Thesis Project Plan

Jawad Zaher

Technical University of Denmark

1 Project description

Recently, there have been cases of students posting course exercises online on different platforms as well as their solutions as PDFs which is against the university policies and has the copyright of the professors/instructors in charge of the course. This issue has been causing cases of plagiarism, and copying of assignments with no way of knowing who posted those solutions on the platforms and even those platforms are taking a lot of time to address those issues and sometimes ignore them and now it is mostly upon the professor to find the necessary culprits in this case and keep the integrity of their course content intact. Such has been the case with mathematics course assignments and exercises since they don't change much and you can easily find their solutions online.

In this thesis, we will be exploring ways to watermark those solutions such that we get to know who posted those solutions online, this will involve a broad survey of existing tools and techniques as well as a survey of existing tools and publications to figure out different robust watermark methods. Then we will try to figure out how to embed them into the PDFs by modifying spacing in documents so that there can be hidden watermarks as well. We will also research techniques for watermarking Webpages and Markdown. After this survey, we will figure out which techniques to choose from.

The second part, which will be an enhancement to this problem would be to generate this watermark efficiently for many students in real-time, who would be downloading the assignments/exercises using the course platform. Then maybe pre-generate the files with it using some custom features on the platform and then integrate this as a plugin to the cloud-based course platform to make it available for professors across the different courses. Finally, we will ensure respecting user privacy and regulations regarding the handling of personal information in all these situations.

1.1 Background

1.2 Prior work

To-do

1.3 Problem statement

With the exponential growth of digital document sharing and dissemination, ensuring the integrity and authenticity of shared documents has become a critical

concern. In this era of digital information exchange, Portable Document Format (PDF) and LaTeX documents are extensively utilized due to their robustness and universality. However, these formats pose significant challenges in terms of information hiding, editing, and protection against various attacks such as print/scan attacks. Additionally, the need to encode watermarks in a manner that ensures robustness against unauthorized modifications while preserving the document's integrity further complicates the process. This thesis delves into the intricate nuances of protecting online document sharing through watermarking techniques, addressing the complexities associated with PDF specifications, information hiding, editing constraints, and protection against malicious attacks.

The proliferation of online document-sharing platforms has revolutionized the way information is disseminated and accessed. However, this convenience comes with inherent risks such as unauthorized distribution, tampering, and plagiarism. PDF and LaTeX documents, being the cornerstone of academic and professional communication, require robust protection mechanisms to safeguard against these threats. The primary challenges lie in embedding imperceptible watermarks that can withstand various manipulations and attacks while preserving the document's readability and integrity. Moreover, the ability to detect and mitigate unauthorized modifications, including space manipulation and pointer updates, is paramount in ensuring the document's authenticity.

PDF specifications serve as the foundation for document rendering and manipulation. Understanding these specifications is crucial for devising effective watermarking strategies that seamlessly integrate with existing document structures. Information hiding techniques play a pivotal role in embedding watermarks within PDF and LaTeX documents without compromising their visual fidelity or structural integrity. By leveraging the intricacies of document encoding and compression algorithms, robust watermarking schemes can be devised to withstand common attacks while maintaining a low perceptual impact.

Editing a PDF poses significant challenges due to its inherent structure, which encapsulates textual, graphical, and metadata elements in a compressed format. Modifying PDF content without disrupting the document's layout or metadata requires specialized tools and expertise. Moreover, encoding watermarks in a robust manner involves carefully selecting embedding locations, adjusting embedding strengths, and incorporating error correction mechanisms to enhance resilience against manipulations. Robust watermark encoding ensures that even after multiple transformations and conversions, the watermark remains intact and detectable.

Protecting against print/scan attacks, where a document is printed and scanned to bypass digital safeguards, necessitates proactive measures. By embedding watermarks with perceptually invisible patterns and employing robust detection algorithms, print/scan attacks can be thwarted effectively. Furthermore, detecting and mitigating space manipulation and pointer updates require sophisticated analysis techniques that discern legitimate modifications from malicious alterations. By employing cryptographic hashes or digital signatures, the

integrity of document pointers can be verified, ensuring that any updates are authenticated and authorized.

Requirements
The following are the requirements of a formal model verified in this work.

Security Requirements
The following is the security requirement that the model must verify to provide the intended user protection.

RS: Confidentiality of the data on the server. No one except the data provider can infer anything about the data other than its length.

This means that information about the data cannot be obtained even with access to the server operating system and software that performs the computation on it. In addition, an adversary can not also perform any side-channel attack successfully to obtain any information about the data. Finally, the computed data returned to a user will not contain any data belonging to other server users.

Functional Requirements
The following is the functional requirement the model must satisfy to be valid.

RF: the model should also prove the correctness of the functionality of units employing the BliMe architecture.

Engineering Requirements
The following is the engineering requirement that the model needs to satisfy to be helpful in the future and upon future extensions.

RE: the model should be modular and extendable so that other extensions in the future can be added to the model.

Threat model
We consider an adversary that can possess complete control over the server's software and the server's peripheral devices, such as IO devices.

The software on the mentioned server includes the software performing computation over the user data and the operating system running on this server.

For instance, this level of control enables the adversary to infer information from the timing and patterns of memory access.

However, similar to the adversary described in the BliMe paper, side-channel attacks that require physical access to the server's main components are not part of the adversary's capabilities. Therefore, methods like differential power analysis fall beyond the security requirements of the model.

Goals
The following are some of the goals that this work aims to achieve by providing the requirements described for each goal. Goals four and five are stretch objectives that we will pursue if time permits.

G0: Extending the model to cover confidentiality of data in regards to information flows that leak data of one client to other clients

G1: Extending the model to define a matrix multiplication accelerator utilizing BliMe architecture

G2: Extending the model to a system with a generic peripheral design that includes a system bus shared between multiple components

G3: Extending the model to a system with multiple CPUs, accelerators, and peripherals

G4: Extending the model to a pipelined CPU and showing safety against micro architectural side-channel attacks

G5: Extending the model to include the encryption engine component

Contributions

The thesis aims to expand the F* model of BliMe from a simple CPU model to a more generic system. This expansion includes:

Extending the model to systems with multiple CPUs and Accelerators, ensuring that the security properties verified by F* remain valid in this context.

Extending the model to a system where CPU communicates with peripherals and memory mapped IO devices in addition to main memory.

Introducing the concept of clock cycles to the model to change the model from a single cycle fetch-execute model to a model that instructions take multiple cycles.

Impact

The resulting impact of the thesis will be to enhance the formal model of the Blinded Memory architecture to advance the security of outsourced computing and prove that confidentiality requirements will be valid under multiple computation units in a Blinded Memory architecture.

2 Intended learning objectives

- Formal verification of security properties
- Defining confidentiality as a formal property
- Modeling and verification of hardware components
- state-of-the-art verification tool F*
- Functional programming
- Project management, including planning, design, implementation, documentation, and presentation

3 Plan

To plan a timeline for the thesis, I have defined some tasks to help achieve the mentioned goals. Figure 1 shows the planned timeline for achieving these tasks and writing the thesis. The defined tasks are the following:

T0: Modifying the safety definition of the model to domain-wise safety

T1: Implementing a matrix multiplication accelerator utilizing BliMe architecture

T2: Verifying that the matrix multiplication accelerator has the defined requirements

T3: Adding a system layer to the model to utilize multiple CPUs scheduled in a round-robin format

T4: Designing a more generic model of a system with a generic definition of the state of CPU and peripherals that includes a system bus, memory-mapped IO devices, and main memory

T5: Modifying the existing model to utilize the new, more realistic model

T6: Adding peripherals other than main memory

T7: Defining Accelerator as one of the peripherals in the system

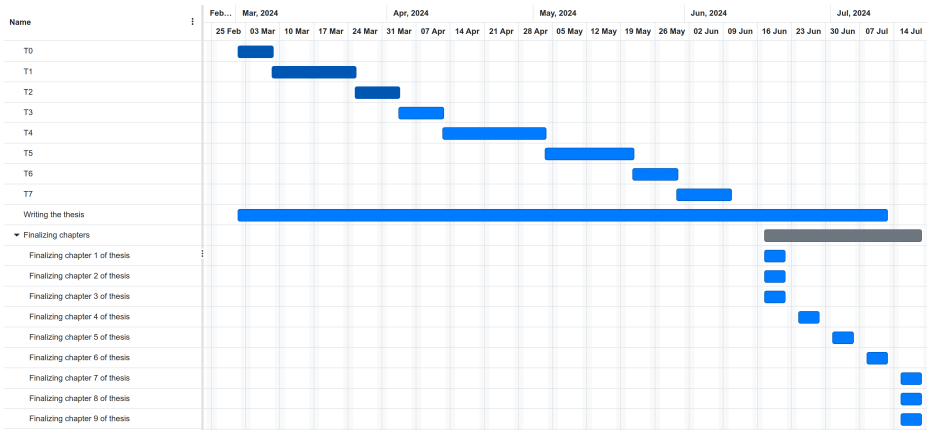

Fig. 1. Planned timeline for achieving the defined tasks of the thesis

References