Poster: Towards Pattern-Level Privacy Protection in Distributed Complex Event Processing

Majid Lotfian Delouee
University of Groningen

Boris Koldehofe
Technische Universität Ilmenau

Viktoriya Degeler
University of Amsterdam

ABSTRACT

In event processing systems, detected event patterns can reveal privacy-sensitive information. In this paper, we propose and discuss *how* to integrate pattern-level privacy protection mechanisms in event-based systems. Compared to state-of-the-art approaches, we aim to enforce privacy independent of the particularities of specific operators. We accomplish this by supporting the flexible integration of multiple obfuscation techniques and studying different deployment strategies for privacy-enforcing mechanisms. In addition, we share ideas on *how* to model the adversary's knowledge to better select appropriate obfuscation techniques for the discussed deployment strategies. Initial results indicate that flexibly choosing obfuscation techniques and deployment strategies is essential to conceal privacy-sensitive event patterns accurately.

KEYWORDS

Privacy, Complex Event Processing, Pattern.

1 INTRODUCTION

Why Pattern-Level Privacy Protection is critical? Distributed Complex Event Processing (DCEP) is a state-of-the-art paradigm to process streams of simple events (e.g., IoT data) and produce valuable information, so-called complex events, in real time. For instance, a *road congestion* can be inferred by a traffic monitoring system via simple events: $Average\_Vehicles\_Speed < 20\ km/h$ and $Vehicles\_Density > Normal\_Density$. Although the output of such systems can help various applications by delivering useful information, some of these complex events might be privacy-sensitive. Thus, it is required to preserve the privacy of data owners by applying *Privacy Protection Mechanisms (PPM)* over the event streams to meet the privacy requirements.

Most PPMs (e.g., access control) provide privacy protection at the level of single events (i.e., by protecting event attributes). In contrast, privacy demands often can be represented using a combination of events through event patterns. It means the capabilities of conventional PPMs should be extended from event-level to pattern-level. For example, the blood pressure and heart rate data can reveal a disease when they are involved in an aggregated analysis, while they are not so helpful separately. Besides, concealing such a privacy-sensitive event pattern (i.e., *private pattern (PrP)*) can impact the detection of non-sensitive event patterns (i.e., *public pattern (PuP)*). Hence, a *pattern-level PPM* is required to provide a trade-off between privacy and utility by obfuscating as many PrPs while publishing as many PuPs.

Existing Solutions and Their Pitfalls. Current approaches towards supporting pattern-level privacy propose pattern-based access strategies [2, 3]. However, these techniques are limited to dealing only with sequence types of patterns. Besides, they depend highly on the input streams to apply obfuscation techniques (OT), thereby failing to obfuscate various types of PrPs well in an environment with dynamic input event streams. In addition, previous works stated that the CEP middleware is not trustable. This assumption reduces the detection of all matches for PuPs since they join the input streams before the CEP middleware. However, suppose PrPs are concealed in the sink node within the CEP middleware for specific queries. In that case, it will not negatively impact the results of the other queries. Besides, the potential knowledge an adversary might have needed to be better studied and involved in the obfuscation procedure.

In this paper, we present a pattern-level PPM to fulfill data owners' privacy requirements while considering adversaries' potential background knowledge. The ending goal of this approach is to provide a system that obtains privacy requirements (i.e., PrPs) and situations of interest (i.e., PuPs), both in the form of event patterns as input, and decide about obfuscation technique by providing a trade-off between concealing PrPs when delivering the results to the queries. Moreover, we model the adversary's potential background knowledge based on event dependencies and possible statistical information gathered from the available event streams. While we initially focused on sequence types of patterns and limited this paper to the two most common obfuscation techniques (i.e., drop and reorder), we believe the final proposed mechanism can eventually cover more pattern types and obfuscation techniques to decrease the adversary's ability to distinguish between the original and modified streams. In the following, we first elaborate on our approach, present the preliminary experiment results, and conclude the paper.

2 OUR APPROACH

This privacy protection mechanism aims to minimize the impacts of concealing PrPs on detecting PuPs, even in the presence of knowledgeable adversaries. In Figure 1 (left), the initial system design of the pattern-level privacy preservation system is exhibited. Here, we do not provide our own pattern generation for PrPs but build on existing concepts for the flexible transformation of privacy requirements to PrPs[4]. However, finding the best obfuscation technique which maximizes the utility and at the same time does not reveal the concealing of the PrPs to adversaries is non-trivial.

The first challenge in this research is the utility metric by which the performance of each OT model on a specific PrP can be assessed. One significant criterion to compare the performance of models is their ability to keep detecting PuPs while decreasing the number of PrPs revealed. Moreover, the false detection of PuPs should be considered a negative factor since it reduces the OT model's detection accuracy. Besides, modeling the adversaries' knowledge is another part of the mentioned utility metric that ensures the obfuscation's performance against the potential statistical knowledge an adversary might gain from the previous queries or other sources. To this end, event dependencies should be defined beforehand to be considered in the process of OT model selection. Such dependencies can be presented as *Causal dependencies* (i.e., the occurrence of event e_2 depends on the event e_1), *Periodic events* (e.g., event e_1 happens every 2 hours), or *Infeasible events* (e.g., event *"at the office"* cannot occur 5 minutes after event *"at home"* when the fastest route between

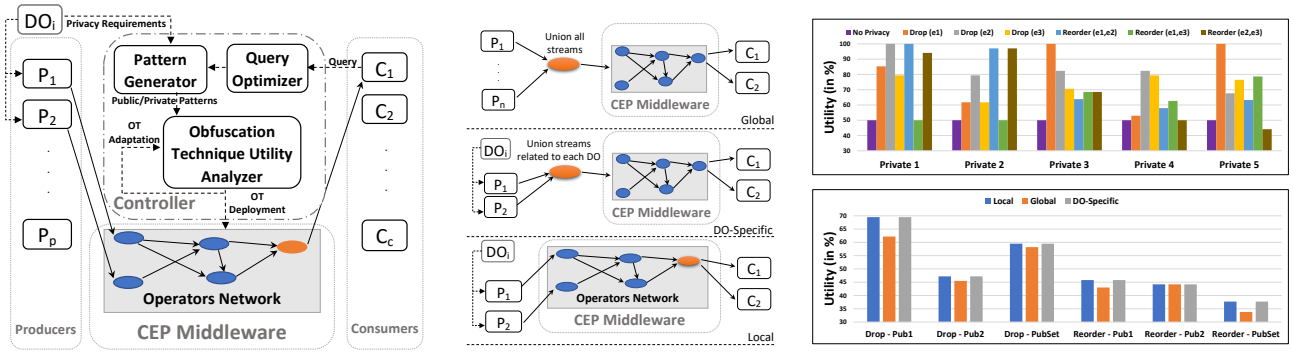

Figure 1: (left) System Design, (middle) Obfuscation Technique's Deployment Strategies, (right) The Initial Results

home and office is 30 minutes). It is worth noticing that, in some cases, concealing a PrP might cause one or more other PrPs to be revealed or obfuscated. Therefore, a potential solution requires considering the impacts of an OT model on the rest of the PrPs.

As a second challenge, where a selected OT model should be deployed brings even more complications, as illustrated in Figure 1 (middle). Accordingly, the decision metric is whether or not the CEP middleware is assumed to be trustable. In the case of no trust, two strategies can be deployed to apply OT models. In the first strategy, it is required to union all the input event streams before entering the CEP middleware, the so-called *Global* strategy. Here, concealing a PrP makes the system more prone to further false detection (i.e., False Positive (FP)) or no detection (i.e., False Negative (FN)) of PuPs. The second strategy, so-called *DO-Specific*, unions streams associated only with a specific data owner and then applies OT models on the combined stream. A problematic assumption with this strategy is that any producer must be assigned only to one data owner, e.g., a sensor can only be used for a specific data owner. Hence, those event producers who generate streams related to multiple targets (e.g., a camera) cannot be involved. On the other hand, in a trustable CEP middleware, each PrP can be obfuscated in the sink node exactly before delivering the results to query consumers. Such a strategy is called *Local* deployment, where the probability of influencing the detection of other PuPs becomes lower, leading to higher utility metric values. On the contrary, it has vulnerability against *Omnipresent Adversary*. Since the system chooses various OT models for different queries, an omnipresent adversary can realize the pattern obfuscation (e.g., a dropped event for a query might be delivered in the results of another query). However, the first two strategies are able to overcome such a threat since they provide the same stream to all consumers related to a specific data owner.

3 PRELIMINARY EXPERIMENTS

Our Main evaluation goals are to figure out 1) makes the choice of obfuscation technique impacts the utility of event processing, and 2) how does the deployment of the obfuscation operators influence the utility and dealing with the adversary's knowledge.

Simulation Setup: We evaluated our ideas by analyzing a synthetic dataset in a scenario implemented in a virtual machine with 6 CPU cores with total execution capacity and 24 GB of main memory. Furthermore, for detecting complex events, we build on *FlinkCEP* [1], a library implemented on top of Apache Flink. PrPs in our simulation contain three events that form a sequence pattern (i.e., $e_1 \rightarrow e_2 \rightarrow e_3$), and PuP is formed by a sequence of two events. We compare the OT models using a utility function calculated as a weighted sum of truly detected PuPs (TP_{pub}), false detected PuPs (FP_{pub}), truly obfuscated PrPs (TO_{priv}), and false revealed PrPs (FR_{priv}). Note that FP_{pub} and FR_{priv} have negative impacts on the utility function, and misdetection of PuPs is already counted in TP_{pub}. Moreover, we assume the weights for detected PrPs (i.e., TO and FR) equal the expected *matches* of PuPs over the expected *matches* of PrPs in each window. This way, we made the obfuscation of each PrP so crucial and dynamic on each window, against static weights assumed in [2] only based on the *number* of PuPs.

Initial results: Figure 1 (right) reports the initial results of using the mentioned utility function. It shows that applying various models of obfuscation techniques on a specific PrP produces unequal utility values. This means selecting the OT model with the highest value for each PrP to improve obfuscation performance matters. Besides, this figure indicates no OT model fits all PrPs. Although dropping events performs better for most PrPs, reordering sometimes shows its value. The main vulnerability of drop-based OT models is the possibility of misdetection in PuPs (i.e., FNs), whereas reordering-based models tend to wrongly detect extra matches in both types (i.e., FP_{pub} and FR_{priv}). Regarding deployment, Local and DO-Specific strategies perform similarly, whereas Both defeat the Global strategy. This proves our primary assumption about the tendency of Global strategy to produce more FPs and FNs for PuPs. Notice that despite Local deployment, the DO-Specific strategy is able to withstand the Omnipresent threat model, which is in harmony with considering the adversary's background knowledge.

In conclusion, involving more PrPs and PuPs with different levels of dependencies will open up more dimensions of this topic. In addition, defining parameters to explain the relationship between PrPs and PuPs would lead to autonomous obfuscation technique adaptation, making obfuscation decisions independent of the event streams.

REFERENCES

[1] Apache Flink Community. 2017. *FlinkCEP*. Retrieved May 1st, 2023 from https://nightlies.apache.org/flink/flink-docs-master/docs/libs/cep/.

[2] S.M. Palanisamy. 2020. Towards Multiple Pattern Type Privacy Protection in Complex Event Processing Through Event Obfuscation Strategies. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2020 International Workshops, DPM 2020 and CBT 2020, Guildford, UK, September 17–18, 2020, Revised Selected Papers 15*. Springer, pp.178–194.

[3] S.M. Palanisamy, F. Dürr, M.A. Tariq, and K. Rothermel. 2018. Preserving Privacy and Quality of Service in Complex Event Processing through Event Reordering. In *Proceedings of the 12th ACM International Conference on Distributed and Event-based Systems*. pp.40–51.

[4] C. Stach and F. Steimle. 2019. Recommender-based privacy requirements elicitation-EPICUREAN: an approach to simplify privacy settings in IoT applications with respect to the GDPR. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. pp.1500–1507.