavior Detection Performance on Evasion Detection
SIEM Events Triggering Sigma SIEM Rules
SIEM Events for Windows Powershell, Windows Registry, & Web
SIEM-Based Malicious Classification
Classifications & Labeled Ground Truth Data
Precision & Recall
● Available
● Not Available
Same
Different

Figure 20: ToV for the paper "You Cannot Escape Me: Detecting Evasions of SIEM Rules in Enterprise Networks," presented at the 2024 USENIX Security Symposium.