Relational Database Similarity Detection via Network Traffic Analysis

Rares Folea, Emil Slusanschi, Mihai Dascalu

Department of Computer Science and Engineering,
Faculty for Automatic Control and Computers,
National University of Science and Technology Politehnica Bucharest

Research objective

Present and evaluate a **methods for measuring similarities between complex software systems** where there is **no access to the internal structure of the system** , such as access to the source code or even compiled server binaries.

Similarity Detection

Techniques

- Traditional approaches
- Software Fingerprints
- Software Birthmarks
- Code Embeddings
- LLM-based

on top of ...

- code (in programming language)
- abstract syntax tree
- code (in binary language)
- profiling data
- sampling data
- OS data

Similarity Detection
(via Network Traffic Analysis)

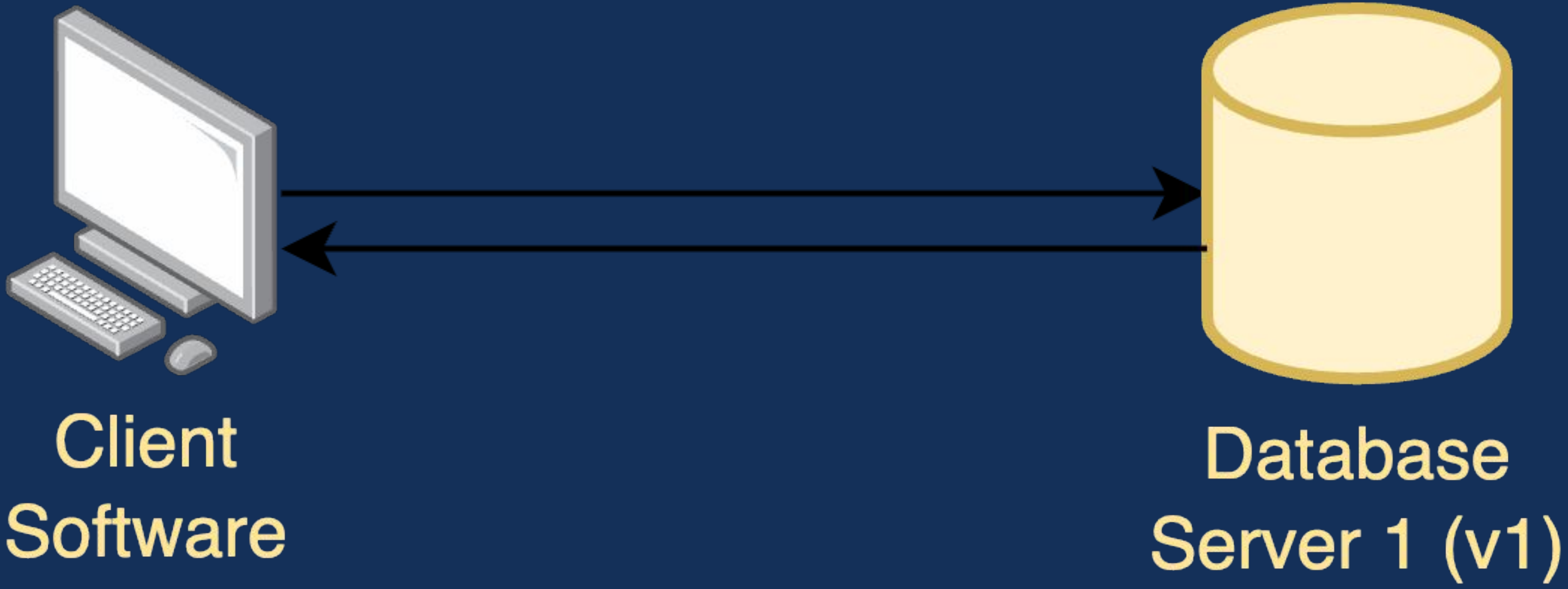
Techniques

- Traditional approaches
- Software Fingerprints
- Software Birthmarks
- Code Embeddings
- LLM-based

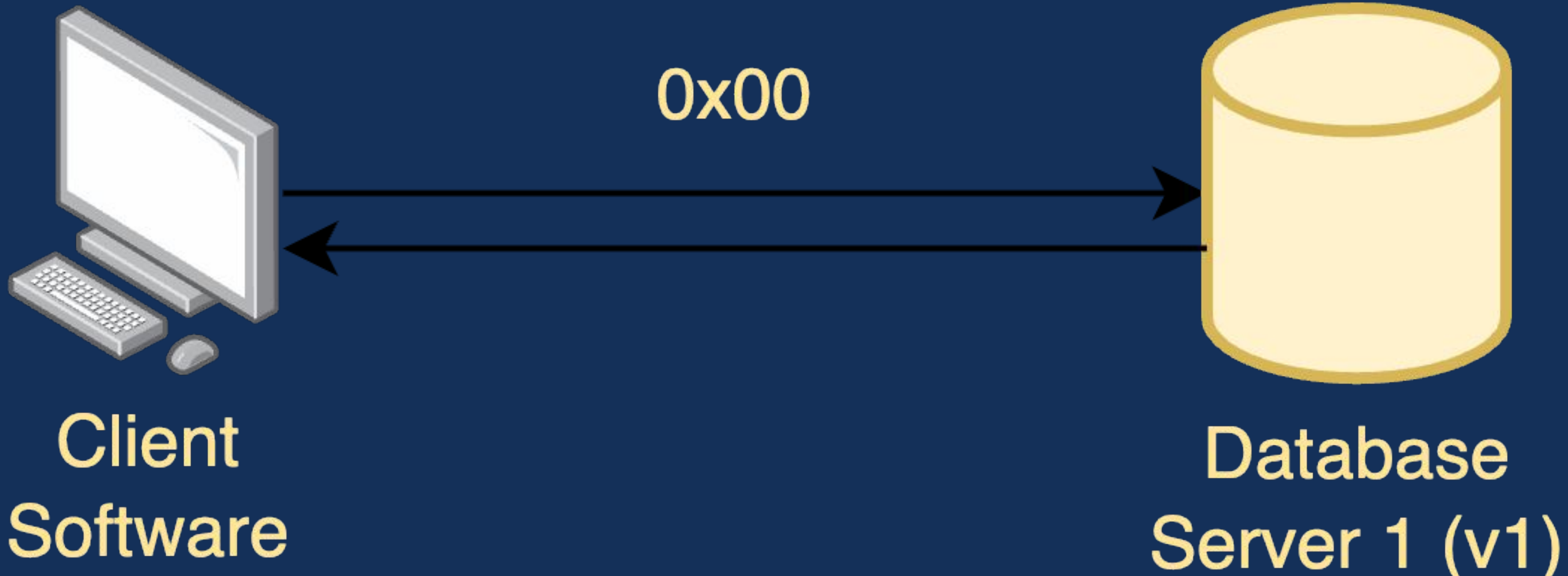
on top of …

- ~~code (in programming language)~~
- ~~abstract syntax tree~~
- ~~code (in binary language)~~
- profiling data
- sampling data
- OS data (network card)

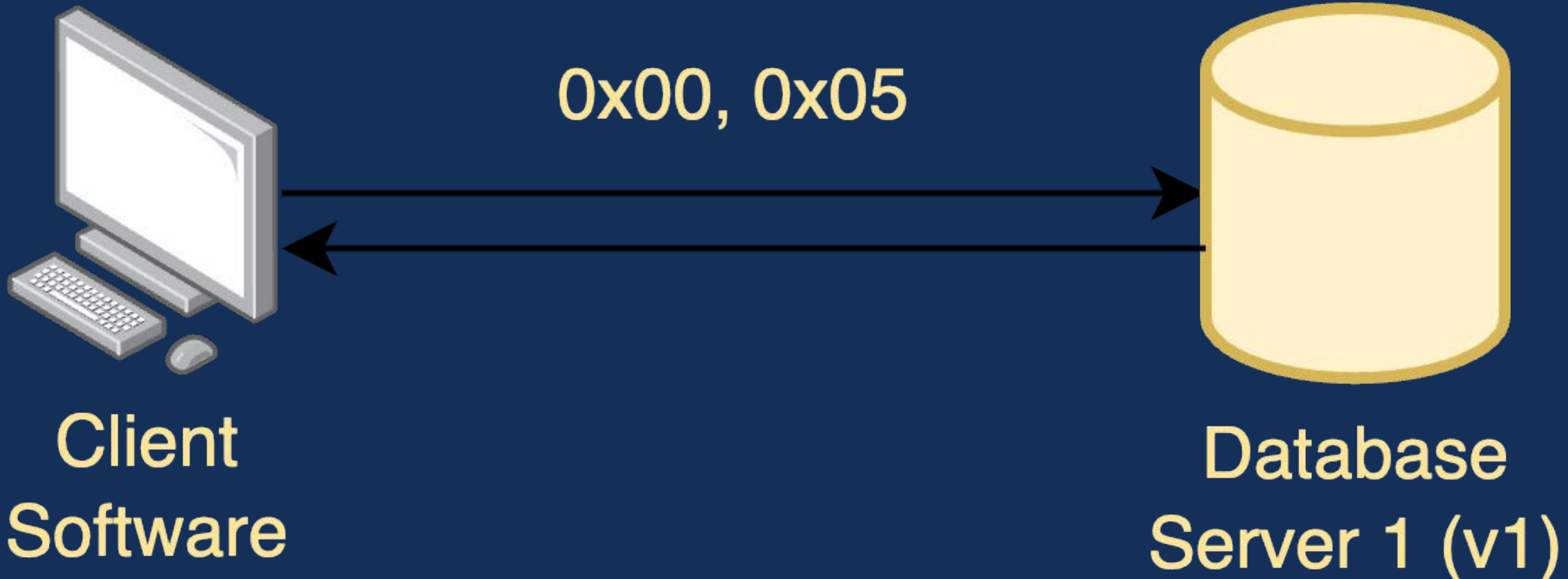
Available monitoring tools



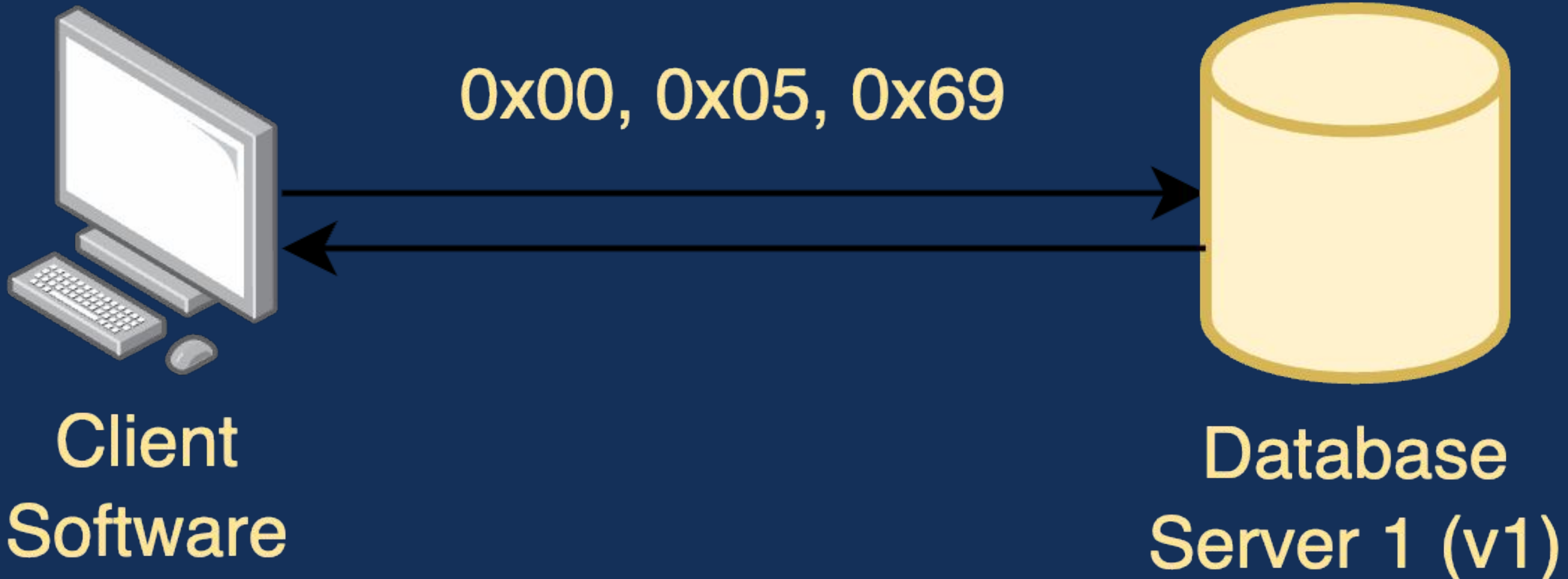
Available monitoring tools



Available monitoring tools



Available monitoring tools

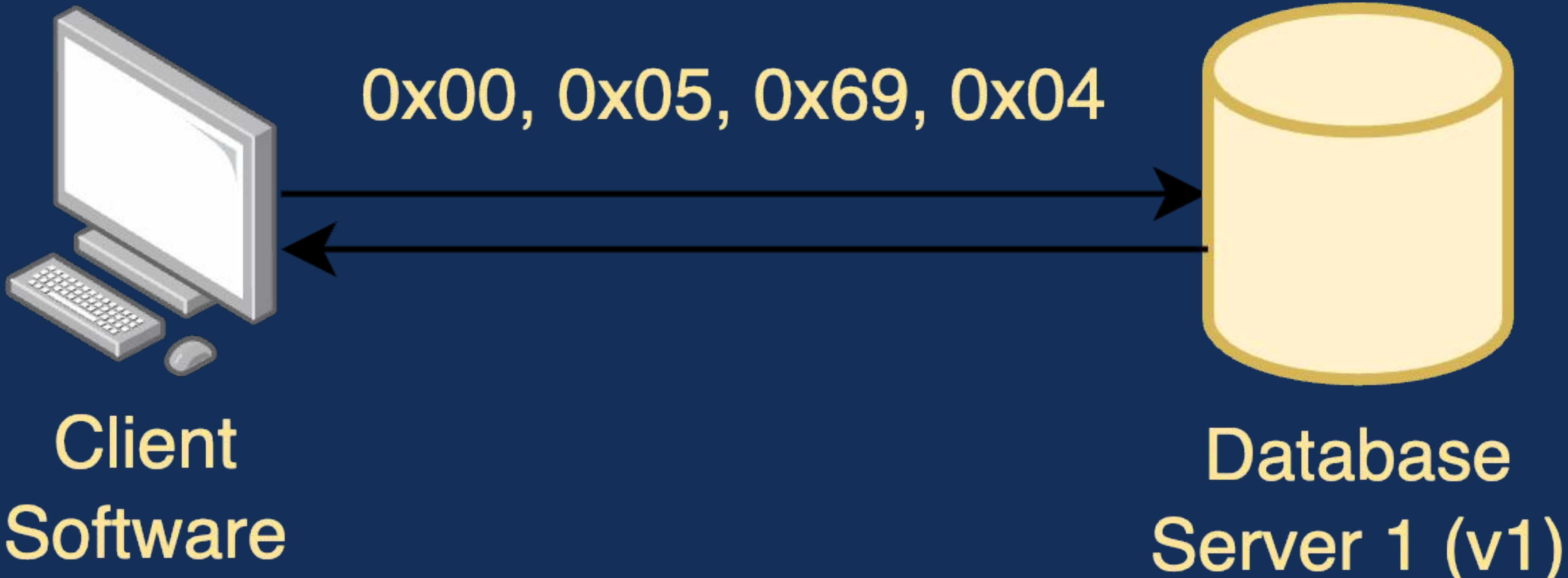


0x00, 0x05, 0x69

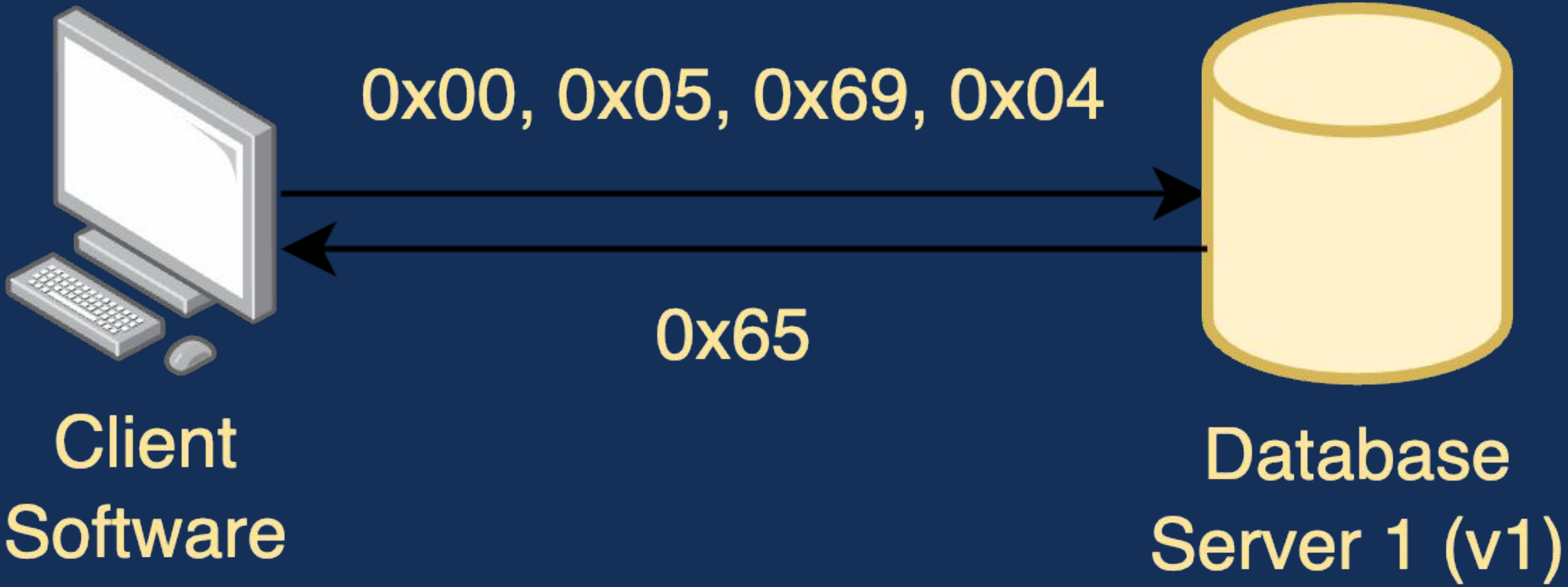
Client Software

Database Server 1 (v1)

Available monitoring tools



Available monitoring tools



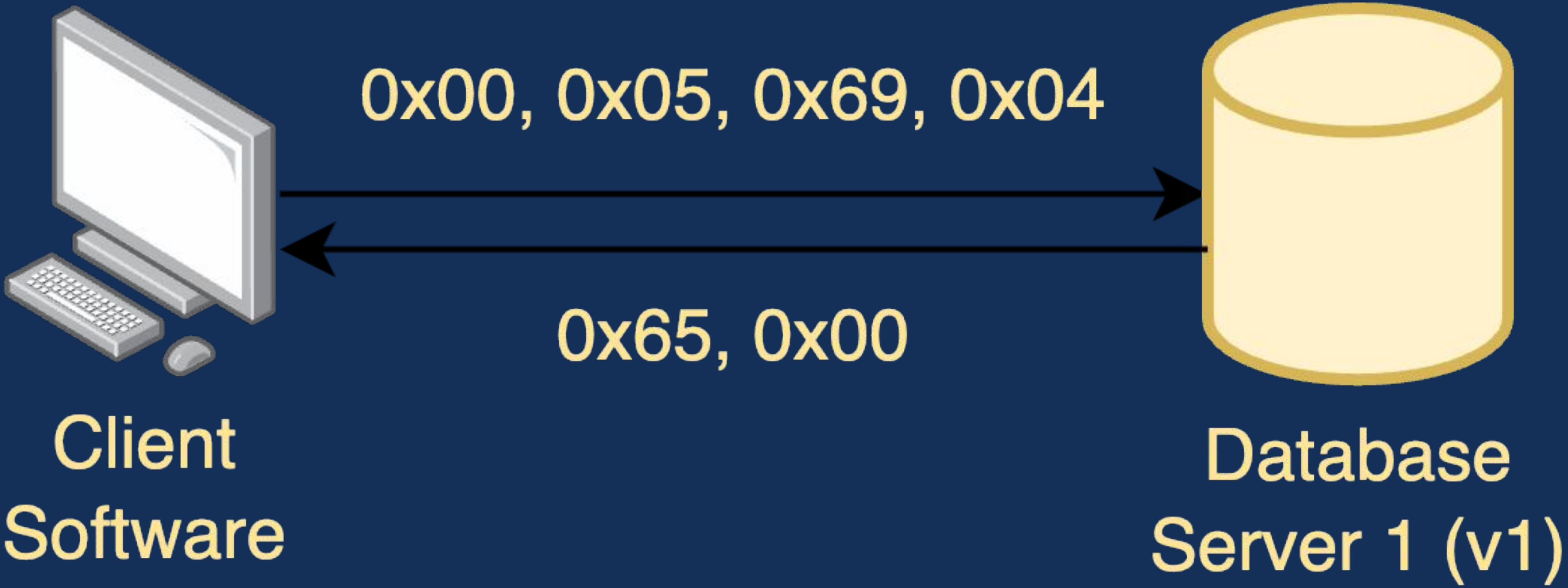
0x00, 0x05, 0x69, 0x04

0x65

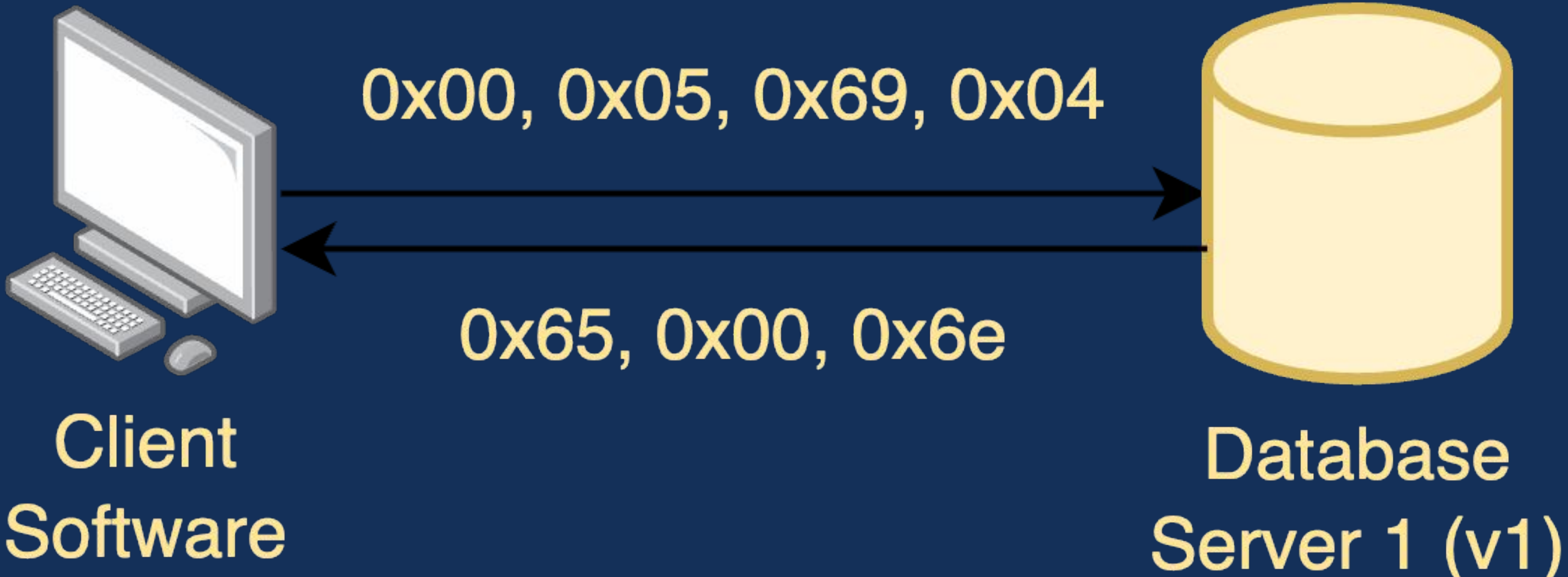
Client Software

Database Server 1 (v1)

Available monitoring tools

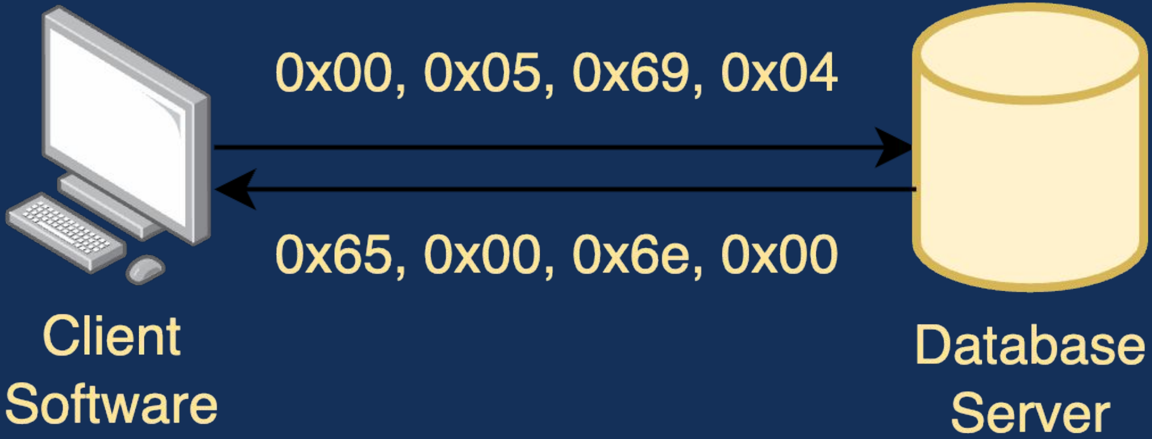


Available monitoring tools



Hypothesis

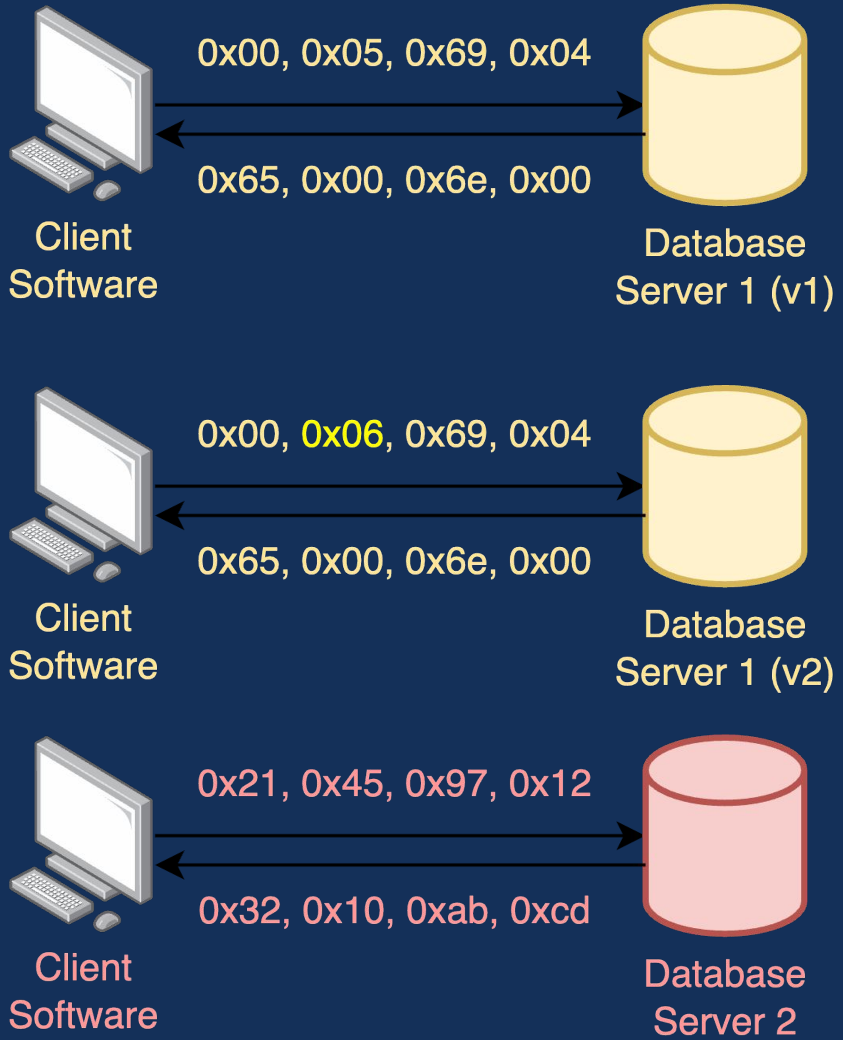
Similar software systems will exhibit comparable communication patterns at application layer, while different systems will show distinct ones.



Research scope

Similar database engines will exhibit comparable communication patterns, while different engines will show distinct patterns.

The research uses n-gram analysis (specifically 3-grams and 4-grams) of network traffic bytes and TF-IDF scores to quantify these similarities.



Analysed software

3 database types (MySQL, PostgreSQL, SQL Server)
20 major versions
2 deployments (official Docker images, Google Cloud SQL)

31 unique databases configurations

Corpus

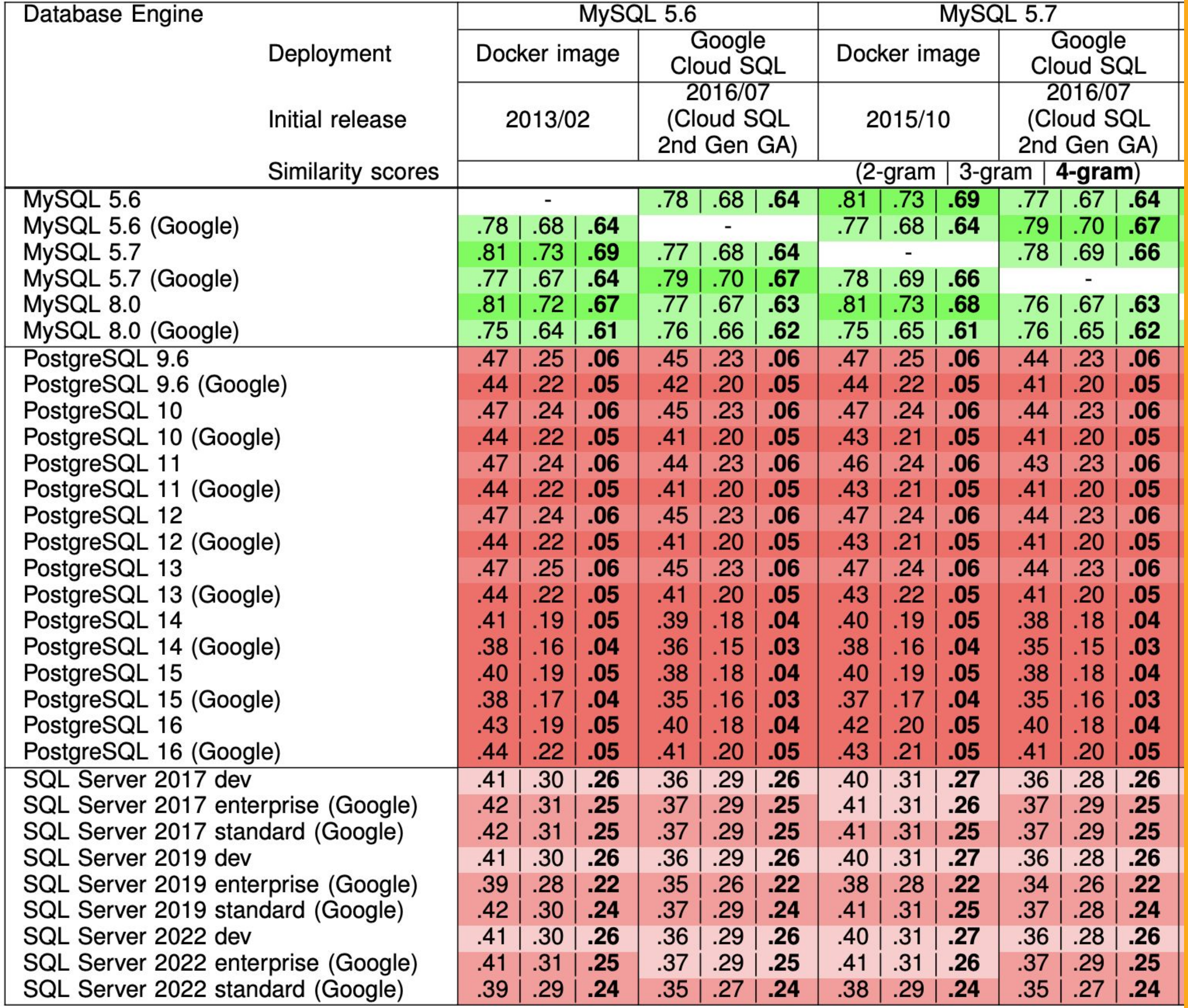
Several tens MB of data of captured database traffic capturing the interaction between clients and database engines during regular interactions (simple interactions, sysbench, custom benchmarks).

Similarity Score Function

4-grams : sequential word combinations

TF-IDF : word importance score

cosine distance : vector angle difference

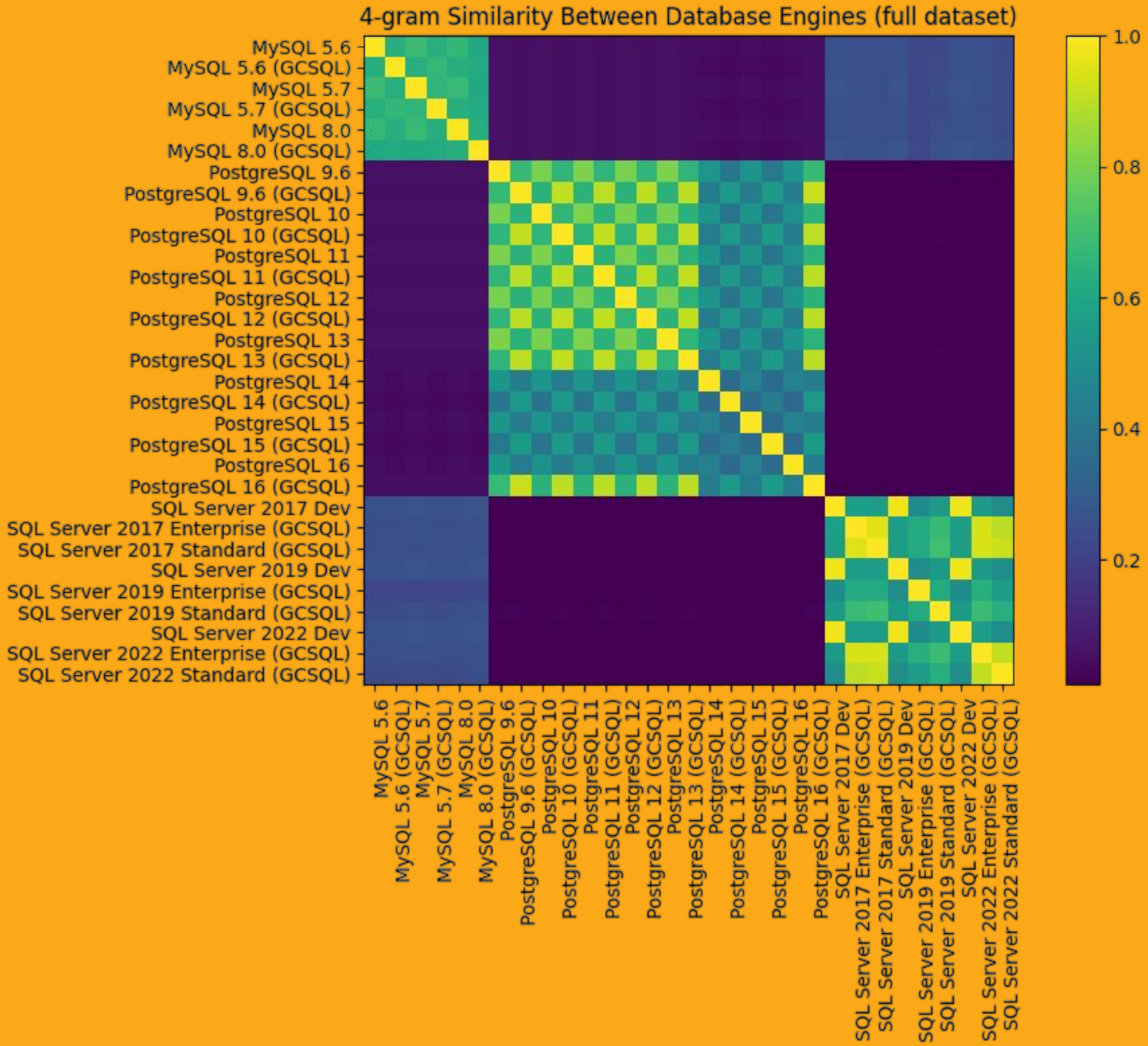


Database Engine	MySQL 5.6						MySQL 5.7					
Deployment	Docker image			Google Cloud SQL			Docker image			Google Cloud SQL		
Initial release	2013/02			2016/07 (Cloud SQL 2nd Gen GA)			2015/10			2016/07 (Cloud SQL 2nd Gen GA)		
Similarity scores							(2-gram	3-gram	**4-gram**)			
MySQL 5.6	-			.78	.68	**.64**	.81	.73	**.69**	.77	.67	**.64**
MySQL 5.6 (Google)	.78	.68	**.64**	-			.77	.68	**.64**	.79	.70	**.67**
MySQL 5.7	.81	.73	**.69**	.77	.68	**.64**	-			.78	.69	**.66**
MySQL 5.7 (Google)	.77	.67	**.64**	.79	.70	**.67**	.78	.69	**.66**	-		
MySQL 8.0	.81	.72	**.67**	.77	.67	**.63**	.81	.73	**.68**	.76	.67	**.63**
MySQL 8.0 (Google)	.75	.64	**.61**	.76	.66	**.62**	.75	.65	**.61**	.76	.65	**.62**
PostgreSQL 9.6	.47	.25	**.06**	.45	.23	**.06**	.47	.25	**.06**	.44	.23	**.06**
PostgreSQL 9.6 (Google)	.44	.22	**.05**	.42	.20	**.05**	.44	.22	**.05**	.41	.20	**.05**
PostgreSQL 10	.47	.24	**.06**	.45	.23	**.06**	.47	.24	**.06**	.44	.23	**.06**
PostgreSQL 10 (Google)	.44	.22	**.05**	.41	.20	**.05**	.43	.21	**.05**	.41	.20	**.05**
PostgreSQL 11	.47	.24	**.06**	.44	.23	**.06**	.46	.24	**.06**	.43	.23	**.06**
PostgreSQL 11 (Google)	.44	.22	**.05**	.41	.20	**.05**	.43	.21	**.05**	.41	.20	**.05**
PostgreSQL 12	.47	.24	**.06**	.45	.23	**.06**	.47	.24	**.06**	.44	.23	**.06**
PostgreSQL 12 (Google)	.44	.22	**.05**	.41	.20	**.05**	.43	.21	**.05**	.41	.20	**.05**
PostgreSQL 13	.47	.25	**.06**	.45	.23	**.06**	.47	.24	**.06**	.44	.23	**.06**
PostgreSQL 13 (Google)	.44	.22	**.05**	.41	.20	**.05**	.43	.22	**.05**	.41	.20	**.05**
PostgreSQL 14	.41	.19	**.05**	.39	.18	**.04**	.40	.19	**.05**	.38	.18	**.04**
PostgreSQL 14 (Google)	.38	.16	**.04**	.36	.15	**.03**	.38	.16	**.04**	.35	.15	**.03**
PostgreSQL 15	.40	.19	**.05**	.38	.18	**.04**	.40	.19	**.05**	.38	.18	**.04**
PostgreSQL 15 (Google)	.38	.17	**.04**	.35	.16	**.03**	.37	.17	**.04**	.35	.16	**.03**
PostgreSQL 16	.43	.19	**.05**	.40	.18	**.04**	.42	.20	**.05**	.40	.18	**.04**
PostgreSQL 16 (Google)	.44	.22	**.05**	.41	.20	**.05**	.43	.21	**.05**	.41	.20	**.05**
SQL Server 2017 dev	.41	.30	**.26**	.36	.29	**.26**	.40	.31	**.27**	.36	.28	**.26**
SQL Server 2017 enterprise (Google)	.42	.31	**.25**	.37	.29	**.25**	.41	.31	**.26**	.37	.29	**.25**
SQL Server 2017 standard (Google)	.42	.31	**.25**	.37	.29	**.25**	.41	.31	**.25**	.37	.29	**.25**
SQL Server 2019 dev	.41	.30	**.26**	.36	.29	**.26**	.40	.31	**.27**	.36	.28	**.26**
SQL Server 2019 enterprise (Google)	.39	.28	**.22**	.35	.26	**.22**	.38	.28	**.22**	.34	.26	**.22**
SQL Server 2019 standard (Google)	.42	.30	**.24**	.37	.29	**.24**	.41	.31	**.25**	.37	.28	**.24**
SQL Server 2022 dev	.41	.30	**.26**	.36	.29	**.26**	.40	.31	**.27**	.36	.28	**.26**
SQL Server 2022 enterprise (Google)	.41	.31	**.25**	.37	.29	**.25**	.41	.31	**.26**	.37	.29	**.25**
SQL Server 2022 standard (Google)	.39	.29	**.24**	.35	.27	**.24**	.38	.29	**.24**	.35	.27	**.24**

MySQL v. other engines

High similarity between different versions of the same database engine.

Low similarity when comparing different database engines.



4-gram similarity scores between various database versions

High similarity between different versions of the same database engine.

Low similarity when comparing different database engines.

Conclusions

It's possible to **distinguish between different relational database engines** (e.g. MySQL, PostgreSQL, and SQL Server) and even different versions or deployment environments of the same engine, solely by examining their network traffic at application layer.

Thank you!

Follow Project Martial development:

https://github.com/raresraf/project-martial