Your own database security Inspector with alerting mechanism

About us..

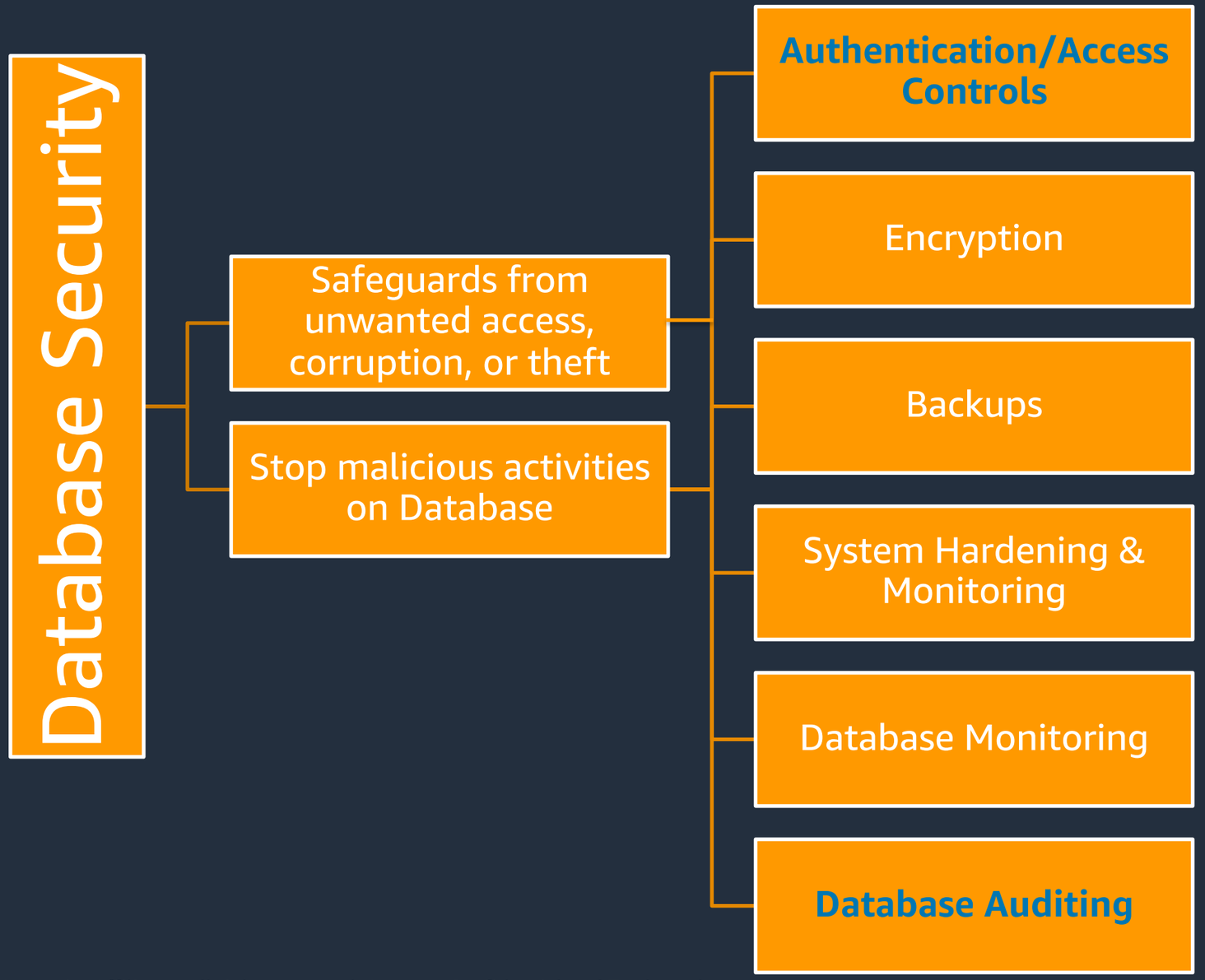
- Senior Consultant at AWS Professional Services
- Database Migration Specialist
- Avid PostgreSQL follower/user
- Email: bkhari@amazon.com

- Consultant at AWS Professional Services
- Database Migration Specialist
- Avid PostgreSQL follower/user
- Email: madiwale@amazon.com

Agenda

- Importance of Database Security & Types of Security
- Use case
- Architecture
- Limitations
- Demo
- Q & A

Importance of Database security & Types of Security



Use Case

Requirement:

One of our customers was looking for a custom solution to audit like "Grant" and "Revoke" performed on a database and alert in near realtime to the concerned authorities

Solution :

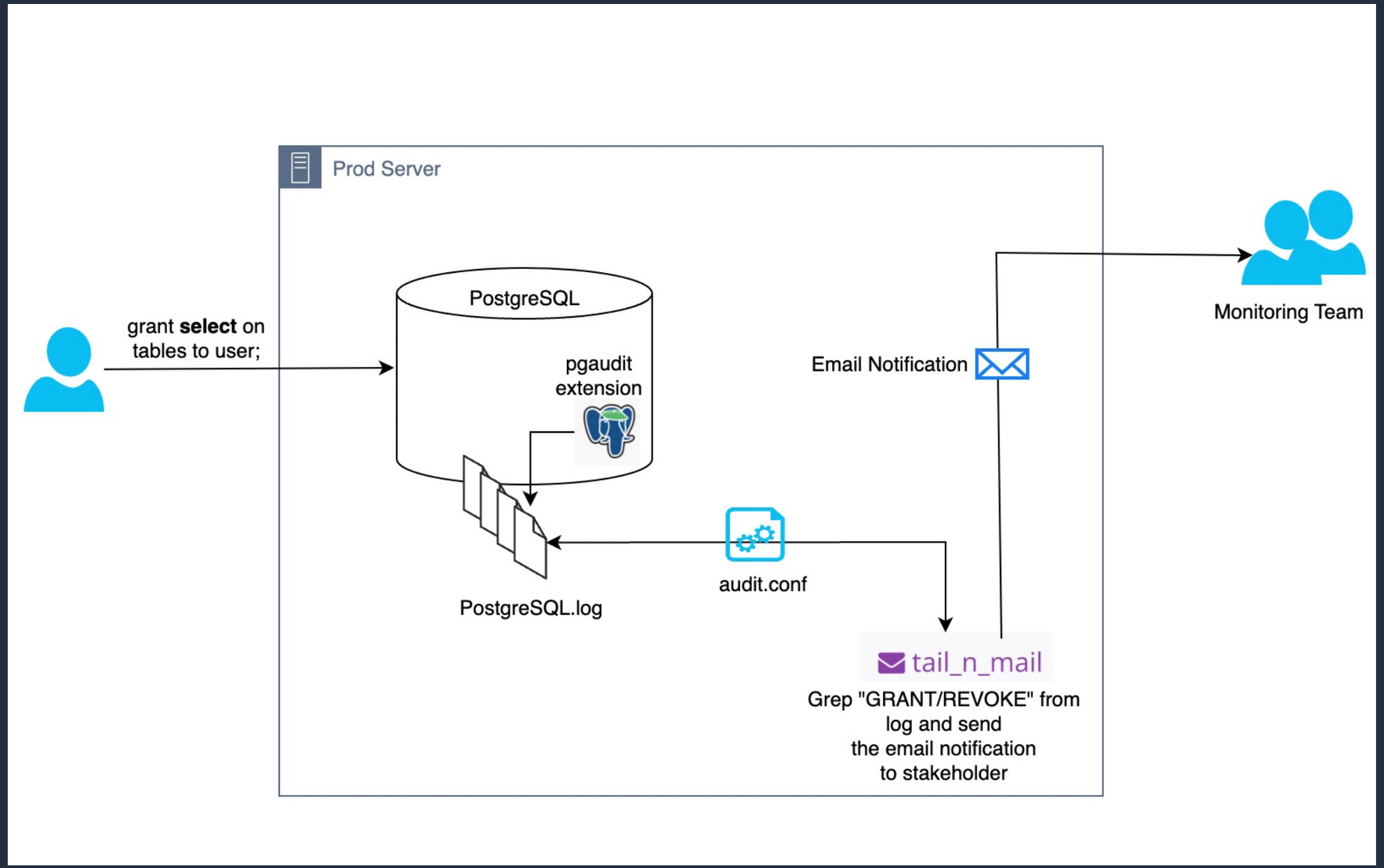
pgaudit + tail_n_mail + crontab

Features :

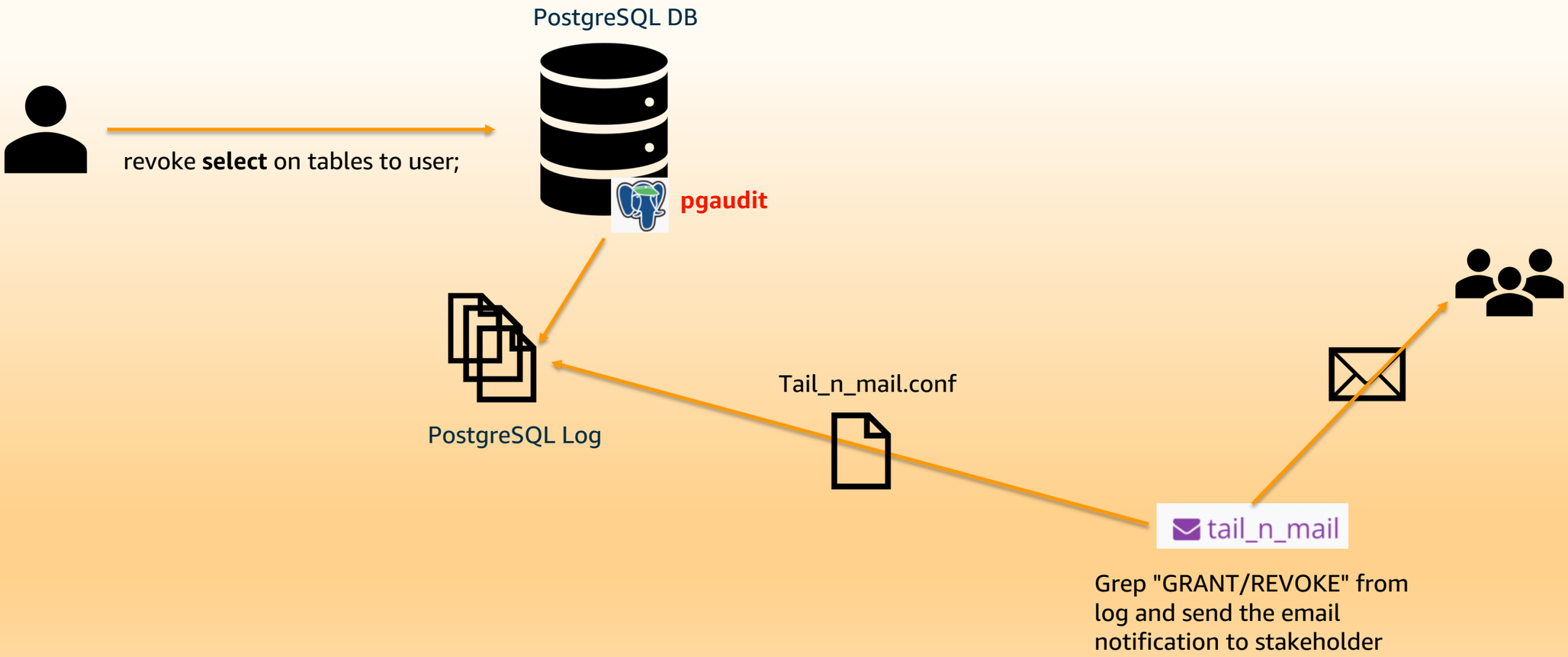
- **Fine grained auditing**
- **Near realtime alerting via email**
- **Light weight use of open source extension and utility**

aws

Solution Architecture



Solution Architecture



pgaudit Options and Parameters

pgaudit

- It’s an extension
- Audits -Instance ,DB, User level
- Available options
  - ALL
  - **ROLE**
  - MISC_SET
  - WRITE
  - **MISC**
  - FUNCTION
  - READ
  - DDL

pgaudit parameters

- **pgaudit.log**
- pgaudit.log_catalog
- pgaudit.log_client
- pgaudit.log_level
- **pgaudit.log_parameter**
- **pgaudit.log_relation**
- pgaudit.log_rows
- pgaudit.log_statement_once
- pgaudit.log_statement
- pgaudit.role

Bucardo tail_n_mail

- Perl script
- Travels to the point in the file
- Parses database logs based on keywords
- Sends email message

High level Steps for setup

Pre-Requisites

- **Perl packages**
- **Pgaudit**
- **tail_n_mail**
- **SMTP**

Create

- **pgaudit extension**
- **Config for tail_n_mail**

Schedule Job

- **Crontab with frequency as per the requirement of every 5 or 1 Min**

Interpret the audit log entry

2023-02-19 09:45:49 UTC [9750] writer@madiwale LOG: AUDIT: SESSION,3,1,ROLE,REVOKE,TABLE,,REVOKE select on dept from reader ;,<not logged>

Field	Description	Value from example
AUDIT_TYPE	Indicates the audit mode: SESSION or OBJECT	SESSION
STATEMENT_ID	Unique statement identifier for each session	3
SUBSTATEMENT_ID	An identifier for each sub statement within the main statement	1
CLASS	Indicates the class of statements like READ, WRITE etc that are defined values for pgaudit.log parameter.	ROLE
COMMAND	The command used in the SQL statement	REVOKE
OBJECT_TYPE	Can be TABLE, INDEX, VIEW, etc.	TABLE
OBJECT_NAME	The fully qualified object name	
STATEMENT	The actual statement executed	REVOKE select on dept from reader;
PARAMETER	When the pgaudit.log_parameter is set to true, the quoted CSV of parameters is listed if present, or "none" if there are no parameters. When the pgaudit.log_parameter is not set, the value will be "<not logged>"	<not logged>

pgAudit Limitations

- Autovacuum and Autoanalyze are not logged
- Object renames are logged under the name they were renamed to
- It is not possible to reliably audit superusers with pgAudit
- Statements that are executed after a transaction enters an aborted state will not be audit logged.

aws

Demo

Q & A