Improving Schedule Indistinguishability in Real-Time Systems

Debopam Sanyal
School of Computer Science
Georgia Institute of Technology
Atlanta, GA, USA
dsanyal7@gatech.edu

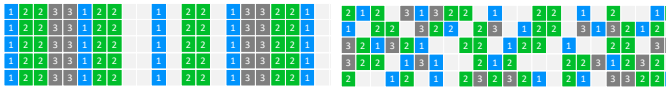
***Abstract*—The current definition of schedule indistinguishability only accounts for randomized mechanisms that satisfy ϵ-indistinguishability. This restricts the search space of mechanisms that we can apply to protect real-time systems (RTS). Hence, there is a need for a generic definition of schedule indistinguishability in terms of the statistical difference between the two adjacent input distributions (schedules in RTS). Not only does this allow for more applicable indistinguishability-preserving mechanisms, but also it provides the potential of giving stricter bounds on the protection duration of a defense mechanism and the advantage of a timing-based side-channel attack.**

***Index Terms*—differential privacy, real-time systems, schedule indistinguishability, side-channel attacks**

I. Introduction

Real-time systems (RTS) are ubiquitous in modern technology like autonomous cars, robots, drones and medical devices. A real-time scheduler is vital for such systems to function because important tasks have to meet timing requirements. A periodic design and a guarantee of meeting deadlines of tasks help RTS to maintain their utility. Security is an important issue in RTS as any breach could cause disastrous effects in safety-critical applications that predominantly employ RTS [1], [2]. An ever-increasing number of attacks on RTS [3]–[5] show the need to focus more on their security. Since RTS have to meet time-sensitive requirements, their implementation often becomes predictable. Many tasks arrive periodically and have to be executed before their respective deadlines, hence giving rise to the predictability as shown in Figure 1(a).

While the predictability may help in maintaining real-time guarantees, it gives rise to timing side-channels that can leak some critical information about the system. In general, there are many side-channels like power consumption [6] and temperature [7], but we focus on timing-based side-channels. An adversary that observes the schedule for long enough can collect important information on the execution pattern of the system. This information, in turn, can aid the adversary in launching successful attacks [8]. Recently, there have been a large variety of attacks that target this vulnerability [9], [10]. There have also been proposed defense mechanisms to prevent adversaries from succeeding with these attacks [11]–[13]. The good mechanisms are the ones that do not significantly reduce the performance of RTS.



(a) vanilla schedule (b) randomized schedule

Fig. 1: Reused from Chen *et al.* [14]. An example of the real-time schedule, consisting of 3 periodic real-time tasks, produced by (a) a vanilla EDF scheduler and (b) a randomized scheduler.

Schedule indistinguishability [14] is one such mechanism that reduces the success of attacks by randomizing the schedule instead of eliminating the presence of scheduler side-channels. Figure 1(b) shows the schedule of periodic tasks after applying randomization. Employing it not only gives protection to RTS, but the protection itself can be measured. This technique also achieves a trade-off between security and utility and provides a parameter that can be used to tweak it, namely ϵ. Schedule indistinguishability heavily relies on mathematical principles from differential privacy (DP) [15], [16]. While DP protects data privacy in the face of queries on databases by adding random noise, schedule indistinguishability adds *Laplace* noise to a task's execution to break easily discernible patterns in the task's schedule.

The idea of schedule indistinguishability was constructed based on ϵ-differential privacy (section 2 in [16]) to be applied to RTS. Defining it in this way gives rise to specific differentially private randomized mechanisms that can be employed to get the desired security like the Laplace noise-induced randomization mechanism [14]. Differential Privacy basically puts a restriction on the statistical difference between two distributions. The two distributions are the results of the randomized mechanism acting on two adjacent databases. Real-time schedules are the analogue of distributions in schedule indistinguishability. However, the domain of randomized mechanisms is restricted due to the selection of a specific statistical difference measure, namely ϵ.

A. Research Goal

We hypothesize that schedule indistinguishability defined using another statistical difference metric, the Rényi divergence [17], can yield stricter bounds on protection duration and give us insights on the advantage of specific timing-based

side-channel attacks on RTS. Moreover, we predict that bounds on protection duration and attack advantage will change if we model schedule indistinguishability in the style of a game between a challenger and an adversary, as seen in membership inference attacks [18]. Hence, we can outline our contributions as follows:

1) Re-write the definition of schedule indistinguishability using Rényi differential privacy [19]. Proofs for indistinguishability-preserving mechanisms will now depend on this definition.
2) Define a game between a challenger and an adversary along the lines of a crypto-style game. The goal of the adversary is to produce a time window within which a particular job of a task will arrive.

II. Background

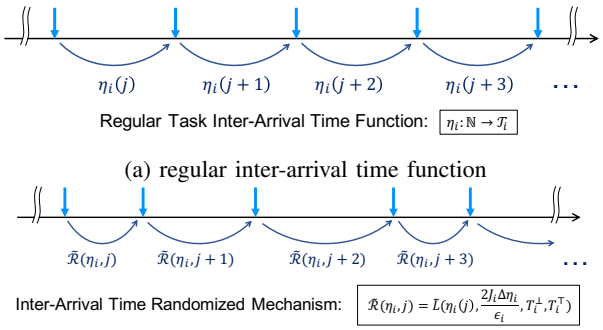
A. Differential Privacy

Differential Privacy was introduced in the context of statistical queries on databases and has grown into a widely used technique to ensure data privacy [15], [16], [20]. It guarantees that a malicious querier (adversary) cannot reason with high confidence about the presence of a particular data entry by just looking at the outputs of queries returned by differentially private mechanisms. Moreover, such protection is quantifiable based on the randomization mechanism used. The *Laplace* mechanism is a prototypical ϵ-differentially private mechanism, allowing release of an approximate (noisy) answer to an arbitrary query. Note that while the high-level goals are similar, leakage of private data is the typical use case for differential privacy as opposed to deterring scheduler side-channel attacks.

In our context, there are task and job level indistinguishabilities that define the probability of distinguishing the execution states of one task/job from another in task schedules. Roughly speaking, a low indistinguishability enables an adversary to identify a task's execution from an observed schedule with high confidence and hence makes the system prone to compromises via scheduler side-channels. To address such a problem, we have an ϵ-Scheduler [14] that offers "ϵ-indistinguishability" at job level and/or task level, subject to system constraints as well as the system designer's security goal. This is achieved by embedding a randomized scheduling mechanism for adding noise to the inter-arrival times for each job at every scheduling point to abate the predictability and determinism.

B. Schedule Indistinguishability

At its core, schedule indistinguishability seeks to break the determinism in schedules of RTS. Precisely, it utilizes randomization techniques to obfuscate task schedules, thereby disrupting the predictability in schedules of RTS. This is achieved by adding a sufficiently large (controlled) noise to the task schedules in order to break their deterministic execution patterns. Not only does it introduce diversity into the schedules of such systems, but also provides a scope for analyzable security guarantees. The goal is to offer protection against scheduler side-channels in RTS while still maintaining good real-time performance.



(a) regular inter-arrival time function

(b) randomized inter-arrival time function

Fig. 2: Reused from Chen *et al.* [14]. Illustration of the task execution model (a) before injecting noise and (b) after injecting noise.

We model a real-time task τ_i by a tuple $(\mathcal{T}_i, \eta_i)$ where $\mathcal{T}_i$ is a set of admissible periods and η_i is the task inter-arrival time function. Every task is made up of a finite number of jobs. To keep things simple, we will not discuss deadlines and execution times in this manuscript. The inter-arrival time function is illustrated in Figure 2. It is defined as

$$\eta_i : \mathbb{N} \to \mathcal{T}_i \tag{1}$$

where τ_i is the task and $\eta_i(j)$ is the task's inter-arrival time (same as period) at the j^{th} instance/job.

The inter-arrival time randomized mechanism, denoted by $\mathcal{R}(\cdot)$, is attached to the scheduler to add random noise. It is defined as

$$\mathcal{R}(\tau_i, j) = \lfloor \eta_i(j) + Y \rceil \tag{2}$$

where $j \in \mathbb{N}$ represents the j^{th} inter-arrival time of the task τ_i. Y is a random noise value drawn from some distribution centered at 0.

An inter-arrival time randomized mechanism is job-level ϵ-indistinguishable if

$$\Pr[\mathcal{R}(\tau, j) \in \mathcal{S}] \leq e^{\epsilon} \Pr[\mathcal{R}(\tau, j') \in \mathcal{S}] \tag{3}$$

for all $j, j' \in \mathbb{N}$ and $\mathcal{S} \subseteq \text{Range}(\mathcal{R})$. This means $\mathcal{R}(\cdot)$ enables inter-arrival time indistinguishability for a single job instance of task τ if (3) is satisfied. To determine the degree of noise to be added, we use inter-arrival time sensitivity, denoted by $\Delta\eta_i$ for a task τ_i. Then, the use of the Laplace distribution $\text{Lap}(\eta_i, \frac{\Delta\eta_i}{\epsilon})$ for generating the randomized inter-arrival times preserves ϵ-indistinguishability from (3) for a single job instance. It can be shown that the Laplace randomized mechanism $\mathcal{R}(\cdot)$ with the scale $\frac{J\Delta\eta_i}{\epsilon}$ is ϵ-indistinguishable up to J job instances.

III. Re-defining Schedule Indistinguishability

Before attempting to re-define schedule indistinguishability, we answer a few fundamental questions about it. **What are we trying to protect?** We are trying to protect a job given its

inter-arrival time. In other words, the attacker shouldn't be able to predict whether a given job was j or not with high certainty. **What does the adversary know?** The assumption is that the attacker knows, via some side-channel, the inter-arrival time function, $\eta_i(.)$, of the task τ_i. Hence, it knows all values $\eta_i(j)$, where j is any job of task τ_i. **What is the trivial solution and why does it not work in RTS?** A trivial solution is mapping $\eta_i(j)$ to the same value for all j, *i.e.*, making all the inter-arrival times equal for task τ_i. Now, all jobs have the same inter-arrival time and hence the jobs cannot be distinguished based on their corresponding inter-arrival times. However, this solution is not feasible in a (not strictly periodic) RTS because often the inter-arrival time function of a particular task maps to a finite set of admissible periods, $\mathcal{T}_i$, and thus $\eta_i(j)$ can output any value within $\mathcal{T}_i$. Hence, the trivial solution will lead to jobs missing their deadlines.

A. Indistinguishability with Rényi divergence

1) Rényi Divergence: For two continuous probability distributions P and Q defined over $\mathcal{R}$ with probability density functions (PDF) p and q respectively, the Rényi divergence of order $\alpha > 1$ is

$$D_\alpha(P||Q) = \frac{1}{\alpha - 1} \log \int_{-\infty}^{+\infty} p(x)^\alpha q(x)^{1-\alpha}\, dx \tag{4}$$

2) Rényi-indistinguishability (RI): An inter-arrival time mechanism $\mathcal{R}(.)$ is job-level (α, ϵ)-indistinguishable or Rényi-indistinguishable if

$$D_\alpha(\mathcal{R}(\tau, j)||\mathcal{R}(\tau, j')) \leq \epsilon \tag{5}$$

for all $j, j' \in \mathbb{N}$.

3) Laplace Mechanism in RI: The inter-arrival time Laplace mechanism for the j^{th} job of task τ_i is defined as

$$\mathcal{R}(\tau_i, j) = \eta_i(j) + \text{Lap}(0, b) = \text{Lap}(\eta_i(j), b) \tag{6}$$

It is important to find out the scale, b, of the noise required in the Laplace mechanism defined in (6). By the definition of Rényi-indistinguishability (5), it must satisfy $D_\alpha(\text{Lap}(\eta_i(j), b)||\text{Lap}(\eta_i(j'), b)) \leq \epsilon$. If $\Delta\eta_i$ is the inter-arrival time sensitivity of task τ_i (Equation 4 in [14]), then solving for b, we get $b \geq \frac{\alpha\Delta\eta_i}{(\alpha-1)\epsilon}$.

Now this scale gives us the Laplace noise distribution required to satisfy Rényi-indistinguishability for just one job of task τ_i. In order to extend it for J jobs, we have to first define a mechanism based on the Laplace distribution for J jobs. This can be easily done by defining the mechanism as $\mathcal{R}^J(\tau_i, j) = \{\mathcal{R}(\tau_i, k) \mid j \leq k \leq J + j\}$. Now we have to satisfy $D_\alpha(\mathcal{R}^J(\tau_i, j)|\mathcal{R}^J(\tau_i, j'))) \leq \epsilon$. After solving for b, we get $b \geq \frac{J\alpha\Delta\eta_i}{(\alpha-1)\epsilon}$.

B. Game-style Indistinguishability

1) Game Definition: The game proceeds between a challenger and an adversary. The adversary predicts a window of arrival (time range) for some job i of a task τ. The adversary's goal in this game is to ensure that the job arrives within the time window it has specified.

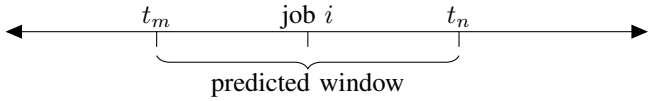


Fig. 3: An adversary strategy where it predicts a window for job i based on the learned system parameters.

- The challenger starts the execution of a real-time schedule at time t_0.
- The adversary observes the schedule via some side-channel from time t_0 till time t_p.
- At time t_p, the adversary chooses a time window of arrival, $w = [t_m, t_n]$, for some job i of the task τ.
- The challenger outputs a bit b. Now, $b = 1$ if job i actually arrives within the time window w, else $b = 0$.
- The adversary wins the game if $b = 1$, otherwise it loses.

2) Adversary's Strategy: An adversary can follow a strategy where it predicts a time window based on a probability distribution that it constructs using the parameters it has learned via the timing side-channel. Let G be a random variable that gives the arrival time of job i of task τ from the given real-time schedule. Let Z be a random variable that gives the arrival time of job i, following the adversary's predicted probability distribution. We assume that the adversary does not know the distribution that G follows. Then the probability that the adversary wins is given by the joint probability $P(t_m \leq G \leq t_n, t_m \leq Z \leq t_n)$. It evaluates to

$$\int_{t_m}^{t_n} f_G(g)\, dg. \int_{t_m}^{t_n} f_Z(z)\, dz \tag{7}$$

where the PDFs of G and Z are f_G and f_Z, and G and Z are independent. Such a strategy is shown in Figure 3.

IV. Conclusion and Future Work

This report presents the problem surrounding the current definition of schedule indistinguishability, where it is defined based on ϵ-differential privacy. We have seen how the requirements for the scale of the noise in the Laplace mechanism change when an alternate definition of schedule indistinguishability based on Rényi-differential privacy is used. The next steps here are to find a bounded Laplace mechanism in Rényi-indistinguishability that satisfies the real-time constraints in RTS and find the corresponding protection duration.

Further, the report explores a new game-style definition of schedule indistinguishability, where the game is inspired from membership inference attacks. We present the rules of the game and a possible attack strategy the adversary can use to win it. The next step here is to find a bound on the advantage of the adversary using the specified strategy or a likelihood ratio attack [18].

Acknowledgment

I would like to express my deepest gratitude to my advisor Prof. Sasha Boldyreva (sasha@gatech.edu) for guiding me throughout the project.

REFERENCES

[1] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, et al., "Experimental security analysis of a modern automobile," in *2010 IEEE symposium on security and privacy*. IEEE, 2010, pp. 447–462.

[2] Defense Use Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, vol. 388, pp. 1–23, 2016.

[3] Thomas M Chen and Saeed Abu-Nimeh, "Lessons from stuxnet," *Computer*, vol. 44, no. 4, pp. 91–93, 2011.

[4] Byungho Min and Vijay Varadharajan, "Design and analysis of security attacks against critical smart grid infrastructures," in *2014 19th International Conference on Engineering of Complex Computer Systems*. IEEE, 2014, pp. 59–68.

[5] Man-Ki Yoon, Bo Liu, Naira Hovakimyan, and Lui Sha, "Virtualdrone: virtual sensing, actuation, and communication for attack-resilient unmanned aerial systems," in *Proceedings of the 8th international conference on cyber-physical systems*, 2017, pp. 143–154.

[6] Ke Jiang, Lejla Batina, Petru Eles, and Zebo Peng, "Robustness analysis of real-time scheduling against differential power analysis attacks," in *2014 IEEE Computer Society Annual Symposium on VLSI*. IEEE, 2014, pp. 450–455.

[7] Hagai Bar-El, Hamid Choukri, David Naccache, Michael Tunstall, and Claire Whelan, "The sorcerer's apprentice guide to fault attacks," *Proceedings of the IEEE*, vol. 94, no. 2, pp. 370–382, 2006.

[8] Paul C Kocher, "Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems," in *Annual International Cryptology Conference*. Springer, 1996, pp. 104–113.

[9] Chien-Ying Chen, Sibin Mohan, Rodolfo Pellizzoni, Rakesh B Bobba, and Negar Kiyavash, "A novel side-channel in real-time schedulers," in *2019 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*. IEEE, 2019, pp. 90–102.

[10] Joon Son et al., "Covert timing channel analysis of rate monotonic real-time scheduling algorithm in mls systems," in *2006 IEEE Information Assurance Workshop*. IEEE, 2006, pp. 361–368.

[11] Hyeongboo Baek and Chang Mook Kang, "Scheduling randomization protocol to improve schedule entropy for multiprocessor real-time systems," *Symmetry*, vol. 12, no. 5, pp. 753, 2020.

[12] Kristin Krüger, Marcus Volp, and Gerhard Fohler, "Vulnerability analysis and mitigation of directed timing inference based attacks on time-triggered systems," *LIPIcs-Leibniz International Proceedings in Informatics*, vol. 106, pp. 22, 2018.

[13] Man-Ki Yoon, Sibin Mohan, Chien-Ying Chen, and Lui Sha, "Taskshuffler: A schedule randomization protocol for obfuscation against timing inference attacks in real-time systems," in *2016 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*. IEEE, 2016, pp. 1–12.

[14] Chien-Ying Chen, Debopam Sanyal, and Sibin Mohan, "Indistinguishability prevents scheduler side channels in real-time systems," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 666–684.

[15] Cynthia Dwork, "Differential privacy: A survey of results," in *International conference on theory and applications of models of computation*. Springer, 2008, pp. 1–19.

[16] Cynthia Dwork, Aaron Roth, et al., "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[17] Tim Van Erven and Peter Harremos, "Rényi divergence and kullback-leibler divergence," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3797–3820, 2014.

[18] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer, "Membership inference attacks from first principles," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1897–1914.

[19] Ilya Mironov, "Rényi differential privacy," in *2017 IEEE 30th computer security foundations symposium (CSF)*. IEEE, 2017, pp. 263–275.

[20] Konstantinos Chatzikokolakis, Miguel E Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi, "Broadening the scope of differential privacy using metrics," in *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 2013, pp. 82–102.