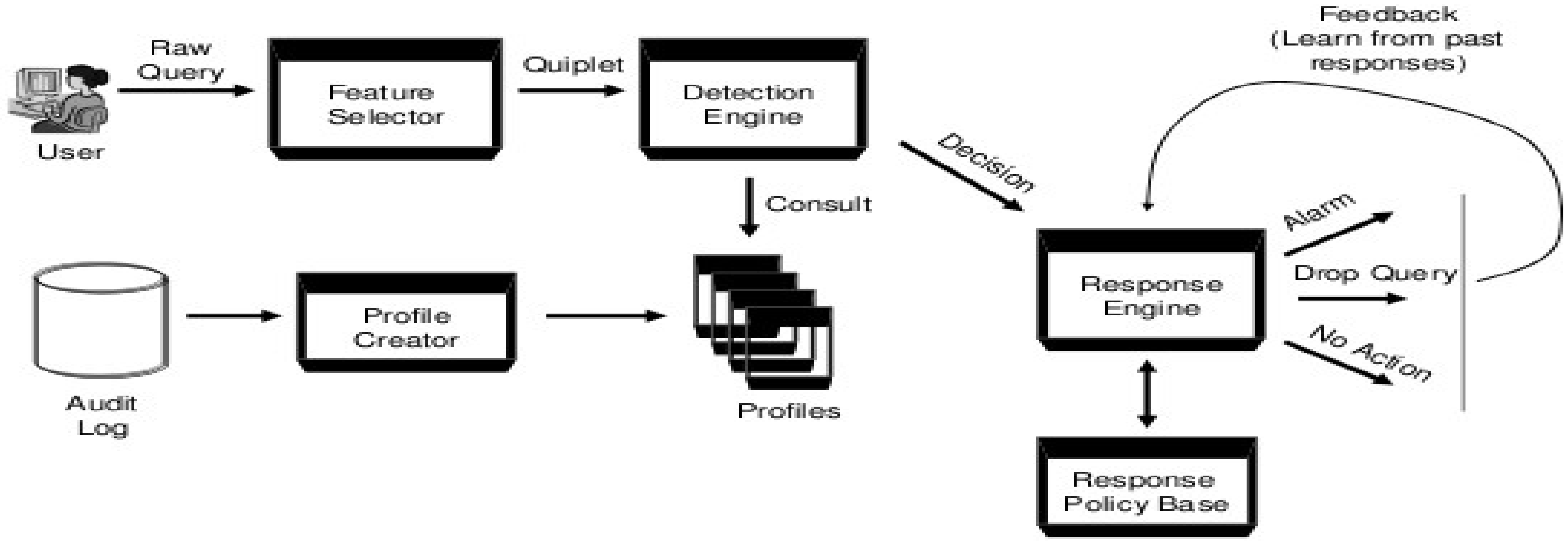
CERIAS

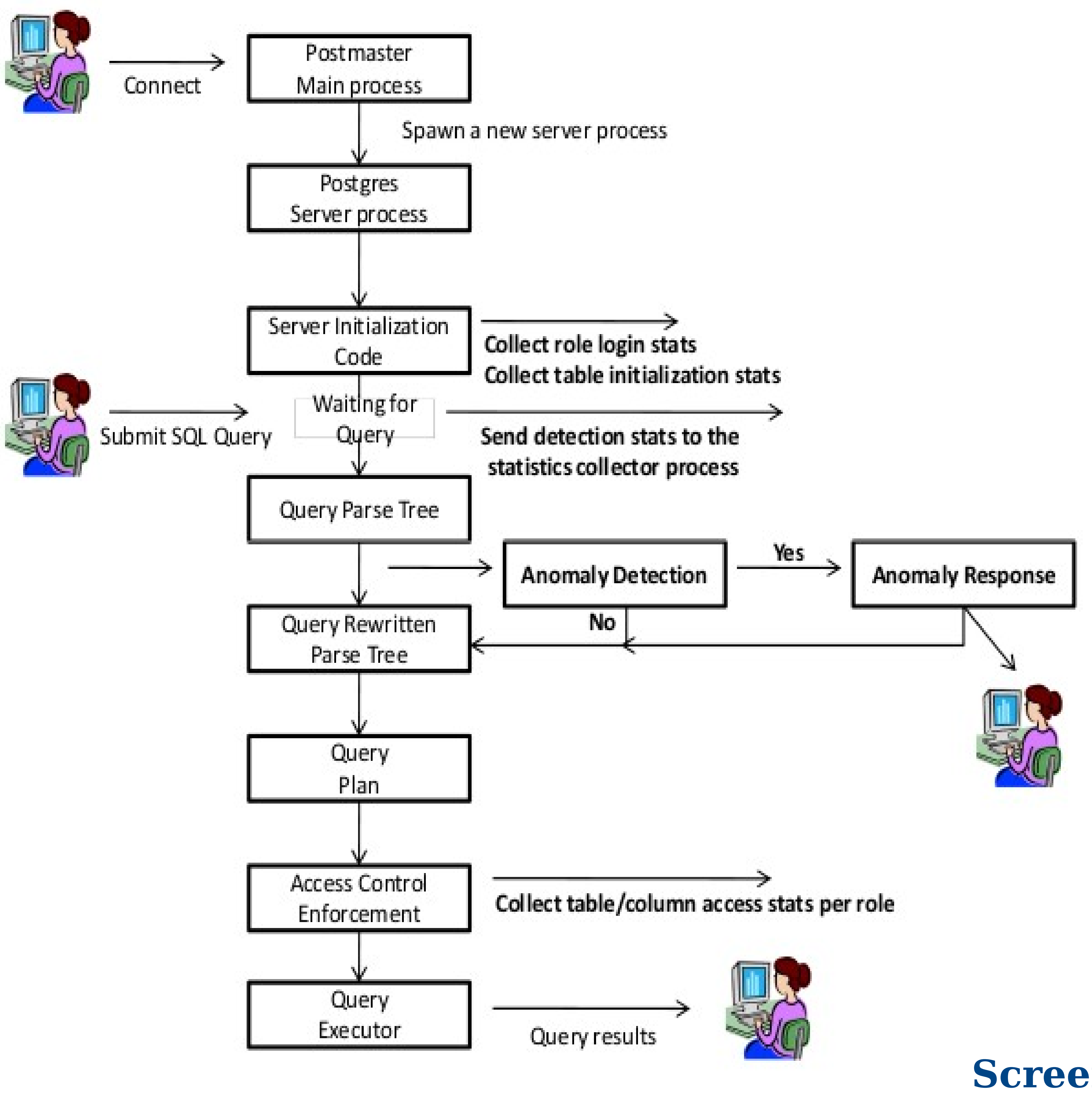
The Center for Education and Research in Information Assurance and Security

PostgreSQL - Anomalous Query Detector

Bilal Shebaro, Asmaa Sallam, Ashish Kamra, Elisa Bertino



Anomaly Detection and Data Collection Hooks in PostgreSQL


Connect
Postmaster Main process
Spawn a new server process
Postgres Server process
Server Initialization Code
Collect role login stats
Collect table initialization stats
Submit SQL Query
Waiting for Query
Send detection stats to the statistics collector process
Query Parse Tree
Anomaly Detection
Yes
Anomaly Response
No
Query Rewritten Parse Tree
Query Plan
Access Control Enforcement
Collect table/column access stats per role
Query Executor
Query results

PostgreSQL

- Open Source DBMS, Employs Role-Based Access Control
- Detect insider attacks

Phases of operation

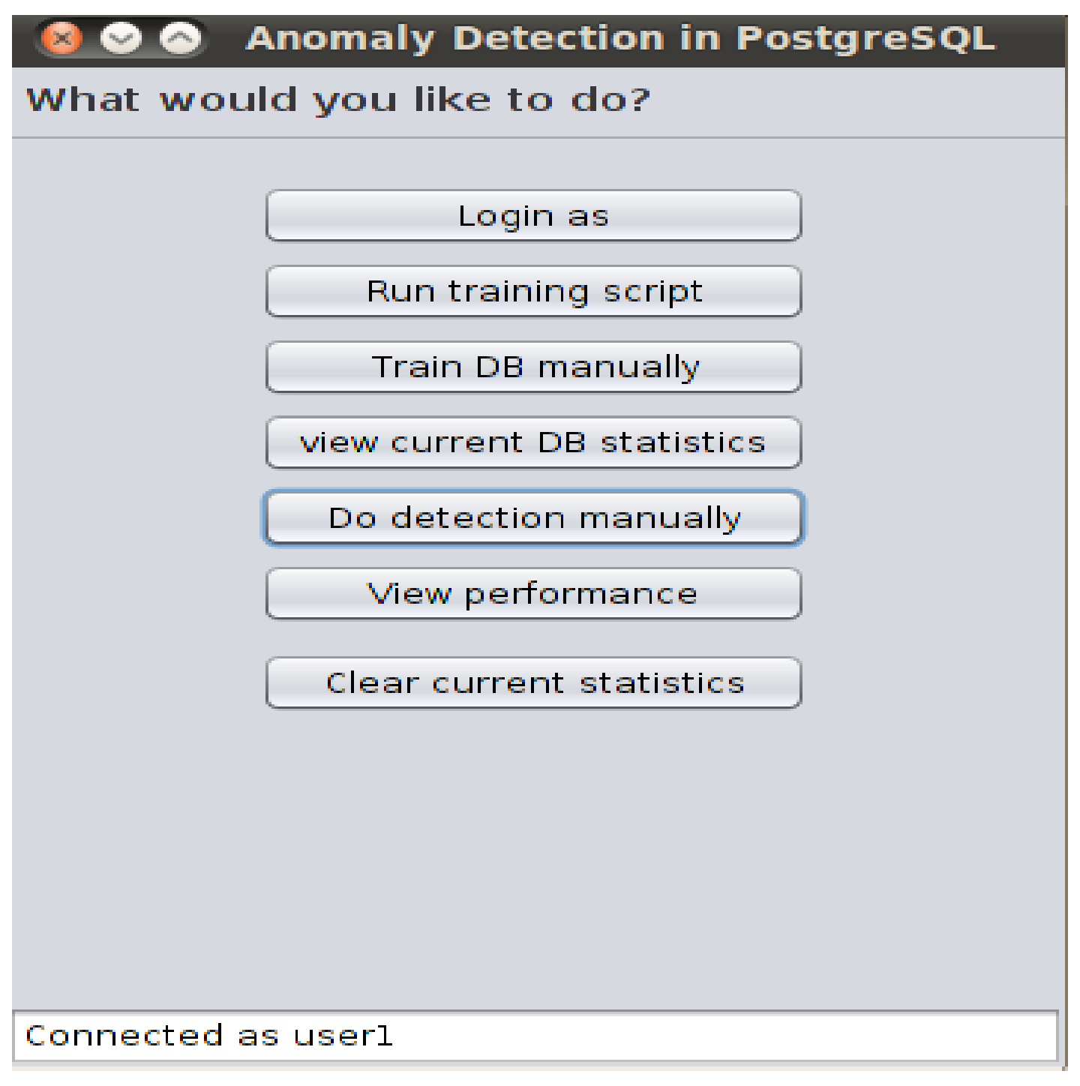
1- Training phase

- Create profiles for roles
- Use valid queries in the Audit Log
- Statistics collector process to report statistics during the DB operation
- Different ranges for information extraction: coarse, medium and fine quiplets

2- Detection phase

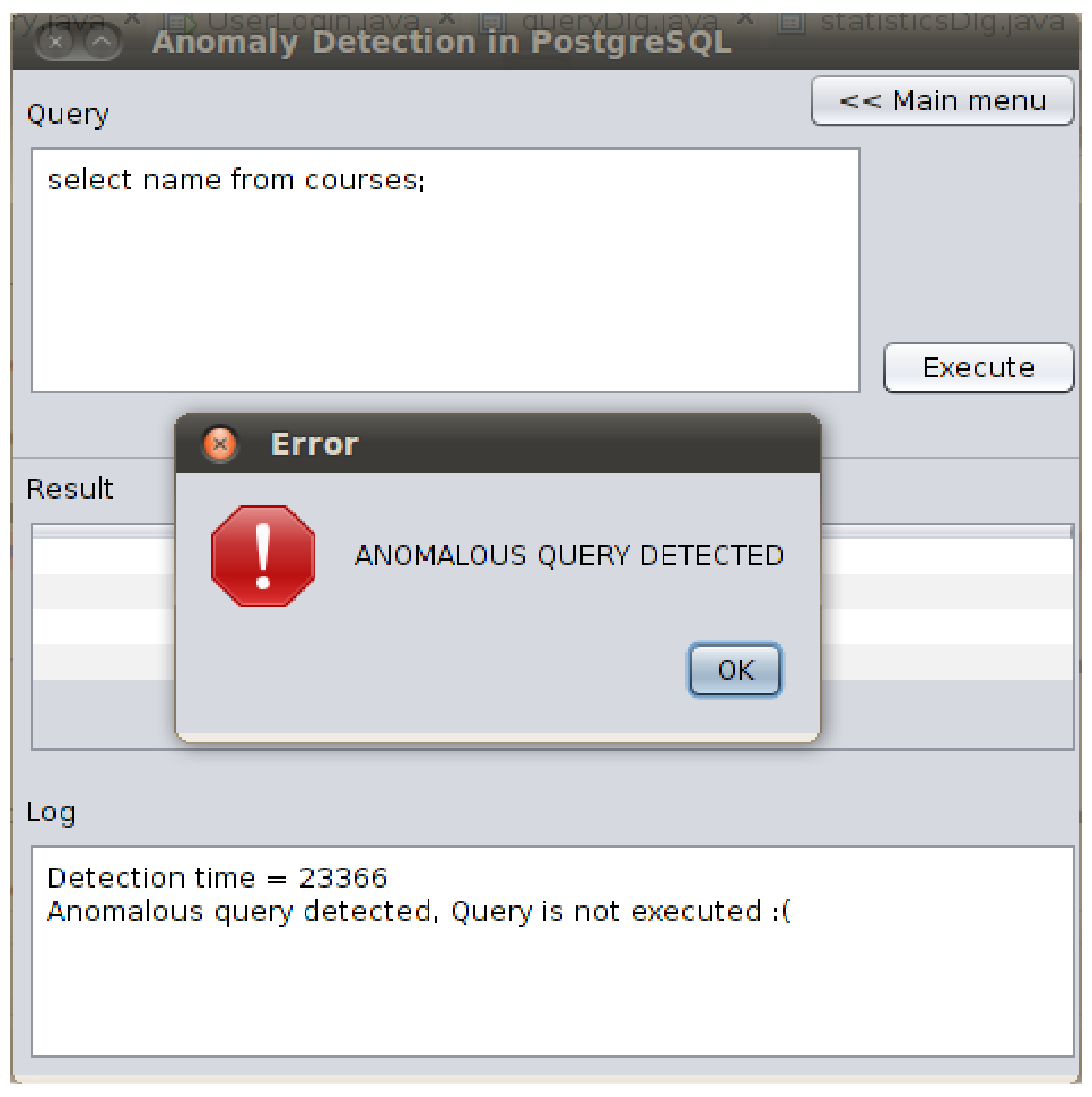
- Query should match profile of user's role
- Use naïve Bayes classifier
- Max. Aposteriori role for classification
- Response (currently): Drop Query

Screen-shots



What would you like to do?

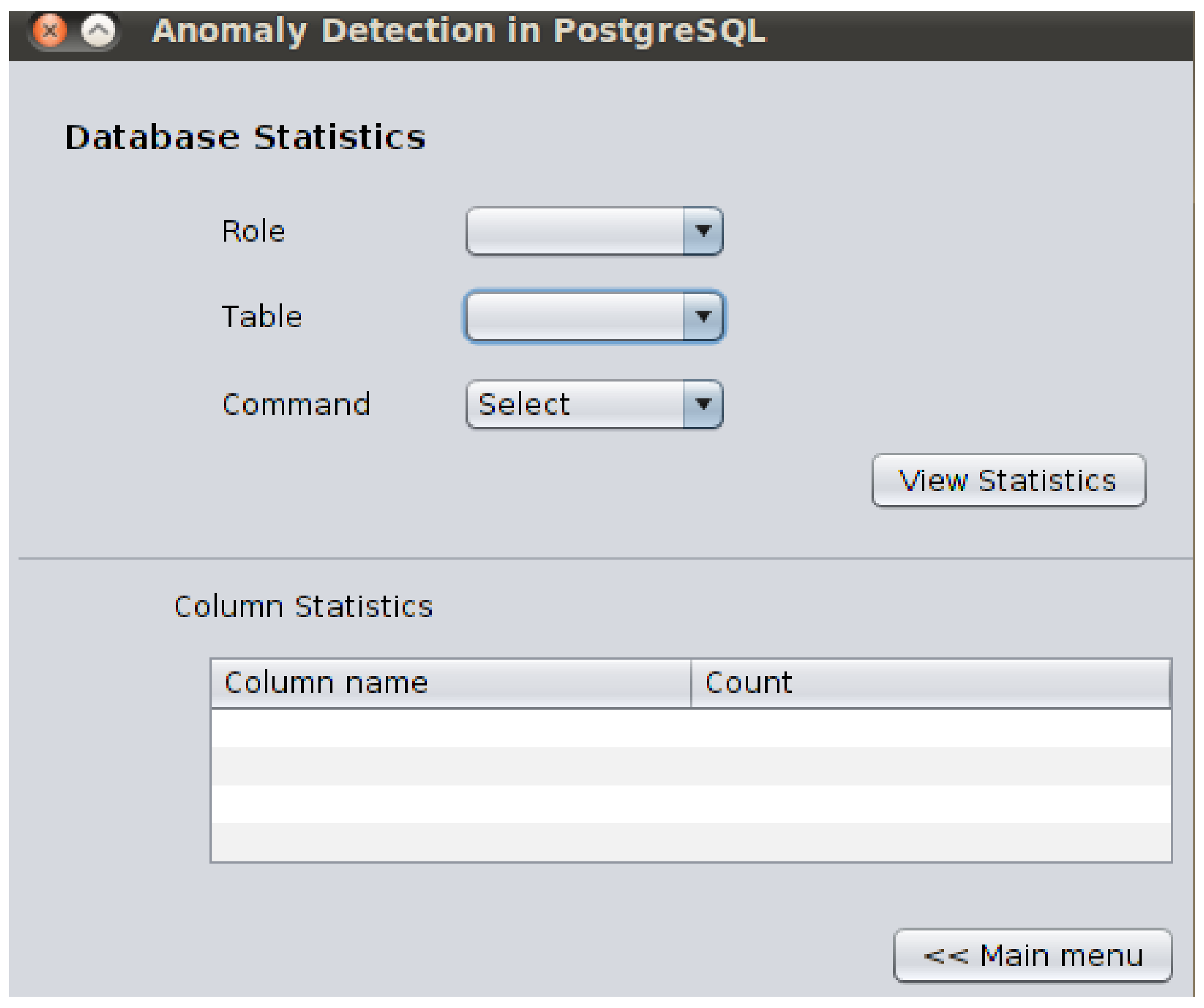
Connected as user1



select name from courses;

ANOMALOUS QUERY DETECTED

```
Detection time = 23366
Anomalous query detected, Query is not executed :(
```



Database Statistics

Role

Table

Command: Select

Column Statistics

Column name	Count

CERIAS

PURDUE UNIVERSITY