ENS 491 – Graduation Project (Design)

Proposal

Project Title: Reproducing and Analyzing Side Channel Attacks

Group Members:

Ali Parlakçı (28114), Doğukan Yıldırım (28364), Rana İşlek (27836)

Supervisor(s): Cemal Yılmaz

Date: 11.12.2022

1. ABSTRACT

Side-channel attacks exploit the internal mechanics of computers. In this project, we first replicate existing state-of-the-art side-channel attacks and analyze system interrupts while the user of the computer is browsing the web, watching a video, playing video games or doing a computation intensive task. We then develop and implement a side-channel attack that uses an interrupt counting approach and runs solely on the GPU. Through the use of machine learning models, we try to determine user behavior. Earlier studies managed to make successful predictions on which web page the victim was browsing or if they are watching a video. We further improve these studies by leveraging web technologies and creating more powerful attacks on browsers.

2. INTRODUCTION

Side-channel attacks are security exploits which try to gather information from the medium. To put into context, an attack where an attacker steals someone's credit card PIN via the fingerprints on the POS machine is a side-channel attack. The vulnerability is not produced by a flaw in the procedure but the whole nature and the fundamental design of the procedure. In a similar sense, attackers get advantage of certain characteristics of computers and how computers function. Although it seems computers can do multitasking and execute tasks in parallel, they can only execute one task at a time. However, they switch between tasks so fast that it creates the illusion of parallelism. Interrupts are one of the reasons that will make computers switch. When an interrupt occurs, the operating system goes to kernel mode, pausing every other task and processes the interrupt. Interrupts usually happen due to an outside effect such as a newly received network packet. This switch of tasks is called context switches.

Commented [1]: indent

An earlier study kept track of the interrupts in computers indirectly. They run a simple loop-counter in a browser tab and read the value periodically. Meanwhile, they loaded different websites in another tab of the browser. Since computers execute tasks linearly, as the browser loads the website, interrupts occur. Due to the processing of the interrupts, the counter in the other will be different in each period. A study shows that when the counter values are fed to a

machine learning model, it can predict which websites are open in the browser by using the increment values in the browser (Cook et al., 2022).

Another study analyzed the interrupts which the graphical processing unit (GPU) of a computer created on the central processing unit (CPU) of the computer. They, however, used static records of these interrupts which are saved by the operating system. Again, with the use of machine learning models, they were able to tell if the user was either viewing a pdf document, watching a video, or running a computing intensive application on their GPU with high accuracy rates. They were even successful at what type of pdf document the user was viewing or what type of codec the video user was watching (Ma et al., 2022).

3. PROPOSED SOLUTION AND METHODS

Earlier studies show that side-channel attacks can be highly successful at extracting information about a victim's sensitive data. The a~~A~~im of this project is to understand the extent of the threat of side-channel attacks which leverage system interrupts and try to come up with mitigation techniques if there are any. In a world where personal information and privacy of individuals are amongst the most valuable, a potential source of information leak can have devastating consequences. It will benefit the security computers and the privacy of individuals as it will unearth the potential risks and let major browser developers and GPU manufacturers implement techniques to avoid these risks.

In this project, we will reproduce the loop-counting attack and the GPU interrupt analyzer attack. After verifying the credibility of the attacks, by combining the two studies, we investigate if the loop-counter attack can also identify the tasks which require GPU workforce. If they are, we will try to execute this attack in the browser and test it in different contexts to understand the practicality of the attack.

Solving this problem of security and data privacy is a complex problem in the context of multiple points; involves engineering constraints as we are limited with the boundaries of modern

computers. We are trying to find different ways to exploit the internal mechanics of computers to extract information about user behaviors.

It has no straight-forward solution since~~as~~ system interrupts operate at the lower levels of computers, ~~-~~and it requires indirect solutions and research-based knowledge about operating systems and the nature of interrupts, to analyze them, understand the information it carries about the user, if there is any, and develop an analytic approach.

Commented [2]: Shorten these sentences

The project also contains multiple parts. It requires web development for creating the spy website to snoop on internet users, GPU programming to analyze if the user is running a graphical related task and appropriate machine learning models to extract sensible information from the collected data.

As stated before, privacy is becoming the most valuable asset of individuals. Advertisements are based on private information. Side-channel attacks generate information about the behaviors and habits of people. The project has economical consequences on the online shopping industry. Also, since this information can also determine the political views of individuals, it can be used in a harmful way which can affect political situations in countries during elections.

3.1. **Objectives/Tasks**

The list of the tasks of the project with their respective objectives are as follows:

1. Read the existing literature about side channel attacks that make use of a web browser tab or the GPU, specifically the papers There's Always a Bigger Fish: A Clarifying Analysis of a Machine-Learning-Assisted Side-Channel Attack (Cook et al., 2022), and On the Effectiveness of Using Graphics Interrupt as a Side Channel for User Behavior Snooping (Ma et al., 2021).

a. The objective of this task is to make sure everyone involved with the project has gained a basic understanding of the existing research before proceeding with the project.

b. This task was assigned to all group members.

2. Clone the There's Always a Bigger Fish (Cook, 2022) GitHub repository mentioned in the research paper, run and test the source code with default parameters given in the Wiki page of

the repository to replicate the website-fingerprinting side-channel attack which makes use of a malicious web browser tab, visualize the data obtained by running the code with Python data visualization libraries, and train the default machine learning models provided in the repository with the data and test their accuracy.

a. With this task, the intended result is that each project member has become familiar with the source code of There's Always a Bigger Fish, has verified that the source code works as intended on their personal or virtual computers, and if not working as intended, has taken notes of the problems in code and/or data.

b. This task was assigned to all group members.

3. The data, which represent the counter values in a single run, obtained by running the source code of There's Always a Bigger Fish (Cook, 2022) has erroneous values in it. Counter values should be starting from "0", however, due to an error, some of the counter values are being recorded as "-1". The erroneous values can still be representing a behavior which can be classified with a machine learning model. Run the source code that can generate "-1" as counter values, store the data, visualize the data and train machine learning and deep learning models with it. Try visualizing the data and training AI models with it after removing "-1" values from it. Compare the heat maps, and AI model accuracies of both approaches.

a. The objective of this task is to observe if "-1" values have a drastic effect on AI model test accuracy and see if there can be any improvement made by removing "-1" values from the data.

b. Finding the root cause of the issue and fixing it in the source code could also be another approach to try.

c. This task was assigned to all group members. Task leader was Ali Parlakçı.

4. There is a web page available which acts as an interactive demo for the There's Always a Bigger Fish (Cook et al., 2022). The same traces, in other words the data which has counter values in it of the specific website given as a parameter, can be obtained from the interactive demo. Unlike the provided source code, the data obtained from the interactive demo does not have any "-1" values. Obtain traces of the same websites used for the previous task, visualize the data and train machine learning and deep learning models with it. After that, compare the heat maps and AI model accuracies of all three approaches.

a. The intended result of this task is to reach a conclusion to decide on with which implementation the project will proceed: The source code written in Python that may produce erroneous results, or the website written in JavaScript which acts as a malicious website.

b. This task was assigned to all group members. Task leader was Ali Parlakçı.

5. In task 3, option b., i.e. fixing the source code of the research paper was chosen and the source code was fixed and pushed to our own code repository. With this code, test different media applications such as image viewers and video players to see if we are able to differentiate between which process is actually running with ML (Machine Learning) models. Testing will be done in such a way that 15 second traces will be taken with a Chrome web browser tab acting as a malicious tab, and during that time period a GPU related application will be opened, run and closed. Then, the traces will be visualized and fed to a ML algorithm.

a. Test different GPU related applications, i.e. image viewers or video players.

b. Test different video players for the same video codec. An example test is opening the same .mp4 file with VLC, Windows Media Player and Media Player Classic and playing the video while capturing 15 second traces.

c. Testing different codecs for the same video player. An example test is opening media files with different codecs (files with .mp4, .flv, .mkv or .3gp extensions) with the VLC media player and playing the video while capturing 15 second traces.

d. This task was assigned to all group members. Task leader was Ali Parlakçı.

6. Implement a loop counting side channel attack similar to the attacks mentioned in both papers which targets the interrupts sent to the CPU by the GPU.

a. The intended result of this task is that with this newly implemented attack, the GPU-intensive applications the user is currently running can be classified with a good accuracy.

b. This task will be assigned to all group members.

7. Write a loop counting side channel attack that will be run on the GPU, while the previously mentioned attacks were running on the CPU, using CUDA programming language and NVIDIA GPUs.

a. The intended result of this task is that with this newly implemented attack, the GPU-intensive applications the user is currently running can be classified with a good accuracy.

b. This task will be assigned to all group members.

Due to the nature of the project and the fact that future tasks are determined with the progress and results of the previous tasks, there may be more tasks added to this list.

3.2. **Realistic Constraints**

Since our project is a software development/computer engineering project, time, scope, and cost (economics) are the three most crucial constraints that are dependent on each other and also affect the quality of our final product. But also there are other regular realistic constraints as mentioned below:

- Economic: We do not have any economic constraints for our project. No target price is mentioned for the cost, but that includes the material resources including the money spent on the project to be delivered on time according to the predetermined scope. That cost covers financial investments in software, labor, quality control, and tools.
- Environmental: There are no environmental constraints for our project.
- Health and safety: This is a project with which we work remotely, therefore there are no health and safety constraints.
- Social: In the duration of the whole project, we care about each team member and treat each other with respect and in a fair way.
- Manufacturability: Since we are developing a software, there exists no material thing that we aim to manufacture, therefore, we do not have any manufacturability constraints at the current stage of the project.
- Sustainability (social, economic, and environmental): We aim to store the code of our implementation in a remote GitHub repository for sustainability purposes and version control so that our code can be accessed from anywhere in the world, at any time.
- Scope: Scope covers all the features of the software products as well as the project's tasks and goals which need to be completed in a restricted time. It is important to stay in the boundaries of the scope of the project to be focused on the main topic; that's why in our project

we focus on reproducing side-channel attacks by checking several URLs and GPU/CPU processes to understand the behavior.

- Time: Time is a realistic constraint for this project because the schedule of the project and the deadlines for which each step of the project should be completed is critical because this is part of an ongoing research project.

3.3. **Engineering/Scientific Standards**

Since our project is a software/computer engineering project we need to satisfy some business-critical and safety factors:

- The internal structure of a software product can be measured using the standard code "ISO/IEC 5055:2021," which is an ISO standard, on four business-critical factors: reliability, security, maintainability, and performance efficiency. These are the elements that establish a software system's dependability, trustworthiness, and resilience.

We follow the best software development practices, publish and maintain our source code in a GitHub repository and write Wiki pages on the GitHub page as documentation. Furthermore, as a team, we meet weekly with our project supervisor to discuss our current progress and determine new tasks to accomplish.

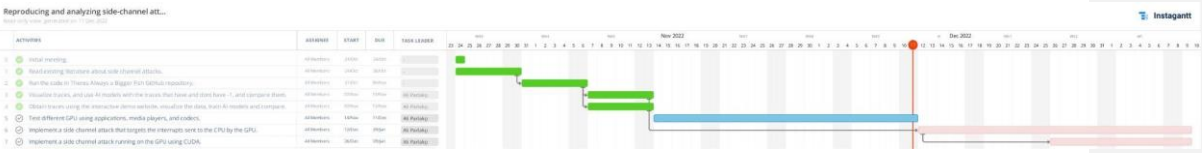
4. **RISK MANAGEMENT**

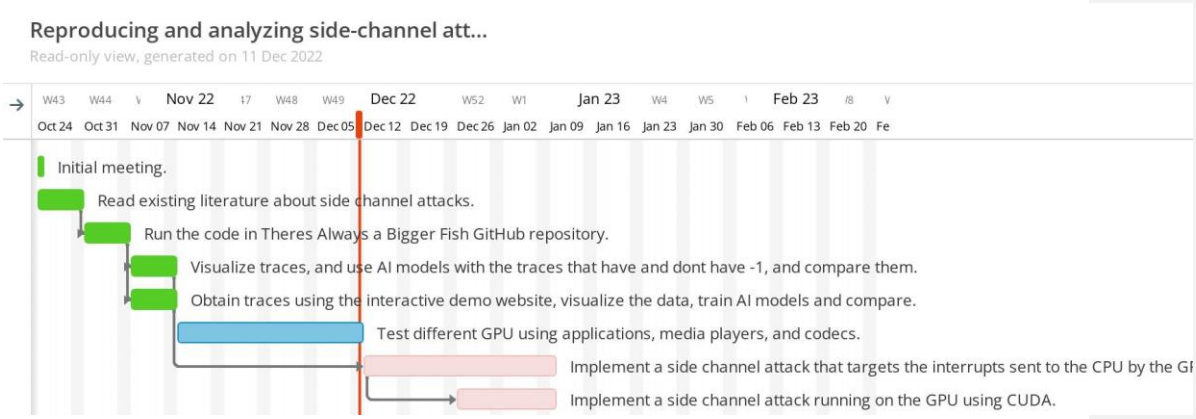
Every software development project carries risks, thus it's crucial to take them into account during the development process. Risks are typically taken into account by project managers as they create the project's final estimate and in this project our project manager is our project supervisor.

- If our knowledge and expertise won't be enough for the requirements of the project, then we need to help this situation with several options. We can ask for help from our supervisor to make things clearer or recommend a way to go. Furthermore, we can use other educational sources such as online courses and sites to gain more experience and get help.

- If we need more time than expected to finish our project by deadline, there are two options to fix this problem: either we can ask our supervisor to reduce some of the responsibilities we have and focused on what we have done, nor we can try to redivide the team responsibilities to according to each teammate's expertise.

5. PROJECT SCHEDULE





The following Gantt charts represent the project schedule:

Delays in any of the tasks will result in the delay of the successor task due to the fact that the tasks depend on one another. Our team makes use of two specific methods in order to eliminate delays as much as possible:

- The tasks are defined as clearly and granularly as possible with a concise definition of done. Smaller and clearer tasks make the planning process more easier, efficient and less prone to delay.
- As a team, we are in constant communication and meet twice on a weekly basis to plan the subtasks and assign someone to them. If there is a sudden workload availability issue of one

of our team members, we can reassign the task to any of the available project members in one of the meetings.

6. ETHICAL ISSUES

In today's general computing platforms, there may be an information leakage through any computational process. As demonstrated in There's Always a Bigger Fish: A Clarifying Analysis of a Machine-Learning-Assisted Side-Channel Attack (Cook et al., 2022), and On the Effectiveness of Using Graphics Interrupt as a Side Channel for User Behavior Snooping (Ma et al., 2021); the state-of-the-art side-channel attacks can exfiltrate secret information and train AI models with that information to detect which website a user is visiting, or what applications the user is running, just with a simple for-loop.

This is a serious breach of security and privacy, which could also have legal implications. Moreover, a simple, carefully, and well-designed malware can abuse these types of side-channel attacks, train AI models specific to the affected users and send the mass obtained data to the attacker. These kinds of big data might be sold on the black market for purposes we may not be able to foresee.

Because of these reasons, it could be a great idea to think about and propose practical defense mechanisms for the side-channel attacks we are developing and implementing.

7. REFERENCES

Cook, J., Drean, J., Behrens, J., & Yan, M. (2022). There's always a bigger fish: a clarifying analysis of a machine-learning-assisted side-channel attack. *Proceedings of the 49th Annual International Symposium on Computer Architecture*., 204-217. https://doi.org/10.1145/3470496.3527416

Cook, J. (2022). There's Always a Bigger Fish. *GitHub Repository*. Retrieved from https://github.com/jackcook/bigger-fish

Cook, J., Drean, J., Behrens, J., & Yan, M. (2022). There's Always a Bigger Fish Interactive Demo. Retrieved from https://jackcook.github.io/bigger-fish

Ma, H., Tian, J., Gao, D., & Jia, C. (2021). On the effectiveness of using graphics interrupt as a side channel for user behavior snooping. *IEEE Transactions on Dependable and Secure Computing*, 1-14.

Software quality standards – ISO 5055. CISQ. (2022, October 4). Retrieved November 13, 2022, from https://www.it-cisq.org/standards/code-quality-standards/