Workload Modeling for Security and Privacy in Databases

Gokhan Kul, Ph.D.
Assistant Professor

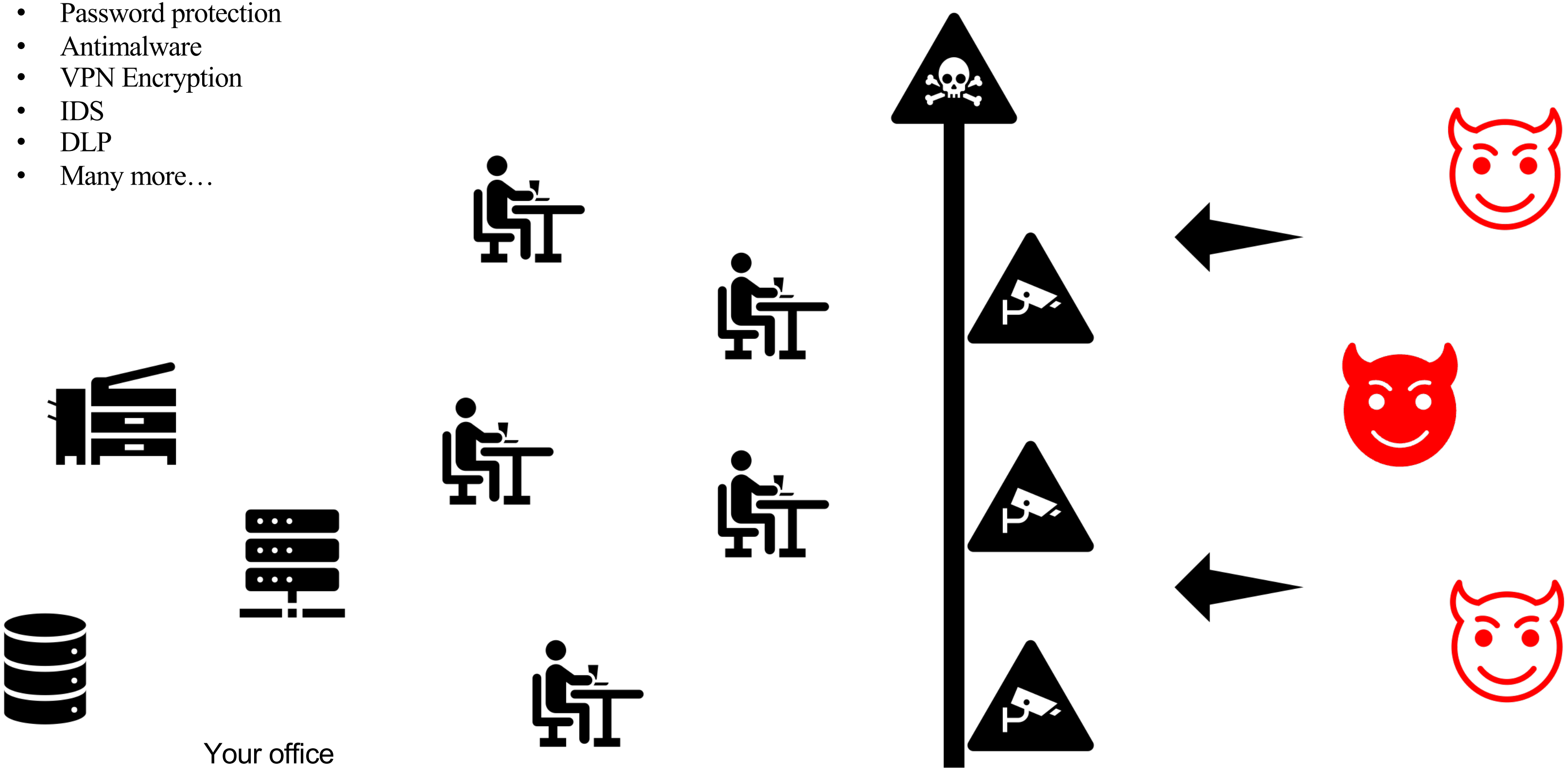
UMass | Dartmouth

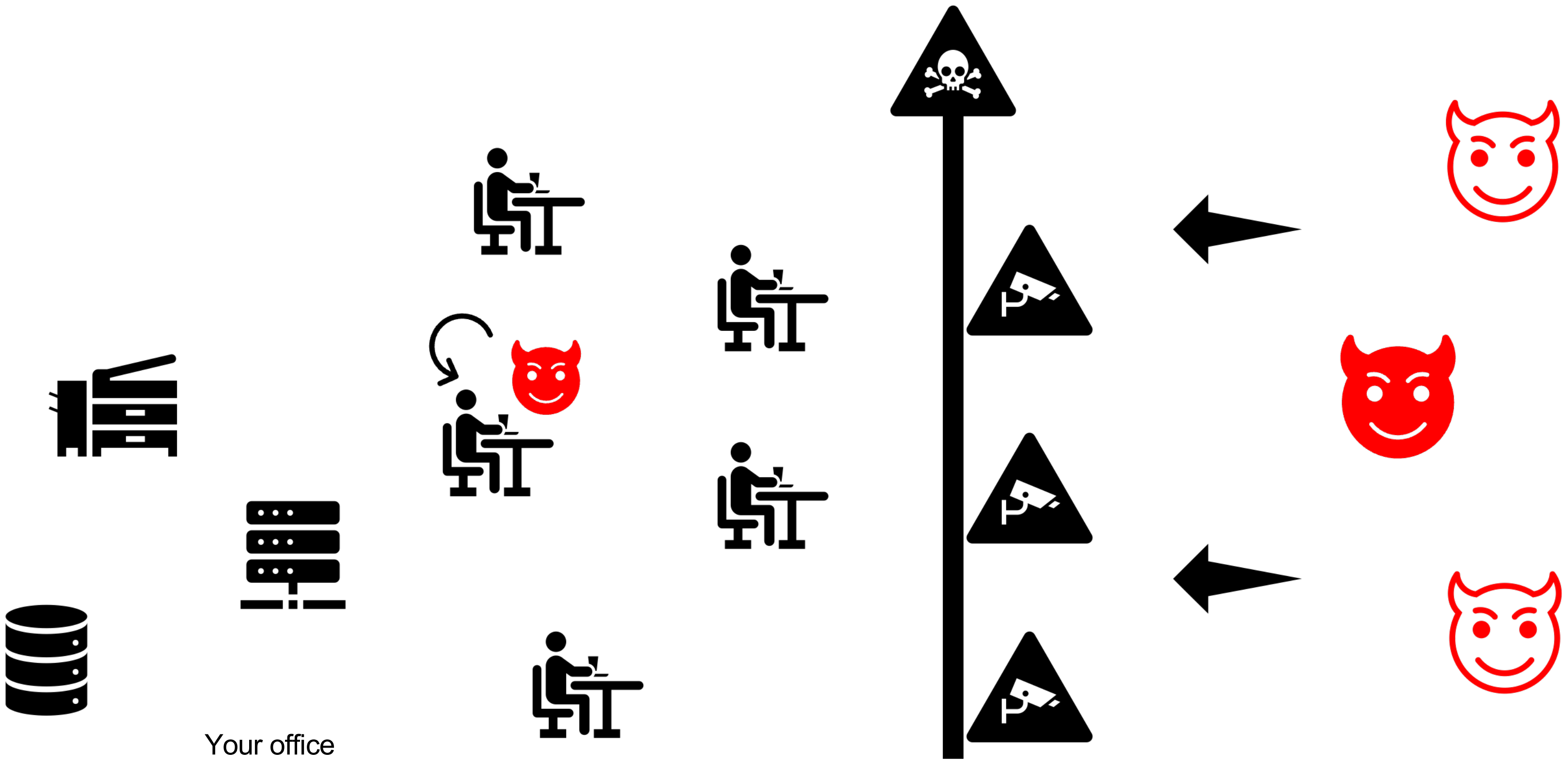
Outline

- Insider Threat Overview
- Workload Modeling
- PocketData Project
- Insider Threats Project
- Other Projects
- Background

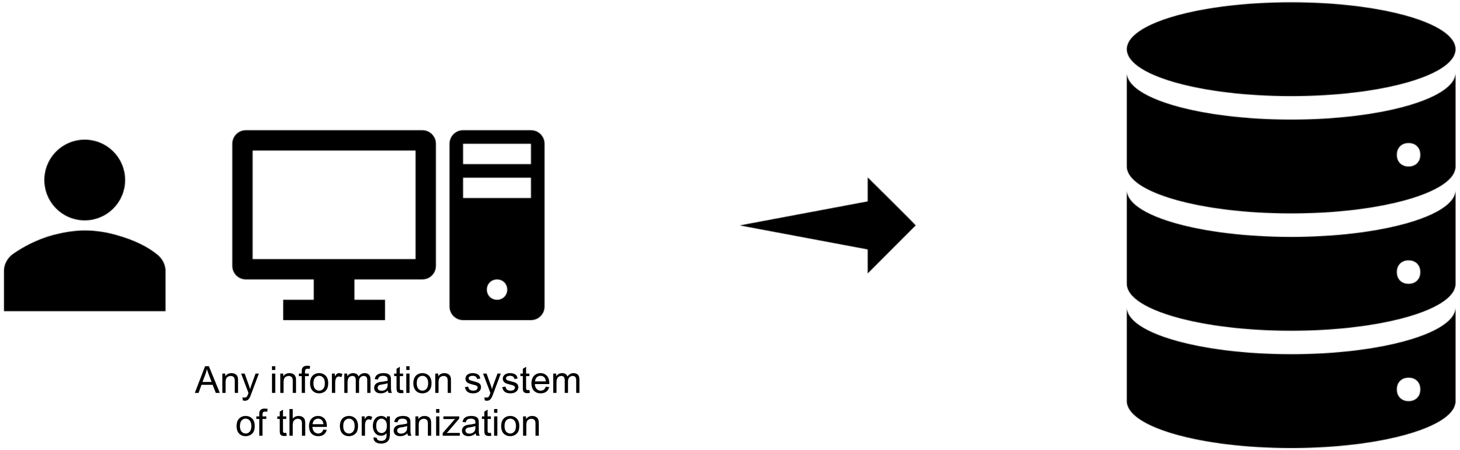
In a standard office environment, there are strong defense mechanisms:

- Firewall
- Password protection
- Antimalware
- VPN Encryption
- IDS
- DLP
- Many more…




Your office


If they don't pay me what I'm worth, I know how I can take it
OMG! What did I do? Did I just send $5000 to the wrong account?
Image credit: shutterstock.com


Any information system
of the organization

Outline

- Insider Threat Overview
- Workload Modeling
- PocketData Project
- Insider Threats Project
- Other Projects
- Background

Traditional Workload Modeling

Question Asked:

What kind of queries do we receive?

Traditional Workload Modeling

Indexes

Joins

Question Asked:

What should we focus on to increase performance?

Database Structure

Primary Keys

Foreign Keys

Application: Benchmarks

Measure **Throughput** & **Latency**

Latency:

is the time required to perform one single action

Throughput:

is the number of such actions executed or results produced per unit of time

Application: Benchmarks

Which one is more important at database performance?

Latency vs Throughput

Hold that thought

Improvement Points

No attention to the activity performed

SELECT on a table with 10 rows vs. 1.000.000 rows

1 access attempt to a row vs 1.000 access attempt

No attention to what the user intends to do

Bring me a customer who’s a **frequent** customer

vs bring me a customer who **last shopped last week**

Outline

- Insider Threat Overview
- Workload Modeling
- PocketData Project
- Insider Threats Project
- Other Projects
- Background

PocketData: Databases on Smartphones

Databases Are Single Client

Latency, Not Throughput, Matters

Workloads Are Bursty

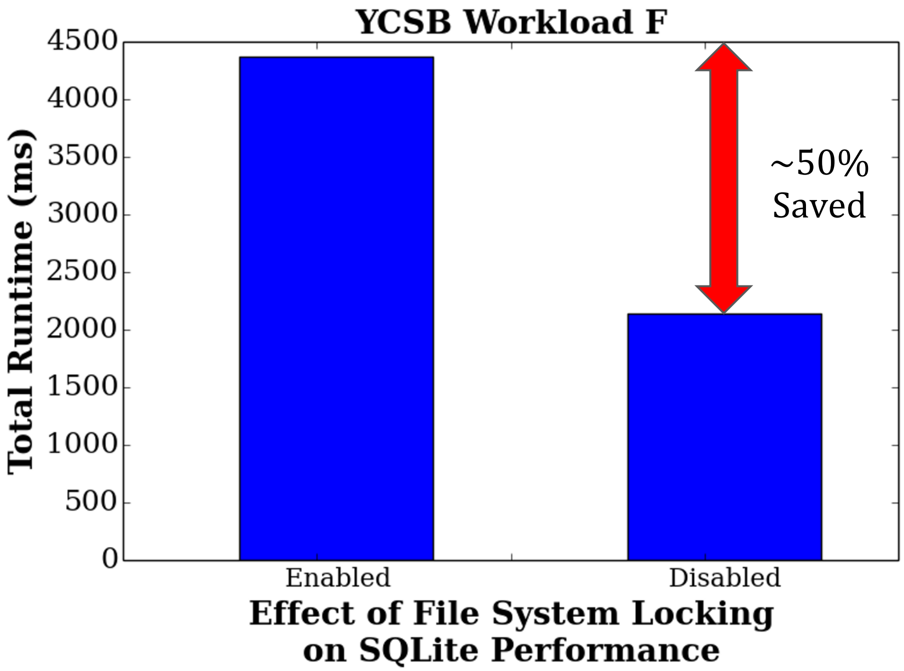
Representative Benchmarks Matter

With Great Differences Come Great Opportunities

Be Smart and Lose the I: ACID => ACD

Atomicity,
Consistency,
Isolation,
Durability

The Cost of Database Isolation



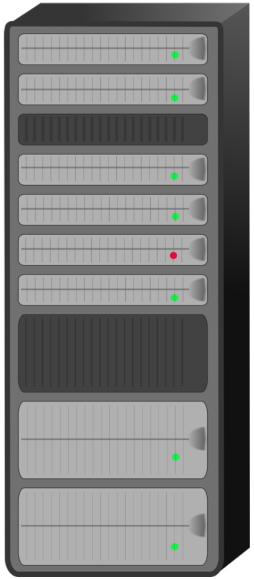
Can we design databases with weaker ACID (more Basic) semantics?

Optimize for Burst Response

Some observed throughputs:

Optimize for Burst Response

Some observed throughputs:



36,000 tpm* ~ 600 tps*

112,000 tpm*

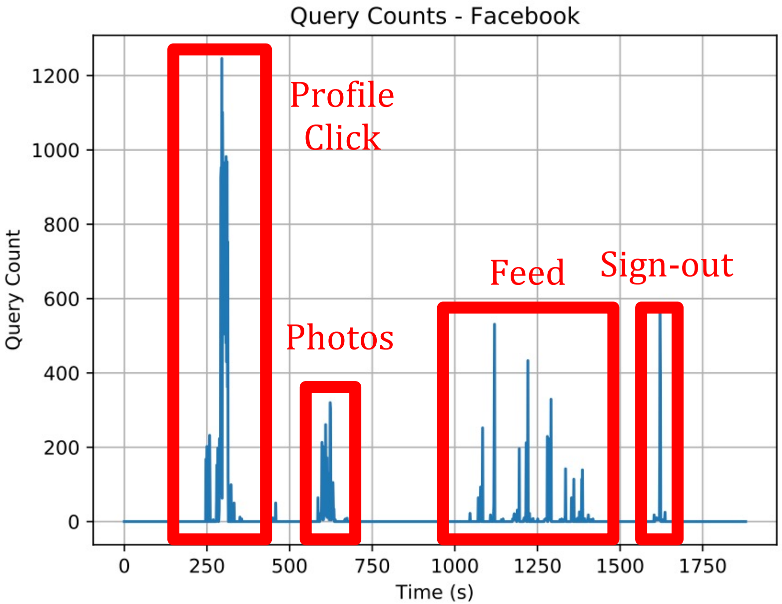
*Oliver Kennedy, Jerry Antony Ajay, Geoffrey Challen, and Lukasz Ziarek. 2015. Pocket Data: The Need for TPC-MOBILE. In TPC-TC.

*http://www.tpc.org/tpcc/results/tpcc_results.asp

Optimize for Burst Response

A typical database operation pattern on a mobile device:

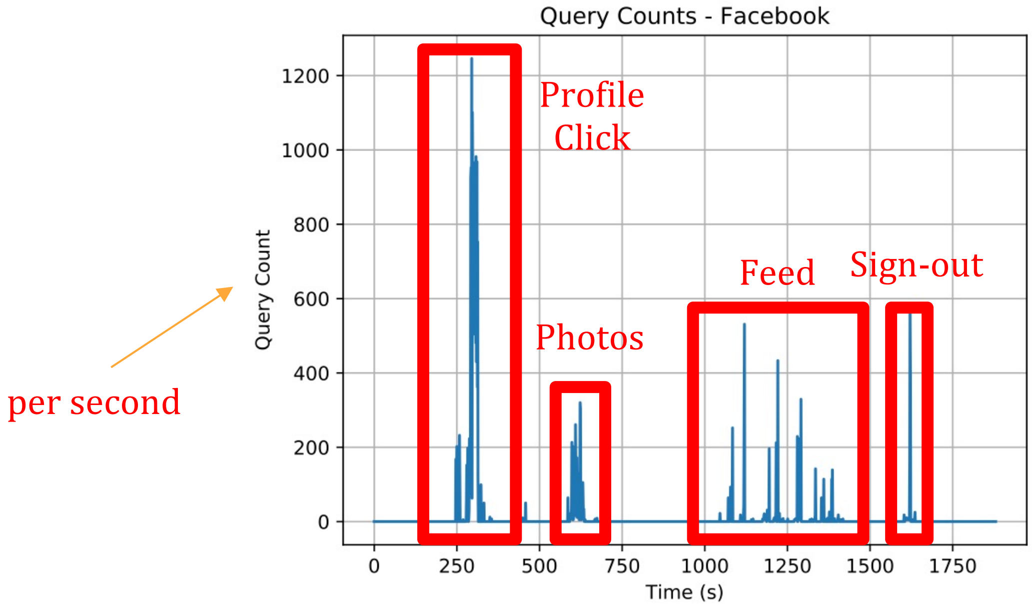
per second



Since we don't have to worry about throughput,
How much can we improve latency?

Security Implications

A typical database operation pattern on a mobile device:



How does a burst change for each user? Can we distinguish different users? Is it possible to perform a side channel attack? Can defense mechanisms respect privacy?

PocketData: Experiments Performed

Two Phases:

(1) 11 lab members

(2) 56 phones deployed in the wild

Next

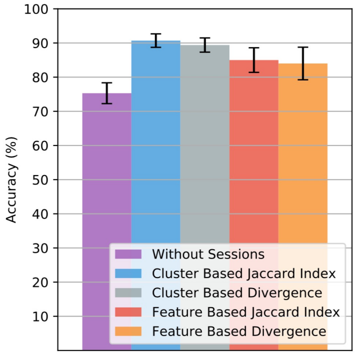
PocketData: Grant Proposal

NSF CISE Community Research Infrastructure (CCRI) (January 2022):

Let’s create a stable testbed and distribute these phones to a larger group

(New data servers, new software versions, etc)

PocketData: Initial Results



Procedures trigger sequence of queries:

First few queries of a burst helps predicting the rest of the queries

Behavior patterns can distinguish users from each other

PocketData: By-Products in Software Engineering

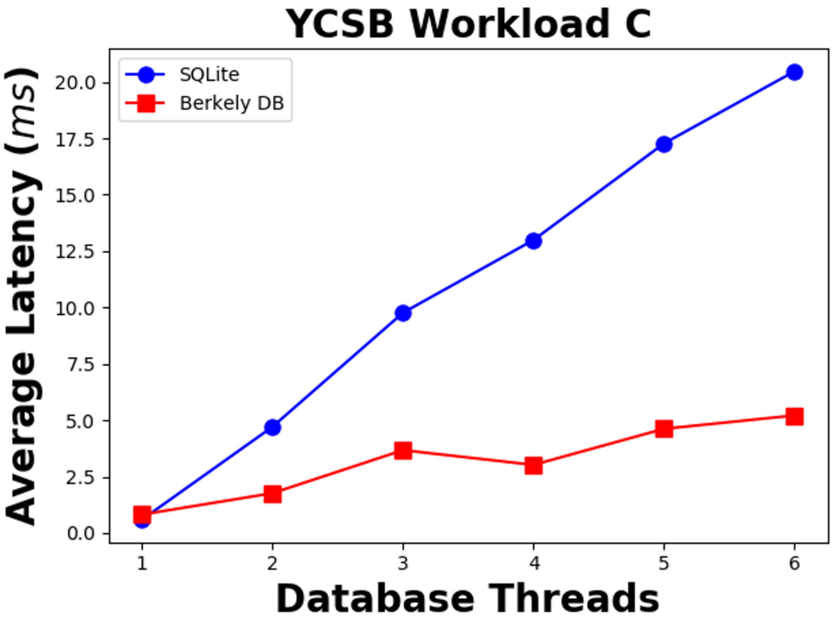
Procedure latency:

How updates in the software will affect the procedure latency?

Should I push this update or not to the software?

New, Representative Benchmarks

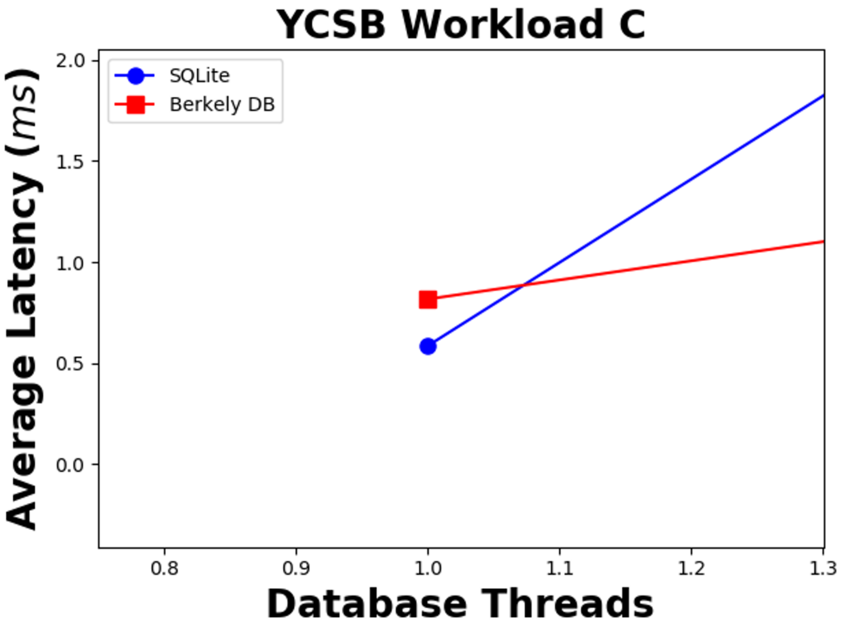
A typical database comparison study:



But scaling doesn't matter on phones.

New, Representative Benchmarks

Databases are per-app



The corner case is the common case

Contributers

Gokhan Kul (Umass Darmouth)
Gourab Mitra (Datometry)
Carl Nuessle (UB)
Darshana Balakrishnan (UB)
Lukasz Ziarek (UB)
Oliver Kennedy (UB)

Interested Community

Arnab Nandi (Ohio State University)
Richard Hipp (SQLite)
Stratos Idreos (Harvard University)
More…

Outline

- Workload Modeling
- PocketData Project
- Insider Threats Project
- Other Projects
- Background

Insider Threat

A trusted person (TP): Employee, Contractors, Vendors

TP may misuse legitimate access:

- Unintentional – incompetency, amateur behavior
- Intentional – Traitor
- Collusion

TP may obtain unauthorized access:

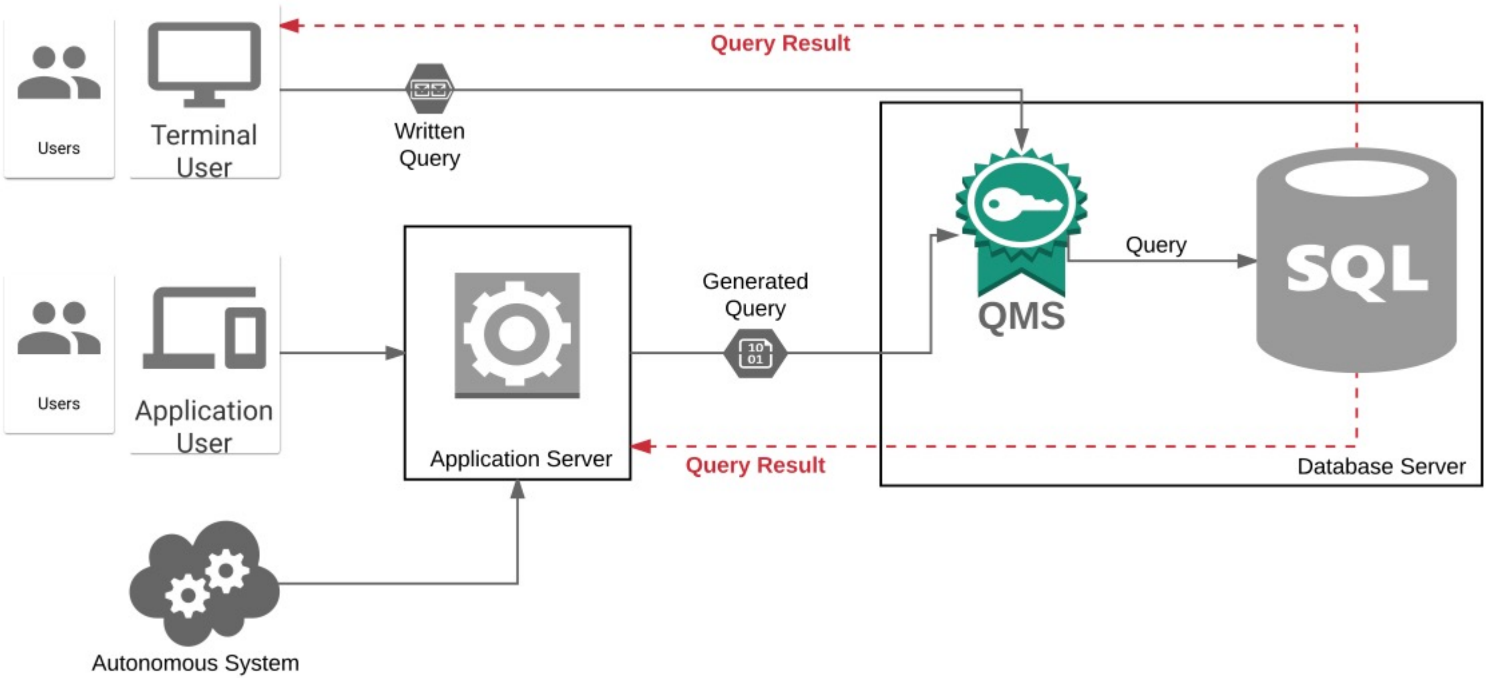
- Masquerading

The Problem

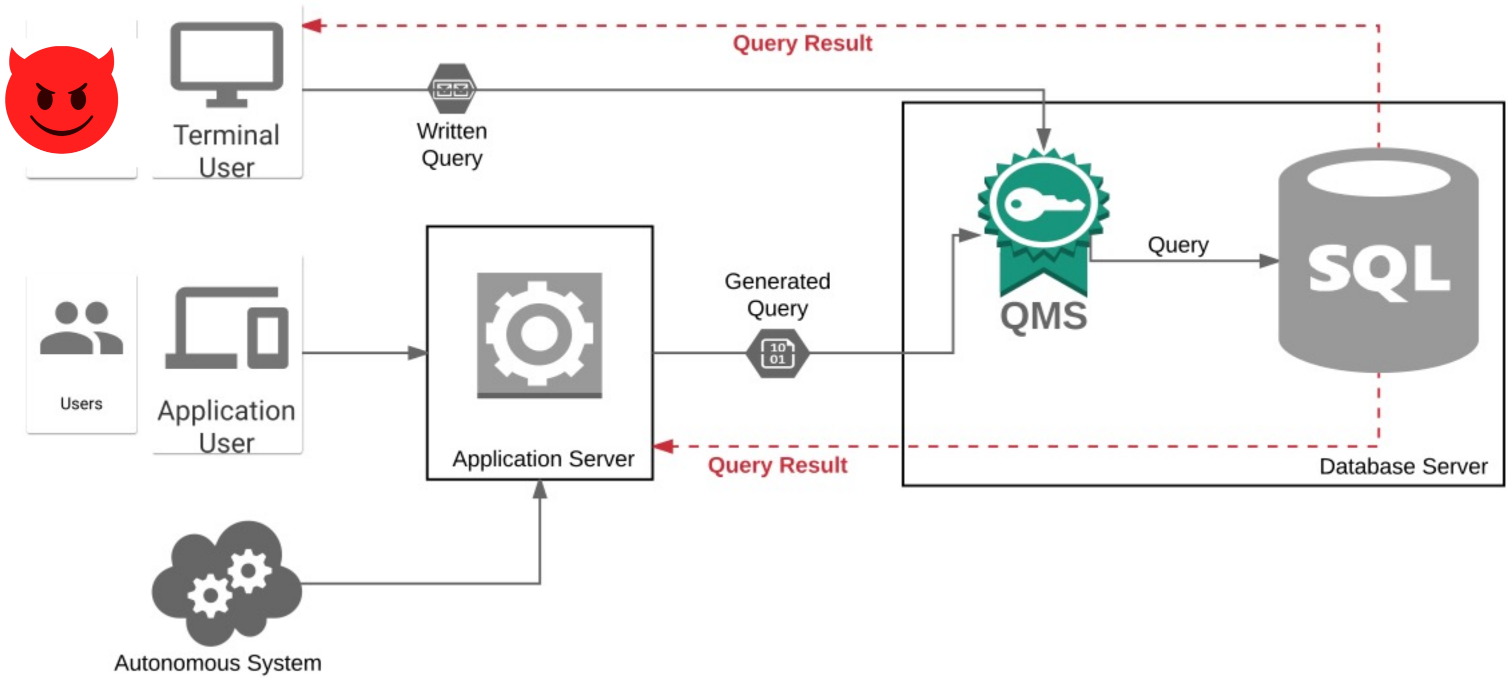
The attackers know you and you trust them

They are inside (almost) all of the security layers

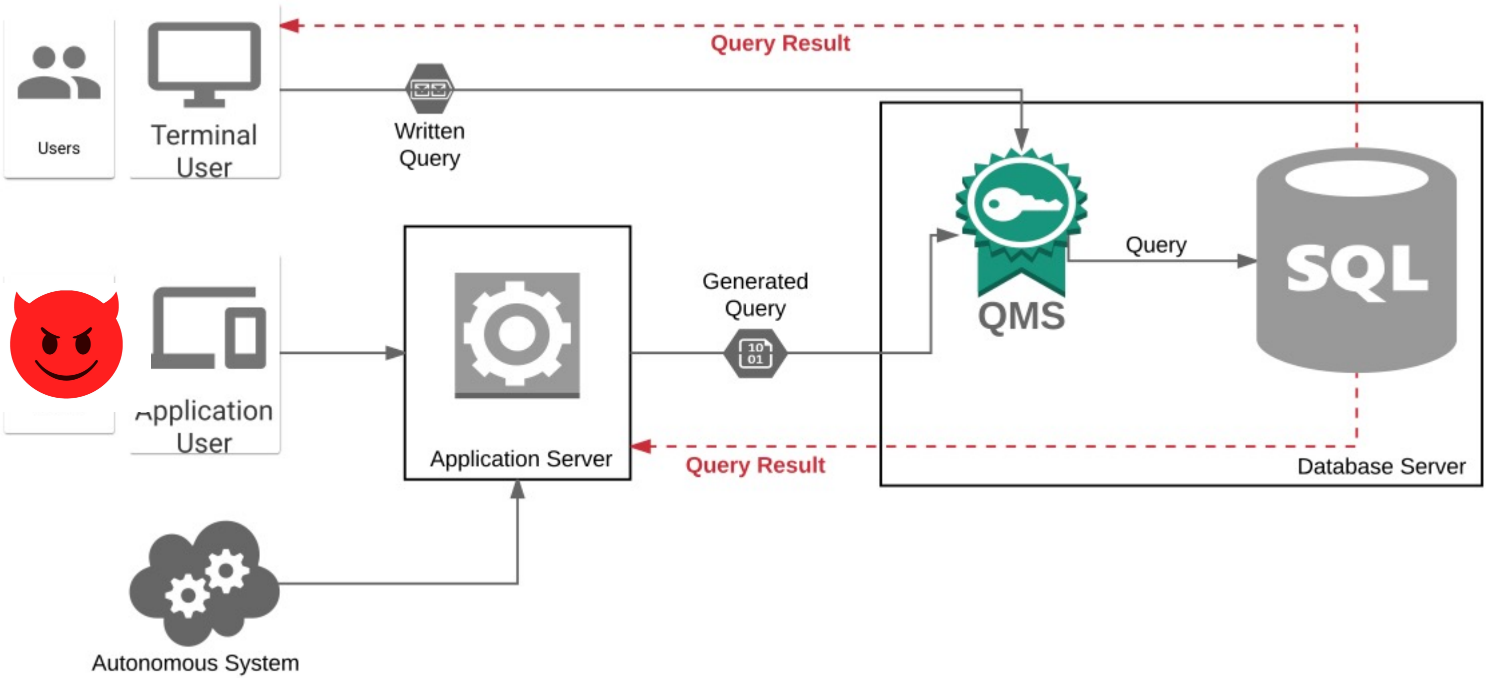
Data Access Architecture



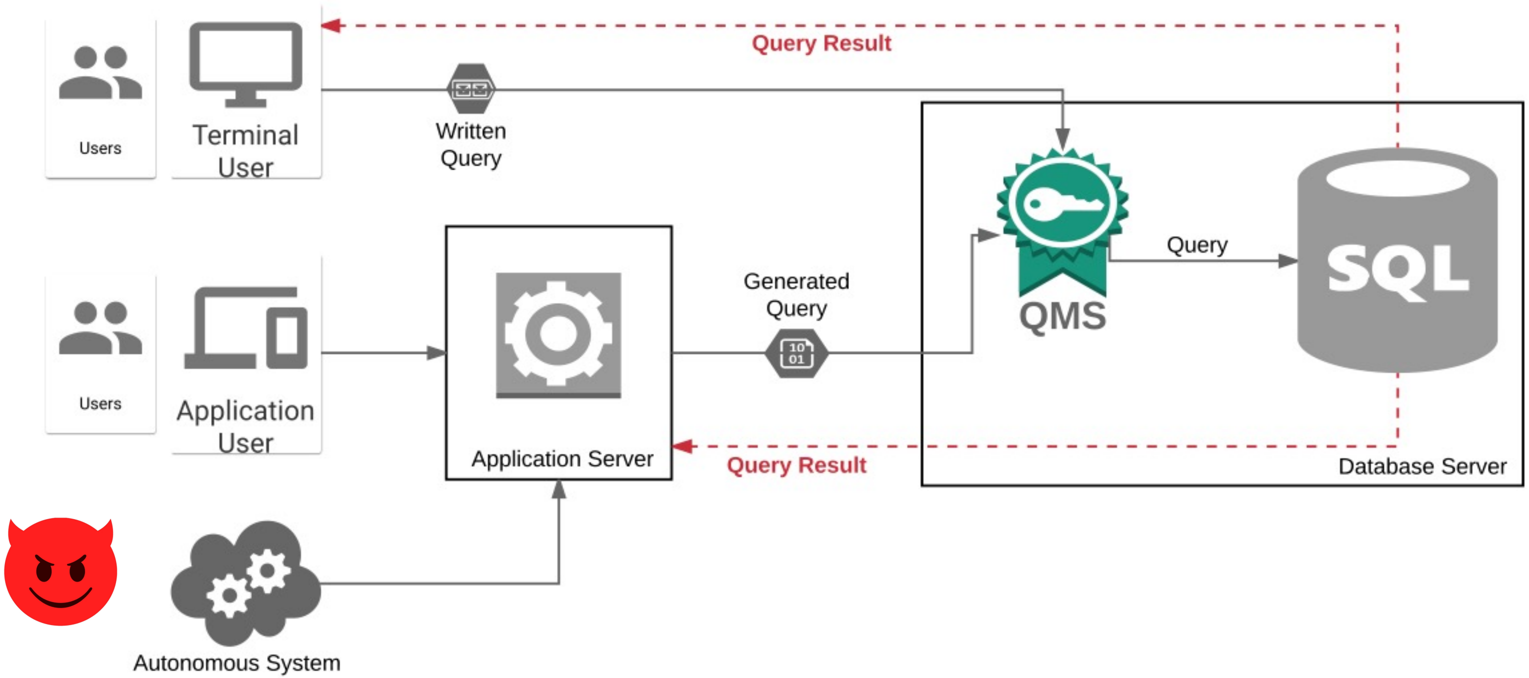
Data Access Architecture



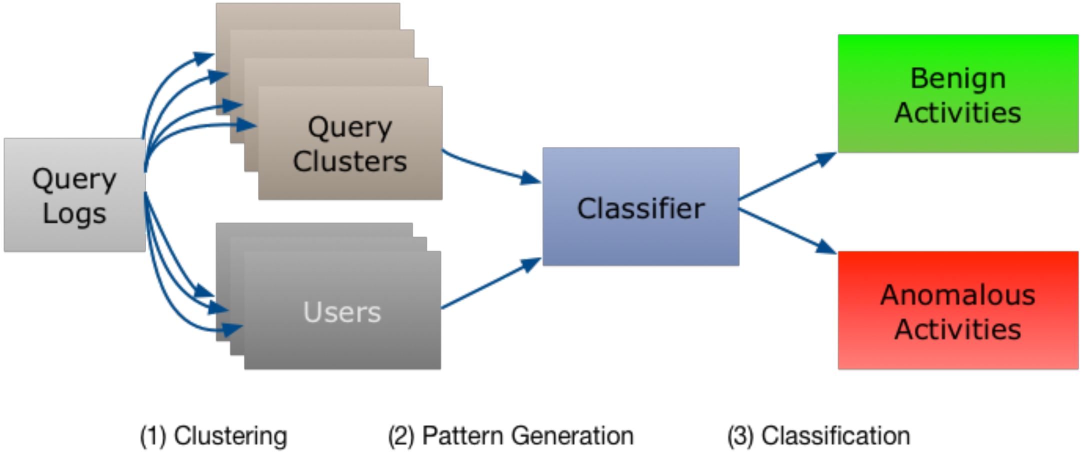
Data Access Architecture



Data Access Architecture



Method



* Gokhan Kul, Duc Luong, Ting Xie, Patrick Coonan, Varun Chandola, Oliver Kennedy, and Shambhu Upadhyaya. *Ettu: Analyzing Query Intents in Corporate Databases*, In Proceedings of the 25th International Conference Companion on World Wide Web (WWW'16). Montreal, Canada

Improvement Points

How to make anomaly detection better?

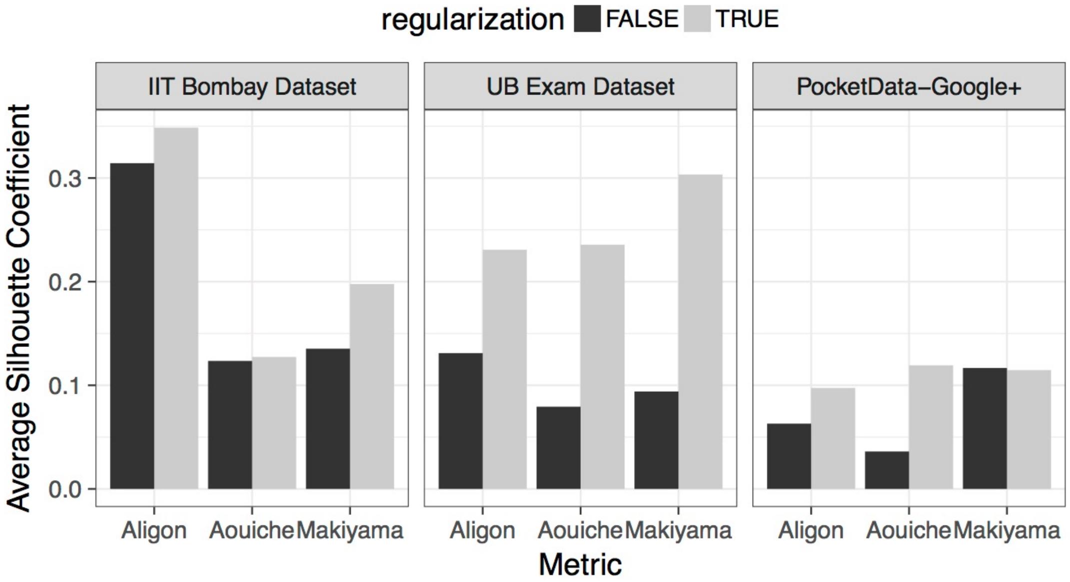
(1) Find ideal similarity metrics for query clustering

(2) Standardize (called Regularization) queries

(3) Exploit user's distinct behavior

(4) Exploit changes in user's habits

Improvement Point (1) & (2)

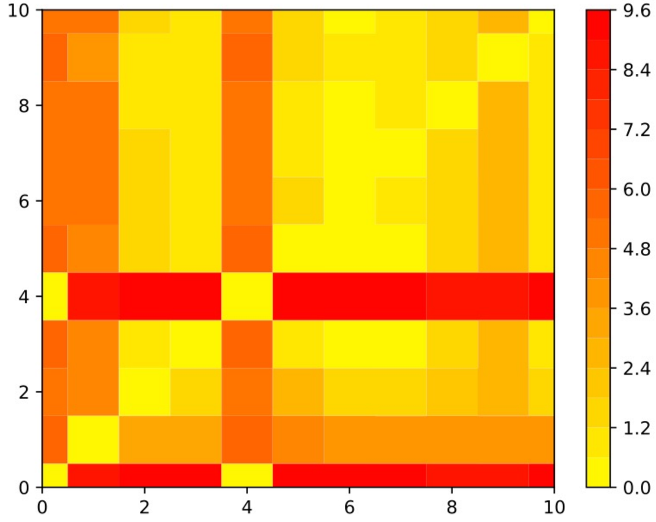


Gokhan Kul, Duc Luong, Ting Xie, Varun Chandola, Oliver Kennedy, and Shambhu Upadhyaya. *Similarity Metrics for SQL Query Clustering*, IEEE Transactions on Knowledge and Data Engineering (TKDE), 2018.

Improvement Point (3)

Can we distinguish two users based on their activity patterns?

Google+ application, 2M SQL queries, 11 users, 1 month

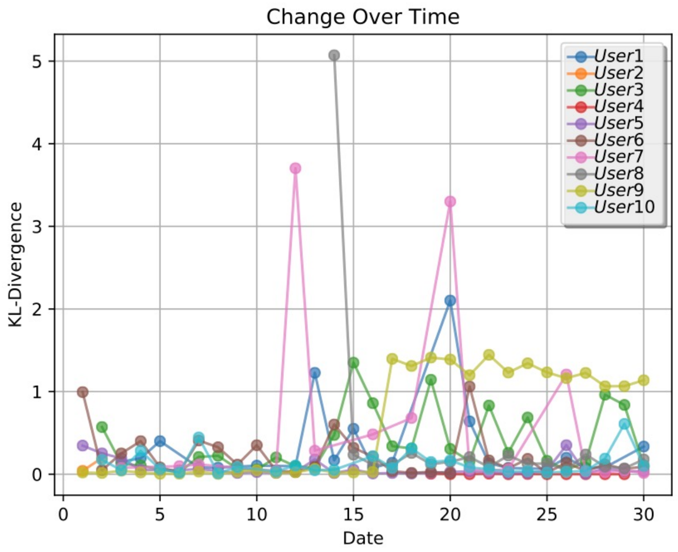


KL-Divergence score heat map for 11 Google+ users

Improvement Point (4)

Can we profile a user based on changing habits?

Google+ application, 2M SQL queries, 11 users, 1 month



Behavior change based on SQL Queries for 11 Google+ users

Data from PocketData

Application	# of Queries
Complete Dataset	45,090,798
Facebook	1,212,779
Google+	2,040,793
Hangouts	974,349
Google Play Services	14,813,949
Media Storage	13,592,982

Simulated Attacks (Queries written by us)

		Ideal Threshold		Behavior Drift	
	# of Attacks Performed	Detected	Success	Detected	Success
Facebook	105	97	92.4%	98	**93.3%**
Google+	225	202	89.8%	214	**95.1%**
Hangouts	239	206	**86.2%**	206	**86.2%**
Google Play	282	261	92.6%	267	**94.7%**
Media Storage	282	251	89.0%	259	**91.8%**

Real Workload Attacks
(Queries injected from other users)

		Ideal Threshold		Behavior Drift	
	# of Attacks Performed	Detected	Success	Detected	Success
Facebook	315	290	**92.1%**	283	89.8%
Google+	2025	1817	89.7%	1818	**89.7%**
Hangouts	2201	1842	83.7%	1853	**84.2%**
Google Play	2583	2066	80.0%	2092	**81.0%**
Media Storage	2583	2099	81.3%	2105	**81.5%**

Contributers

Gokhan Kul (Umass Dartmouth)
Shambhu Upadhyaya (UB)
Varun Chandola (UB)
Oliver Kennedy (UB)
Ting Xie (UB)
Duc Luong (UB)
Long Nyugen (UMich)

Outline

- Workload Modeling
- PocketData Project
- Insider Threats Project
- Background

Background

Research

Cybersecurity of Database & Cloud Systems