An Information-Theoretic Approach to Time-Series Data Privacy

Yousef Amar
Queen Mary University of London *
y.amar@qmul.ac.uk

Hamed Haddadi
Imperial College London
h.haddadi@imperial.ac.uk

Richard Mortier
University of Cambridge
richard.mortier@cl.cam.ac.uk

Abstract

Access control is central to interfacing with personal data, however most systems today are too coarse and disconnected from the privacy context of the data. Granular access control rarely goes beyond limiting sample rates or enforcing time limits. In this paper, we present a system for tuning a data consumer's access to personal data based on real-time privacy metrics. We first explore the potential definitions of privacy in this context with a focus on information theoretic metrics for defining privacy in sensitive time series data. We then implement and evaluate our system for embedding risk thresholds into bearer token-based access control systems to attenuate access to data of different granularities based on these metrics. Our results show that our system provides privacy gains wihout a significant utility cost, and can run efficiently and scale well on cheap hardware with high-frequncy sensor data.

1. Introduction

When it comes to personal data, traditional access control mechanisms between a *producer* and a *consumer* of data exhibit critical problems. With social media APIs, phone sensors, and even files on a PC, access is most often a binary "all or nothing". While in some cases access can be attenuated to read or write, and can expire after a certain time period, this level of granularity is too coarse, and does not consider the content of the data in any way.

APIs that do allow more fine-grained controls often require an understanding of the context and possible inferences [5], to then allow a user take context-specific actions such as spoofing GPS coordinates or occluding faces in images. While this is useful, it is difficult to scale and generalize to arbitrary data types without user interaction or complexity for understanding semantics.

Furthermore, users are often unaware of just what information they really are exposing [1], and cannot be expected to keep track of inferences that may be caused by anomalies or patterns in their data, especially not in real time. Dynamically adjusting access control restrictions based on online privacy and risk metrics remains an open problem.

These shortcomings culminate in access control mechanisms with only very superficial privacy awareness. The goal of this work is to introduce a scalable, privacy-aware access control system that solves these problems.

We seek to do this by applying established, information-theoretic privacy metrics as criteria in access control systems over time series data. These metrics must be *context-independent*, operate in *real time* on *cheap hardware* located *at the source* of data. These constraints are imposed by the context in which we deploy this system and evaluate its performance and extent of privacy preservation.

While our method is applicable to any system where a consumer pulls personal data from a producer under the restrictions of an access control system, we implement and evaluate it in the context of the Databox platform [7] — a home-based networked device that provides a controlled, sandboxed environment for processing personal data. Here, third-party *drivers* query external data sources and write data to system-managed *stores*. These are then queried by *apps* that perform analytics and, if necessary, only emit results back to third parties.

Figure 1 shows the components pertinent to this system. Here, solid arrows denote the paths that data can flow. As a single app can read from and write to many stores, these paths can manifest themselves as complex networks of cross-source analytics and derived stores. Our access control systems act at the red arrows. The *arbiter* mints signed bearer tokens with privacy thresholds embedded within them, and passes these on to apps and drivers. When interfacing with stores, these tokens are independently verified by said stores. Finally, any data leaving the box to be consumed by a third party is similarly subjected to the same thresholds.

* This work was done while the author was visiting Imperial College London.

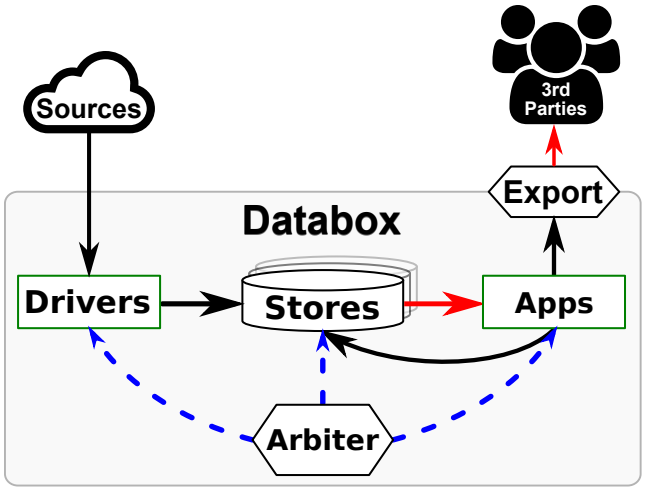

Figure 1. A high-level overview of Databox components

The core approach to how our system is deployed in this context is therefore twofold. As stores act as a border between producers and consumers, we first continuously update and maintain a *privacy context* for each data stream within each store using common privacy metrics. Then in tracking these metrics, we adjust the flows (in the simplest case by suppressing them or repeating old values) based on thresholds embedded within the tokens used to query a stream.

In this paper, we describe an implementation of this using information theoretic privacy metrics on time series data. We then evaluate our system's privacy-utility trade-off, as well as its performance in real-time, low-latency use cases, such as in embedded and home IoT devices. We show that our system provides privacy gains without a significant utility cost, and can run efficiently and scale well on cheap hardware with high-frequency sensor data.

2. Background

2.1 Privacy Metrics

Privacy in the context of personal data is a well explored topic. It is not the goal of this work to develop new metrics, but rather to apply and evaluate existing metrics to the systems we develop.

Often privacy and anonymity are used interchangeably, but there are some very distinct differences between the two. The demand for privacy exists despite anonymity, and is indeed more pronounced when individuals are not anonymous.

Dalenius first coined the term *quasi-identifier* in 1986 [9] and since then, a number of seminal publications have dealt with the process of identifying individuals by making inferences from data that may not contain any explicit identifiers (such as a UID).

Famous examples include the ability to uniquely identify 87% of the population of the United States by combining gender, birth date, and post code information [20], as well as the deanonymization of the Netflix Prize Dataset by combining it with public IMDB data [19].

While deanonymization relies on the linkage of data to explicit identifiers, more privacy-centric methods focus on making it more difficult to connect sensitive attributes to individuals.

Recent, comprehensive survey papers describe an extensive range of metrics for a vast array of different purposes [10] and organise these into taxonomies [22].

While many of these provide average risk measurements over a given dataset, some have been repurposed to provide "one-symbol information", or the marginal mutual information from the appending of an additional record [2, 3].

The following is a description of a number of metrics suitable to our method. We divide these by output measure into two categories based on Wagner and Eckhoff's taxonomy [22].

2.1.1 Similarity/Diversity

K-anonymity [21] is an exceedingly prevalent privacy measure. To quote the original paper, "A release provides k-anonymity protection if the information for each person contained in the release cannot be distinguished from at least k-1 individuals whose information also appears in the release". Explicit identifiers are completely suppressed and quasi-identifiers generalised. Similarly, rows in time series data can form equivalence classes after microaggregation based on a quasi-identifier column, where the smallest cluster has k rows.

L-diversity [18] is an extension of k-anonymity that additionally requires that sensitive attributes are well-represented in each equivalence class (for various definitions of "well represented"). It is therefore less susceptible to homogeneity attacks and background knowledge attacks. L-diverse data is by definition at least l-anonymous.

T-closeness [16] goes yet another step beyond and takes account of the distance between the distribution of sensitive attributes in any single equivalence class, to the distribution of sensitive attributes across the whole dataset. The distance measure is arbitrary, though the original paper uses Earth Mover's Distance. T-closeness for the whole dataset is the maximum of t-closeness for each equivalence class. This addresses potential attacks on l-diversity such as skewness attacks.

2.1.2 Information Gain/Loss

While pure entropy is an average value over a distribution, we want a marginal measure for every symbol. This is where information **surprisal** becomes useful. Surprisal is also known as self-information, however as this term is sometimes used interchangeably with entropy, we will refer to it as surprisal throughout this paper. As a measure, it has been used in the past to, for example, measure information gain from the attributes of public social media profiles [8]. When sampling a variable, surprisal is a measure

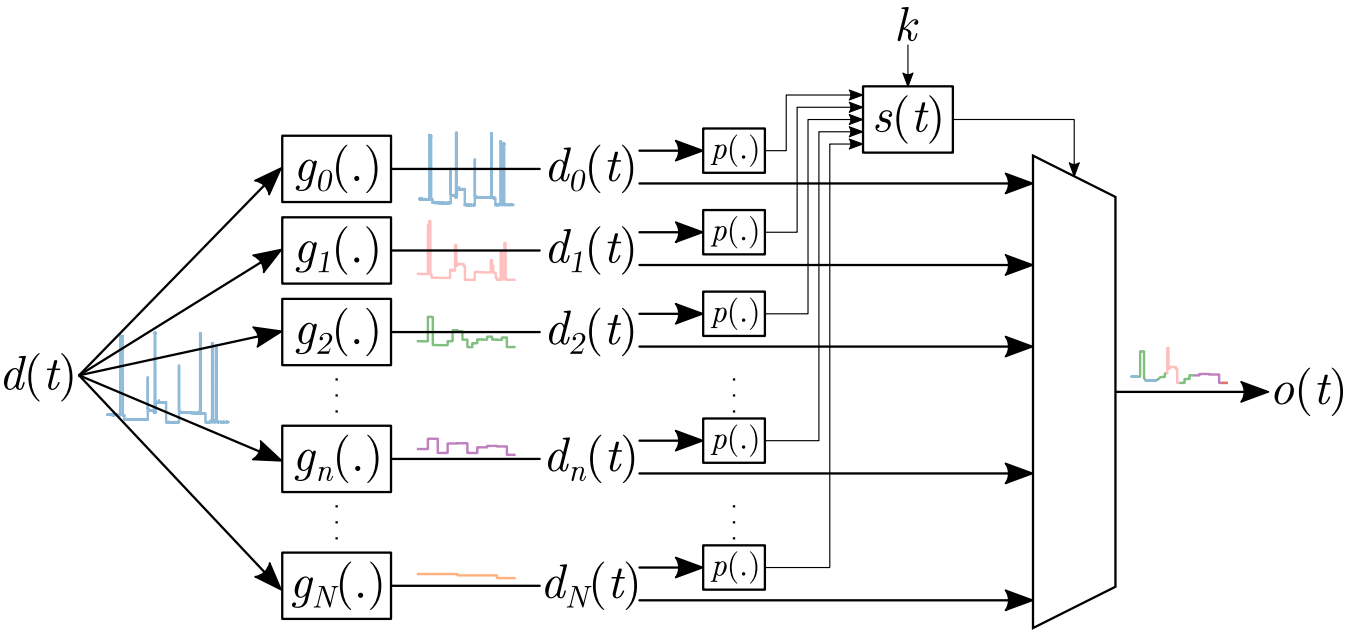

Figure 2. An overview of of our implementation

(in information-entropic bits) of uncertainty associated with sampling this variable — the negative logarithm of the probability of a sample.

Finally, many inferences can be made just from identifying patterns in time series data. A simple example is inferring location from temperature data, while a more advanced example is identifying what you watch on TV from smart meter data [11]. The Pearson correlation coefficient has been used in the past to compare smart meter data before and after anonymization [14] as opposed to other common distance measures such as KL-divergence [13].

Similarly, we can can cross-correlate data with itself shifted by varying time lags: **autocorrelation**. This allows us to detect seasonality and patterns in time series data, and on doing so, suppress, shift, or perturb the output.

2.2 Access Control

Our method can be adapted to any access control system that has a notion of per-consumer permissions, and as such, access control mechanisms are outside the scope of this paper. For our purpose, we implement a macaroon-based [4] bearer token system.

The *arbiter* in figure 1 mints tokens with embedded privacy thresholds (as macaroon caveats) that correspond to the permissions a bearer has. It maintains a record of these permissions that a user can modify at any time and take effect when an old token expires or is revoked.

The *arbiter* then cryptographically signs these tokens and passes them to data consumers potentially controlled by third parties (in this case *drivers* or *apps*). When a consumer makes a request to a *store*, it attaches the relevant token to the request. Permissions are embedded in these tokens, so the store/producer is aware of the privacy context and permissions, and can verify these tokens through their signatures using a secret key shared by the arbiter and the store beforehand.

Thus, access control decisions based on privacy can be made on a request to request basis.

3. Implementation

This section details our implementation and describes figure 2 which provides a visual overview of it.

We begin by transforming time series data $d(t)$ continuously to N different granularities. This can be for any definition of "granularity", however in our implementation, we calculate the mean of every 2^n samples for $n = 0, \cdots, N$ and interpolate by nearest neighbour. Alternatives include plain downsampling, summing/aggregation, or other forms of averaging. We calculate means as these have the greatest utility to our evaluation use case.

We denote these granularity transformation functions as $g_n(x)$, the outputs of which map to the original time series in the following manner: $d_n(t) = g_n(d(t))$.

For every new sample in the transformed data, $d_n(t)$, we update one or more corresponding privacy scores based on the privacy metrics described earlier. For our evaluation, we use surprisal so the unit of these scores is bits or shannons. We describe how we implement this and other privacy metrics in more detail in the next sections. We denote this privacy measure as the function $p(x)$.

The final component in this system is a multiplexer that selects the transformed data stream with the highest granularity but with a privacy score that is still below a threshold k. The moment a data stream's score exceeds k, the multiplexer

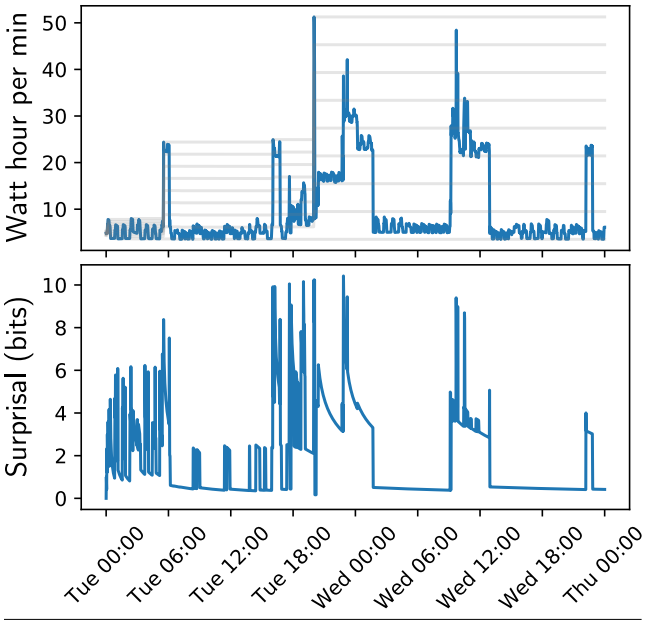

Figure 3. Surprisal over active energy consumed each minute with eight bins (gray) and an infinite window size

drops to a lower granularity, until it reaches the level of granularity that is the equivalent of a fixed grand mean across the entire stream. This is the point at which $n = N$. With our previously defined notation, our final output is $o(t) = d_m(t)$ where $m = \min\{n \mid n \in \mathbb{Z} \wedge n \in [0, N] \wedge p(d_n(t)) < k\}$. In other words, the highest resolution transformed stream with a privacy score below a given threshold.

$$d_n(t) = g_n(d(t)) \tag{1}$$

$$s(t) = \underset{\substack{n \in \mathbb{Z} \\ n \in [0,N] \\ p(d_n(t)) < k}}{\arg\max}\ d_n(t) \tag{2}$$

In equation 2, $s(t)$ is a shorthand to represent a function that returns the index of the data stream selected at time t used in figure 2. For a sample $d(t)$ at time t, the overall output can be formulated as a dynamic optimization problem. Here, selected data stream index n is our decision variable.

$$\begin{aligned} \min_{n} \quad & d_n(t) \\ \text{s.t.} \quad & n \in \mathbb{Z} \\ & n \in [0, N] \\ & p(d_n(t)) < k \end{aligned} \tag{3}$$

The threshold k is embedded in the tokens attached to requests made by data consumers. By modifying k for a consumer, we can modify the extent in permissions with respect to the metric used, and thus achieve privacy-aware access control.

It is important to note that dropping to a lower level of granularity is done *transparently* without the consumer's

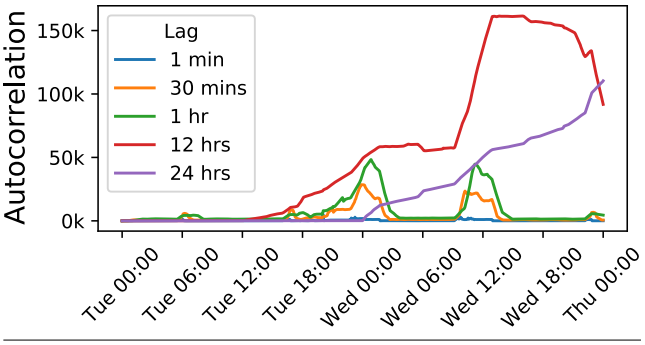

Figure 4. Autocorrelation over power consumption for different fixed lags

knowledge. We also test a variant of this system where the consumer directly requests a specific level of granularity, and their request is rejected if it is above what they have permission to access. The consumer must then make a new request to a lower granularity, which not only adds significant latency, but the act of denying access itself reveals some information on the nature of changes in the data, for instance if a stream that a consumer originally had access to suddenly became restricted.

As a proof of concept, we run the metrics described in the previous section over UCI's Individual Household Electric Power Consumption Data Set [17]. We focus on just two columns from this dataset: the timestamp column, and the global (minute averaged) active power, which we convert to watt hour. In order to more closely conform to realistic scenarios, we treat the range of this data as an unknown that is continuously updated as the maximum and minimum seen values are exceeded.

As data processed is continuous, it must be quantized first for certain metrics in order to become discrete. This is not the case for input data that is for example a byte of sensor data in the range $[0, 255]$, or strings from a set of limited size like country names. With continuous variables however, the probability of any one sample is near-zero and so we must divide the data into bins of a set interval in order for the metrics to make sense.

Furthermore, as we perform these measurements online, it is practically infeasible to repeat these for the entire dataset on every new addition. In our final implementation, we therefore only consider the data before a temporal cutoff point using simple, fixed-length, rectangular windows. We note however that windows with different configurations and weightings may potentially yield cleaner results.

In figure 3, it is clear that as samples are added to empty bins, surprisal spikes. As more are added to one bin (such as around Wednesday at midnight), surprisal slowly decreases, which makes sense intuitively. Suprisal is lowest for power consumption values below 10 watt hour, as this is the most replete bin.

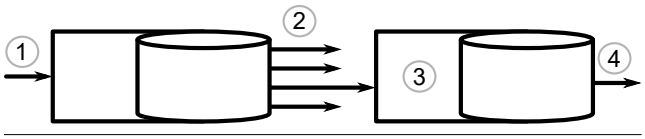

Figure 5. Stages of data transformation

To measure seasonality, we track autocorrelation across a set of lags over time. This is different than simply building a correlogram over a fixed length sequence, as we continuously calculate cross-correlation in an online manner. Figure 4 visualises the most significant lags. For 12 hours, autocorrelation is mostly higher than for 24 hours, which would also hinted by maxima in a correlogram. This tells us that patterns are most likely to arise at 12 hour periods, and we can automatically take steps to suppress this by for instance only emitting 12 hour averages. We can also normalise this data and embed cross-lag, or per-lag autocorrelation thresholds into our access control system. This could for example block access to data with exceedingly high 12 hour autocorrelation, but allow access to the same data only once it has been downsampled to obfuscate this pattern. As such, this is a simple, yet powerful metric to use in our system.

3.1 Databox Integration

Our system functions between data producers and consumers. In Databox, this means between driver and app, between app and app, and between app and the outside world. Every driver and app must output data via stores, therefore the store is at first glance the most obvious place to implement our system.

There is however another alternative that is more versatile from a development perspective, which is to implement our system as a "privacy filter" app, that reads private data from one store, and writes transformed data to a derived store. This approach has the advantage of being modular as only a single type of store is needed, while any number of type-specific filtering apps can exist (e.g. an app that blurs faces in images). The increase in network traffic and added overhead would hurt latency however.

We therefore implement our solution as an app, but future work may include hardcoding the most general types of time series aggregation into stores, and the monitors for the most common privacy metrics. Anything less common can be delegated to an intermediate app. Figure 5 shows this eventual pipeline — raw data enters at ①, is written into a store after being filtered/transformed at ②, and a second-order app ③ performs any additional more specific transformations before the data ④ is output in its final form.

4. Evaluation

Our system evaluation is in two parts: first, we evaluate the trade-off between privacy benefits and utility using our system, then we measure the marginal latency and performance when deploying this system in a realistic scenario on limited hardware.

4.1 Privacy/Utility Evaluation

Our system provides privacy along the axis of the privacy metric used based on the thresholds used. For example, tokens that only allow access to data with a surprisal value of less than four bits will implicitly favour less granular data. An emergent effect of this property is that higher frequency inferences can be suppressed in aggregate time-series datasets.

To test this, we take the same approach as previous work in the same domain [15] and make use of the Reference Energy Disaggregation Data Set (REDD). This dataset contains detailed power usage data from a number of houses including mains readings as well as on a per-device basis.

Our goal is to show that after subjecting aggregate mains data to our privacy-aware access control system, we can infer washer/dryer state while concealing microwave state. A realistic use case may be a smart meter app that suggests the best times to do your laundry based on your flatmates' habits, but does not need to know anything about your eating habits. Our system allows for this gain in privacy without compromising utility.

We use the state (on or off) of the washer/dryers and microwaves as ground truth and a Gaussian Naive Bayes classifier to predict whether or not these devices are turned on given mains data. The data has occurrences of both devices being turned on both separately and together. For each device, we plot Receiver Operating Characteristic (ROC) curves where we modify the privacy metric threshold, which is in this case an upper bound on bits of entropy.

In reality, an app might not have access to household-specific data to train such classifiers. We show however that even in this case (figure 6) utility remains virtually the same across thresholds while the undesired inference degrades.

4.2 Performance Evaluation

While the previous section showed that significant privacy gains can be made without degrading utility, previous work has achieved more impressive privacy/utility trade-offs [15]. Where our work differs is that it is also efficient enough to run online for real-time streaming data on cheap hardware.

In this section we show this by implementing our system over Databox, running it on typical hardware (an Intel NUC6i3SYH), and measuring the added latency in the pipeline. We examine the difference in *time to availability* (TTA) — the time between when a sensor emits a sample and when it becomes available in its final form to an app at the end of the pipeline. We can of course repeat the measurements on all derived data ad infinitum, but this has limited practical benefits.

We measure this latency for 20k samples under three conditions:

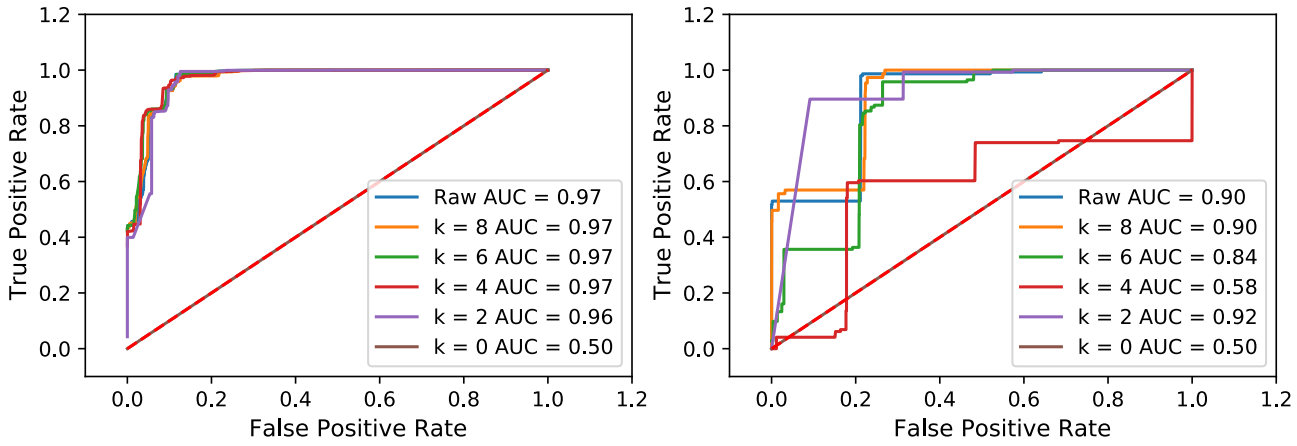

Figure 6. Receiver Operating Characteristic (ROC) curves for washer-dryer (utility; left) and microwave (attack; right)

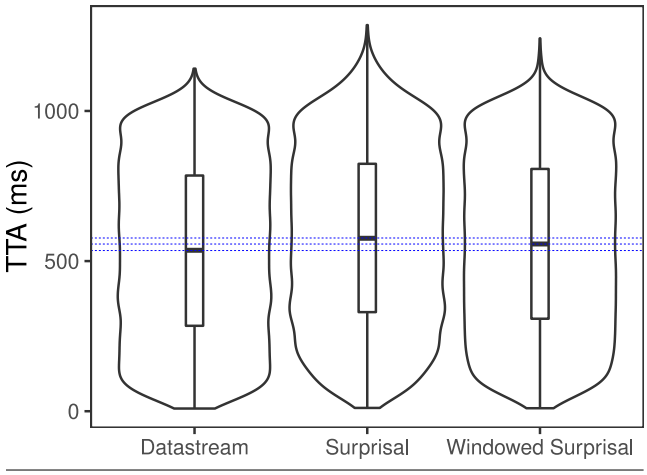

Figure 7. Distributions of time to availability under different conditions

- *Datastream*: Vanilla Databox times as a baseline. Access control is binary and at the datastream level.
- *Surprisal*: Inclusion of our system in the pipeline using surprisal as a privacy metric with a fixed threshold and infinite window size.
- *Windowed Surprisal*: The same as the previous experiment, but with a falloff of one minute.

Figure 7 shows the density of these latencies. The means, with blue dotted lines drawn through, are at 535.1633 ms, 576.9499 ms, and 556.7980 ms respectively. The difference between with and without surprisal calculation is negligible and well within the tolerance for real-time applications. The added latency is small enough that privacy filtering at this fidelity is possible without impacting user experience.

The small difference of 41.7866 ms is only because the calculation gradually gets slower as the number of samples increase. This can be mitigated by limiting the number of past samples processed (in this case the window is one minute or 6k samples long). As soon as the window is saturated, the upwards trend in latency flattens and remains constant. This way, the difference in latency was further reduced by almost half to 21.6347 ms.

5. Conclusion

This paper presented a system for efficiently augmenting token-based access control with privacy-awareness without significantly impacting performance or utility. We described our implementation and demonstrated its practicality through experimental evaluations in terms or privacy gains and performance on cheap hardware.

5.1 Future Work

In this work we focused on time series sensor data and evaluated a single privacy metric. The clearest next step is expanding this work to explore application on structured data as well as other promising metrics such as similarity/diversity measures and autocorrelation for measuring seasonality. The latter can be compared with more nuanced methods such as Kullback-Leibler divergence [12] (relative entropy), cluster classification and regression analysis. Capacity [6] can similarly be explored as a measure of loss of anonymity.

Independent of the privacy metrics adapted to our system, our multiplexer component (figure 2) can be expanded to smoothly interpolate between granularity streams by assigning weights to each stream that sum to one.

Similarly, different user-definable policies for how our system reacts to passing thresholds can be explored. Instead of modifying granularity, possibilities include entirely blocking access, repeating old samples, adding noise, or generating fake “safe” data through runtime supervised learning (such as with an LSTM neural network). Each of these can be evaluated against each other in terms of privacy gains and performance.

Our work shows that there is a lot of potential in the space of information-theoretic, context-independent, real-time privacy awareness for access control on the edge.

References

[1] ALMUHIMEDI, H., SCHAUB, F., SADEH, N., ADJERID, I., ACQUISTI, A., GLUCK, J., CRANOR, L. F., AND AGARWAL, Y. Your location has been shared 5,398 times!: A field study on mobile app privacy nudging. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems* (2015), ACM, pp. 787–796.

[2] BEZZI, M. Expressing privacy metrics as one-symbol information. In *Proceedings of the 2010 EDBT/ICDT Workshops* (2010), ACM, p. 29.

[3] BEZZI, M. An information theoretic approach for privacy metrics. *Trans. Data Privacy 3*, 3 (2010), 199–215.

[4] BIRGISSON, A., POLITZ, J. G., ÚLFAR ERLINGSSON, TALY, A., VRABLE, M., AND LENTCZNER, M. Macaroons: Cookies with contextual caveats for decentralized authorization in the cloud. In *Network and Distributed System Security Symposium* (2014).

[5] CHAKRABORTY, S., SHEN, C., RAGHAVAN, K. R., SHOUKRY, Y., MILLAR, M., AND SRIVASTAVA, M. ipshield: a framework for enforcing context-aware privacy. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)* (2014), pp. 143–156.

[6] CHATZIKOKOLAKIS, K., PALAMIDESSI, C., AND PANANGADEN, P. Anonymity protocols as noisy channels. *Information and Computation 206*, 2-4 (2008), 378–401.

[7] CHAUDHRY, A., CROWCROFT, J., HOWARD, H., MADHAVAPEDDY, A., MORTIER, R., HADDADI, H., AND MCAULEY, D. Personal data: Thinking inside the box. In *Proceedings of The Fifth Decennial Aarhus Conference on Critical Alternatives* (2015), AA '15, Aarhus University Press, pp. 29–32.

[8] CHEN, T., CHAABANE, A., TOURNOUX, P. U., KAAFAR, M.-A., AND BORELI, R. How much is too much? leveraging ads audience estimation to evaluate public profile uniqueness. In *International Symposium on Privacy Enhancing Technologies Symposium* (2013), Springer, pp. 225–244.

[9] DALENIUS, T. Finding a needle in a haystack or identifying anonymous census records. *Journal of official statistics 2*, 3 (1986), 329.

[10] FUNG, B., WANG, K., CHEN, R., AND YU, P. S. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (CSUR) 42*, 4 (2010), 14.

[11] GREVELER, U., GLÖSEKÖTTERZ, P., JUSTUSY, B., AND LOEHR, D. Multimedia content identification through smart meter power usage profiles. In *Proceedings of the International Conference on Information and Knowledge Engineering (IKE)* (2012), The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), p. 1.

[12] JOYCE, J. M. Kullback-leibler divergence. In *International Encyclopedia of Statistical Science*. Springer, 2011, pp. 720–722.

[13] KALOGRIDIS, G., EFTHYMIOU, C., DENIC, S. Z., LEWIS, T. A., AND CEPEDA, R. Privacy for smart meters: Towards undetectable appliance load signatures. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on* (2010), IEEE, pp. 232–237.

[14] KIM, Y., NGAI, E. C.-H., AND SRIVASTAVA, M. B. Cooperative state estimation for preserving privacy of user behaviors in smart grid. In *Smart Grid Communications (SmartGridComm), 2011 IEEE International Conference on* (2011), IEEE, pp. 178–183.

[15] KOLTER, J. Z., AND JOHNSON, M. J. Redd: A public data set for energy disaggregation research.

[16] LI, N., LI, T., AND VENKATASUBRAMANIAN, S. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on* (2007), IEEE, pp. 106–115.

[17] LICHMAN, M. UCI machine learning repository, 2013.

[18] MACHANAVAJJHALA, A., GEHRKE, J., KIFER, D., AND VENKITASUBRAMANIAM, M. l-diversity: Privacy beyond k-anonymity. In *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on* (2006), IEEE, pp. 24–24.

[19] NARAYANAN, A., AND SHMATIKOV, V. Robust de-anonymization of large sparse datasets. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on* (2008), IEEE, pp. 111–125.

[20] SWEENEY, L. Simple demographics often identify people uniquely. *Health (San Francisco) 671* (2000), 1–34.

[21] SWEENEY, L. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10*, 05 (2002), 557–570.

[22] WAGNER, I., AND ECKHOFF, D. Technical privacy metrics: a systematic survey. *arXiv preprint arXiv:1512.00327* (2015).