

**AUDIT LOG FOR MONITORING USER ACTIVITIES AND
AUTHORIZATION FEATURE FOR SECURITY AND VERIFICATION**

PREPARED BY
Deeksha Gupta (2018BTechCSE126)

FACULTY GUIDE
Dr Taruna Sunil



**Department of Computer Science and Engineering
Institute of Engineering and Technology (IET)
JK Lakshmipat University Jaipur**

May 2022

**AUDIT LOG FOR MONITORING USER ACTIVITIES AND
AUTHORIZATION FEATURE FOR SECURITY AND VERIFICATION**

Practice School-2

Submitted in partial fulfilment of the requirements

for the degree of

Bachelor of Technology in Computer Science and Engineering

By:

Deeksha Gupta

Faculty Guide:

Dr Taruna Sunil



Department of Computer Science and Engineering

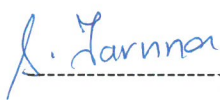
Institute of Engineering and Technology (IET)

JK Lakshmipat University Jaipur

May 2022

CERTIFICATE

This is to certify that the Practice School-2 project work aimed at “Providing Solutions for Telecom Operators and Agencies” submitted by **Deeksha Gupta (2018BTechCSE126)**, towards the partial fulfillment of the requirements for the degree of **Bachelor of Technology in Computer Science and Engineering** of JK Lakshmipat University Jaipur is the record of work carried out by them under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for Practice School-II examination.



Dr Taruna Sunil

Associate Professor,

Department of Computer Science and
Engineering

Institute of Engineering & Technology (IET)
JK Lakshmipat University Jaipur



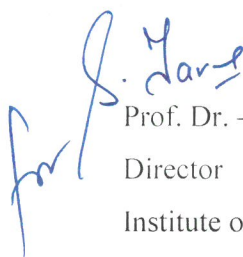
Dr Sonal Jain

Professor,

Department of Computer Science and
Engineering

Institute of Engineering & Technology (IET)
JK Lakshmipat University Jaipur

Date of Submission 13-05-2022



Prof. Dr. – Sanjay Goel

Director

Institute of Engineering & Technology
JK Lakshmipat University Jaipur

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to **Prof Sanjay Goel**, Director, Institute of Engineering & Technology, JK Lakshmipat University Jaipur for providing me with such a great opportunity. I would also like to thank my faculty guide **Dr Taruna Sunil** for constant support and motivation.

The internship opportunity in **Sandvine Technologies** has given me a great chance to grow and gain experience in professional world. I am fortunate enough for this amazing opportunity. I am very thankful and grateful for working with great people and professionals in Sandvine Technologies.

I am deeply grateful to Sandvine Technologies for their guidance and for providing all the necessary information for project which helped me in completing my project.

I would also express my gratitude towards my Manager and Mentor **Mr Varun Kumbheshwar** for constant support, motivation and guidance for completion of project.



Sincerely yours,

Deeksha Gupta

(2018BTechCSE126)

ABSTRACT

Sandvine helps customers deliver exceptional personalized services and improve network performance by combining real-time contextual insights, customer quality of experience, and network visibility in convenient, pre-packaged use cases. New 5G, Cloud, Edge, and Artificial Intelligence technologies are transforming networks, creating new market opportunities and challenges for service providers, governments and enterprises. Sandvine's Application and Network Intelligence Portfolio is helping these industries deliver exceptional application experiences, while managing the associated surges and shifts in network traffic.

An audit log is essentially a record of events, user activities and system changes. Audit logs are records of sequence of user activities and they capture who, what, where, and when of an activity and whether the activity was successful or failure, and if it is failure why so.

Authorization is the function of specifying access rights/privileges to resources, which is related to general information security and computer security, and to access control in particular.

LIST OF FIGURES

Figure 2.1.2: POST	6
Figure 2.1.2: GET	7
Figure 2.1.3: Blocking Queue	7
Figure 2.2.1: Horizontal Scaling DB Query Service	9
Figure 2.2.3: Interaction Oauth2	13
Figure 3.1.1: Multithreading	15
Figure 3.2.3: Docker Architecture	21
Figure 3.3: Rest API Model	23
Figure 3.4.1: Vertica Projection	25

LIST OF TABLES

Table 2.1 Audit Log Fields	5
Table 2.1.3: Blocking Queue Operations	8

LIST OF FLOWCHARTS

Flowchart 1 Audit Log Service Block Diagram	4
---	---

CONTENTS

CERTIFICATE	III
ACKNOWLEDGEMENTS	IV
ABSTRACT	V
LIST OF FIGURES	VI
LIST OF TABLES	VI
LIST OF FLOWCHARTS	VI
1. INTRODUCTION	1
1.1 REGULATORY R&D	3
1.2 PROBLEM STATEMENT	3
2. ABOUT THE FEATURE	4
2.1 AUDIT LOG	4
2.1.1 <i>Audit Log Fields</i>	5
2.1.2 <i>Rest API Endpoints</i>	6
2.1.3 <i>Blocking Queue</i>	7
2.2 DB QUERY SERVICE	8
2.2.1 <i>Features</i>	8
2.2.2 <i>Authorization Feature</i>	9
2.2.3 <i>OAuth2</i>	11
3. TOOLS AND TECHNOLOGIES	14
3.1 LANGUAGES	14
3.1.1 <i>Java</i>	14
3.1.2 <i>Golang</i>	16
3.2 FRAMEWORKS	18
3.2.1 <i>Springboot</i>	18
3.2.2 <i>Keycloak</i>	20
3.2.3 <i>Docker</i>	20
3.3 REST API	22
3.4 DATABASE	24
3.4.1 <i>Vertica</i>	24

3.5	COLLABORATION PLATFORM	26
3.6	TOOLS	26
3.6.1	<i>Mobaxterm</i>	26
3.6.2	<i>Eclipse</i>	26
3.6.3	<i>Postman</i>	26
3.6.4	<i>Putty</i>	26
4.	CONCLUSION	27
	REFERENCES	28

CHAPTER 1: INTRODUCTION



Sandvine helps customers deliver exceptional personalized services and improve network performance by combining real-time contextual insights, customer quality of experience, and network visibility in convenient, pre-packaged use cases.

Actionable Intelligence

Active and passive intelligence enables immediate actions. These include: prioritizing and monitoring application and network traffic, ensuring quality of experience, managing congestion, monitoring cyber threats, and fraud management.

High-Quality Experiences

Sandvine provides over 95% traffic visibility by user, application, network, device, demographic, location, and other parameters – the best in the industry. This helps our customers analyze, optimize, and monetize applications to deliver exceptional consumer and enterprise experiences.

Sandvine classifies application traffic across mobile and local networks by user, device, network type, location and other parameters. The company then applies machine learning-based analytics to real-time data and makes policy changes to optimize, secure, and monetize applications.

New 5G, Cloud, Edge, and Artificial Intelligence technologies are transforming networks, creating new market opportunities and challenges for service providers, governments and enterprises.

Sandvine's Application and Network Intelligence Portfolio is helping these industries deliver exceptional application experiences, while managing the associated surges and shifts in network traffic.

Sandvine's ActiveLogic equips operators with a simplified, hyperscale, automation-ready data plane to tackle the next era of network challenges. With ActiveLogic, operators have the flexibility, performance, and machine learning-powered traffic classification needed to keep up with internet phenomena.

Sandvine's ActiveLogic, a hyperscale data plane and policy enforcer, is a key network element for operators to optimize network resources. In order to manage the ever-increasing band-width demands, operators need to rethink the "dumb data pipes" approach, and adopt an application-centric view of the network. By taking this approach, operators can bring surgical-level optimization, translating into better quality of experience and resource allocation, versus using strict policies for policing or blocking traffic.

Sandvine's TCP Accelerator allows network operators to dramatically improve network efficiency and quality of experience (QoE), without making changes to the network infrastructure.

TCP Accelerator addresses the main issues associated with TCP, including the incompatibility of bandwidth availability and TCP speeds. It also addresses the inefficiencies caused by multiple concurrent TCP connections by transparently bridging the access network with the transit network; it is in a position to manage all the network's TCP connections as a collective whole.

1.1 REGULATORY R&D

Government agencies are responsible for keeping citizens and societies safe and secure, and this includes connectivity and internet services. As the world becomes more connected and the reliance on internet services increases, the work for government agencies becomes more complex. Sandvine can help telecom agencies enforce their telecom policies and ensure the quality of broadband while protecting citizens against threats.

Sandvine delivers a unique independent view of national broadband quality, enabling them to enhance their worldwide competitiveness without sacrificing the safety and security of their citizens.

Sandvine's Government solutions are developed for a modern, connected society to serve and protect citizens, secure infrastructure, and ensure a fair and competitive market environment. Sandvine's unique ability to scale to nationwide deployments for hundreds of millions of

broadband connections while delivering multiple solutions positions Sandvine as a strategic partner to telecom agencies worldwide.

1.2 PROBLEM STATEMENT

- Audit logs help with security because they provide records of all IT activity, including suspicious activity. Audit logs can assist with monitoring data and systems for any possible security breaches or vulnerabilities.
- Authorization is a security mechanism to determine access levels or user/client privileges related to system resources including files, services, computer programs, data and application features. This is the process of granting or denying access to a network resource which allows the user access to various resources based on the user's identity.

OBJECTIVE 1: To create an Audit Log for User Activities

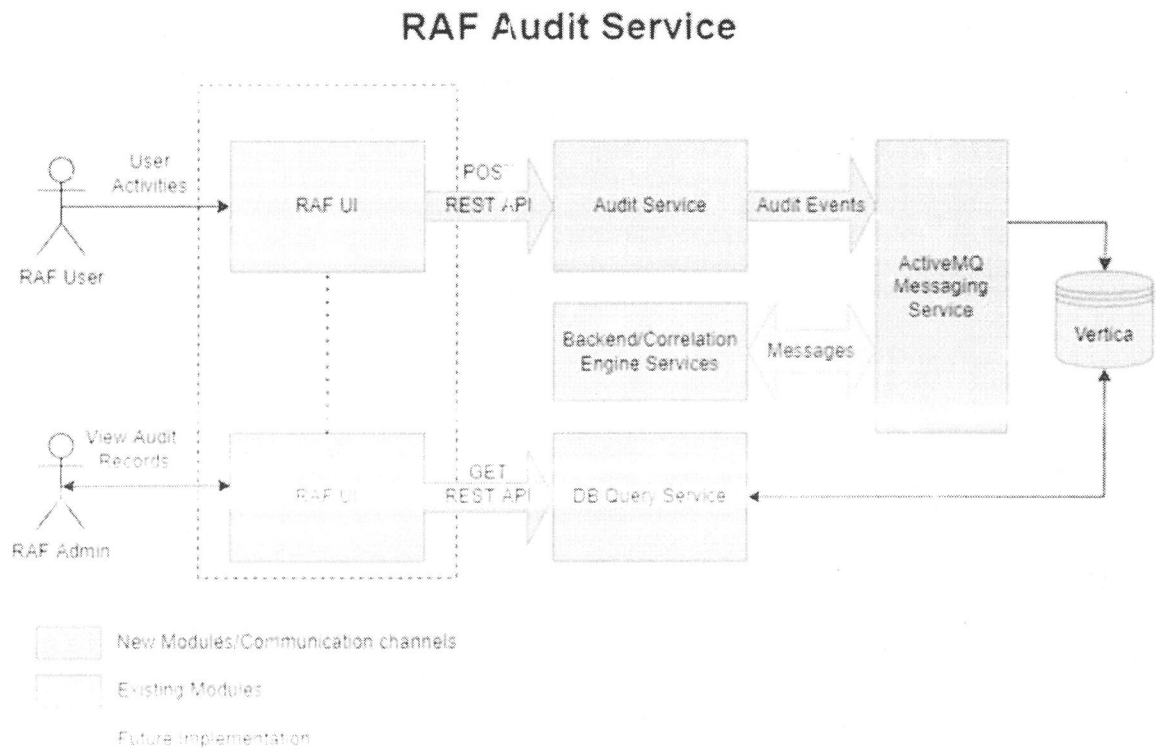
OBJECTIVE 2: To create an Authentication/Authorization Feature.

CHAPTER 2 : ABOUT THE FEATURE

2.1 AUDIT LOG

An audit log is essentially a record of events, user activities and system changes. Audit logs are records of sequence of user activities and they capture who, what, where, and when of an activity and whether the activity was successful or failure, and if it is failure why so.

It is a matter of sensitive data. Activities of users using system must be logged automatically and available for review by Admin user.



Flowchart 1: Audit Log Service Block Diagram

2.1.1 AUDIT LOG FIELDS

The information included in the log should provide the context of the event, the “who, what, where, and when,” and anything else of relevance.

Field	Description	Is Mandatory?
User Name		NO
User ID		YES
Agency		YES
Event Time		YES
Operation		YES
Operation Type		YES
Client ID		YES
Origin		YES
Log Message		NO
EntityValue		YES
EntityType		YES
Metadata		NO

Table 2.1 Audit Log Fields

2.1.2 REST API ENDPOINTS

Endpoint	HTTP Method	Description
/raf/v1/audit/events	POST	POST HTTP method to send the audit events. Consumes AuditRecord in a JSON format. Validates the AuditRecord If its invalid response with "BAD request" response Else pushes audit event message to Durable Topic Once pushed response with 200 OK

Figure 2.1.2: POST

/raf/v1/audit/events	GET	GET HTTP method to retrieve list of audit events based on the filter parameters passed. • API supports paginated response • API supports below query params to filter the Audit Records: • By StartTime, EndTime • By UserName • By Agency • By Operation • By OperationType
----------------------	-----	---

Figure 2.1.2: GET

2.1.3 BLOCKING QUEUE

A blocking queue is a queue that blocks when you try to dequeue from it and the queue is empty, or if you try to enqueue items to it and the queue is already full. A thread trying to dequeue from an empty queue is blocked until some other thread inserts an item into the queue. A thread trying to enqueue an item in a full queue is blocked until some other thread makes space in the queue, either by dequeuing one or more items or clearing the queue completely.

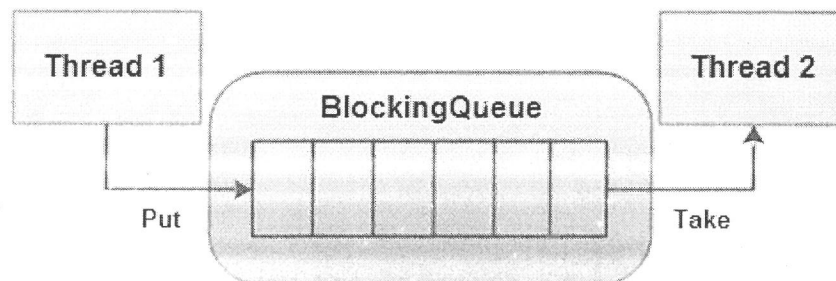


FIGURE 2.1.3: Blocking Queue

A BlockingQueue with one thread putting into it, and another thread taking from it.

	Throws Exception	Special Value	Blocks	Times Out
Insert	<code>add(o)</code>	<code>offer(o)</code>	<code>put(o)</code>	<code>offer(o, timeout, timeunit)</code>
Remove	<code>remove(o)</code>	<code>poll()</code>	<code>take()</code>	<code>poll(timeout, timeunit)</code>
Examine	<code>element()</code>	<code>peek()</code>		

TABLE 2.1.3: Blocking Queue Operations

2.2 DB QUERY SERVICE

Db query service is used to convert queries into Rest Api.

2.2.1 FEATURES

- Provide APIs to extract stats and other data from Insights storage (i.e., Vertica)
- Few powerful APIs are sufficiently expressive as to require minimal config to extract data from Vertica tables
- Single-use queries ideally be provided by a **HTTP RESTful API**
 - Favouring GET as it can be cached
 - A superset API via POST can provide more complex queries (e.g., provide more complex filtering on WHERE clause than simple equality)
- Real-time continuously updating APIs (based on timeseries data) would ideally be provided using a **websockets based API**
- This service allows scaling horizontally by running multiple instances concurrently
- The APIs does not allow modifications of data or the database, but rather only enable querying of data

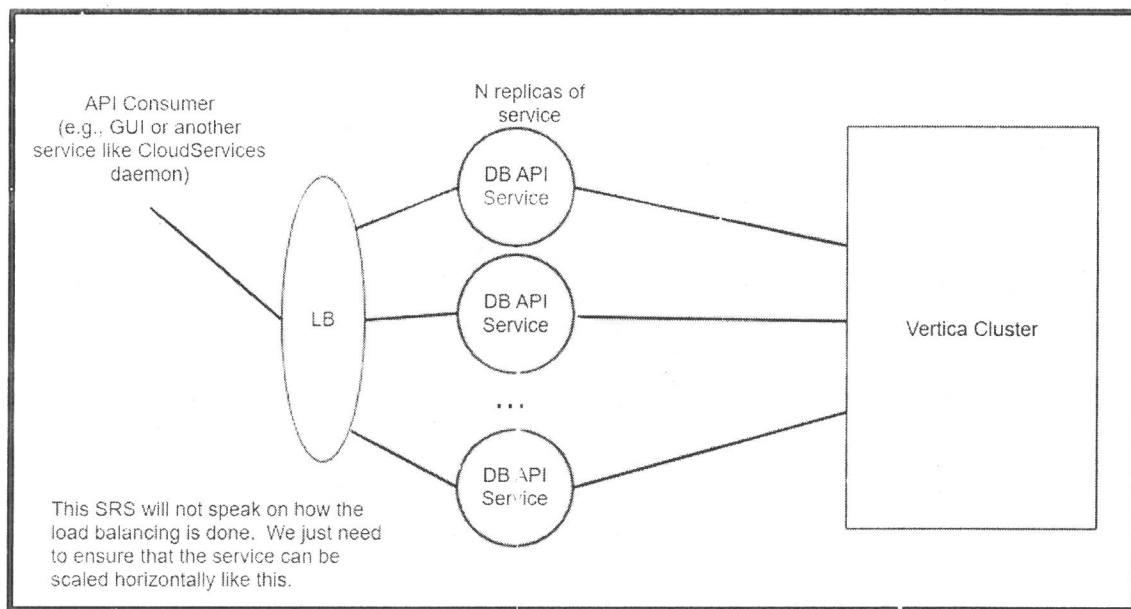
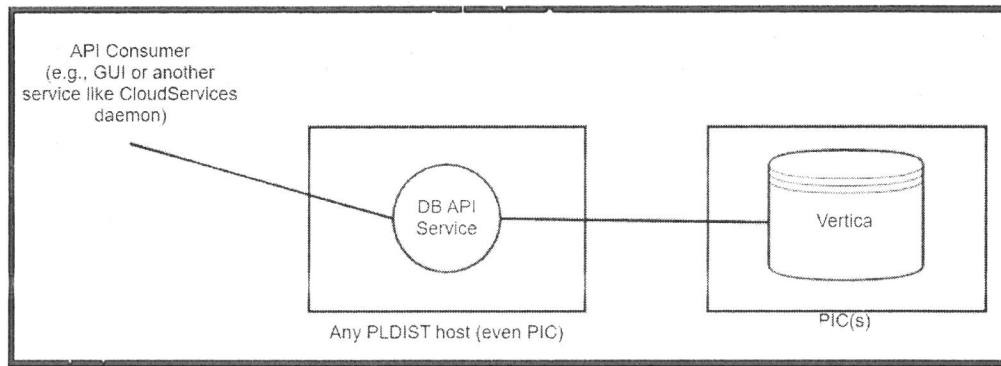


FIGURE 2.2.1: Horizontal Scaling DB Query Service

2.2.2 AUTHORIZATION FEATURE

Token-based authentication is a protocol that generates encrypted security tokens. It enables users to verify their identity to websites, which then generates a unique encrypted authentication token. That token provides users with access to protected pages and resources for a limited period of time without having to re-enter their username and password.

Token-based authentication works through this five-step process:

- Request: The user logs in to a service using their login credentials, which issues an access request to a server or protected resource.

- **Verification:** The server verifies the login information to determine that the user should have access. This involves checking the password entered against the username provided.
- **Token submission:** The server generates a secure, signed authentication token for the user for a specific period of time.
- **Storage:** The token is transmitted back to the user's browser, which stores it for access to future website visits. When the user moves on to access a new website, the authentication token is decoded and verified. If there is a match, the user will be allowed to proceed.
- **Expiration:** The token will remain active until the user logs out or closes the server.

An authentication token securely transmits information about user identities between applications and websites. They enable organizations to strengthen their authentication processes for such services.

An authentication token allows internet users to access applications, services, websites, and application programming interfaces (APIs) without having to enter their login credentials each time they visit. Instead, the user logs in once, and a unique token is generated and shared with connected applications or websites to verify their identity.

These tokens are the digital version of a stamped ticket to an event. The user or bearer of the token is provided with an access token to a website until they log out or close the service.

An authentication token is formed of three key components: the header, payload, and signature.

Header

The header defines the token type being used, as well as the signing algorithm involved.

Payload

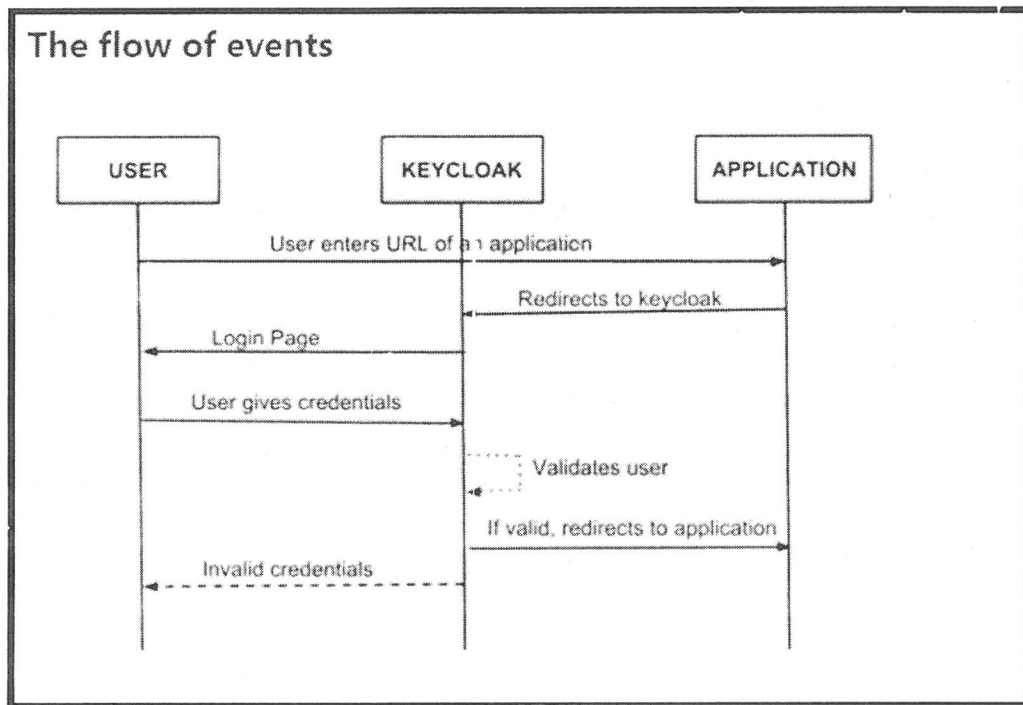
The payload is responsible for defining the token issuer and the token's expiration details. It also provides information about the user plus other metadata.

Signature

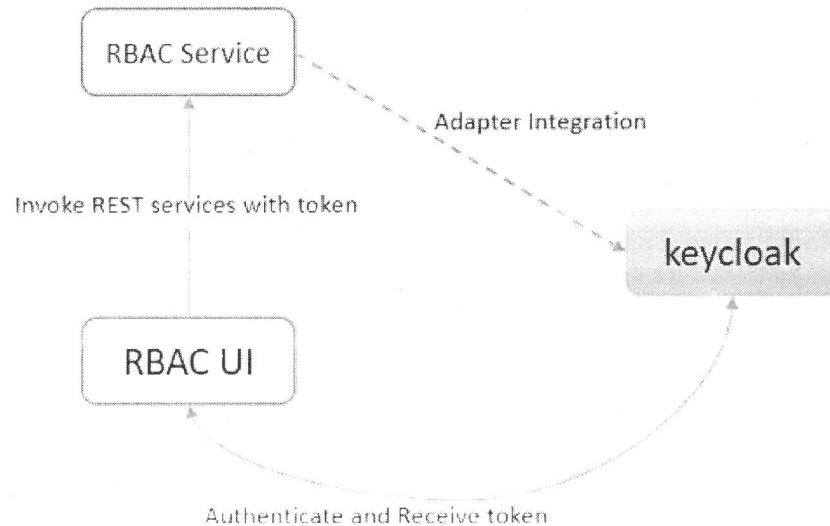
The signature verifies the authenticity of a message and that a message has not changed while in transit.

2.2.3 OAUTH 2

The authentication and authorization is controlled by Keycloak server for the RBAC Server and UI



Authorization Flow



Key responsibilities

- 1) User Management
 - a. User creation, updating user details like "Change Password", "Forgot Password", etc
- 2) Role Management
 - a. Assigning roles to user. There are 2 roles, Operator and Administrator
 - i. Administrator can Create/Modify warrant, execute reports
 - ii. Operator can view warrant information, execute reports
- 3) Domain Management (Multi-tenancy)
 - a. All the users and roles are grouped into a domain which is known as a tenant. This helps in isolating one LEA agency from another.
 - b. All the users and roles are under a LEA Agency. There can be multiple LEA Agency created which are maintained by a "Super User"
- 4) Authorization Services
 - a. Keycloak is responsible for authorization of applications as well. The RBAC UI and RBAC Server authorize themselves via Keycloak.

- b. The UI and Server obtains token from keycloak which are used to access the resources via REST

An authorization grant is where an OAuth2 client is given access to a protected resource using the OAuth2 protocol.

There are many kinds of grants including:

- Authorization Code
- Implicit
- Password
- Client Credentials
- Device Code
- Refresh Token

Amongst these, the safest and most used are the password, the authorization code, and the refresh token grants.

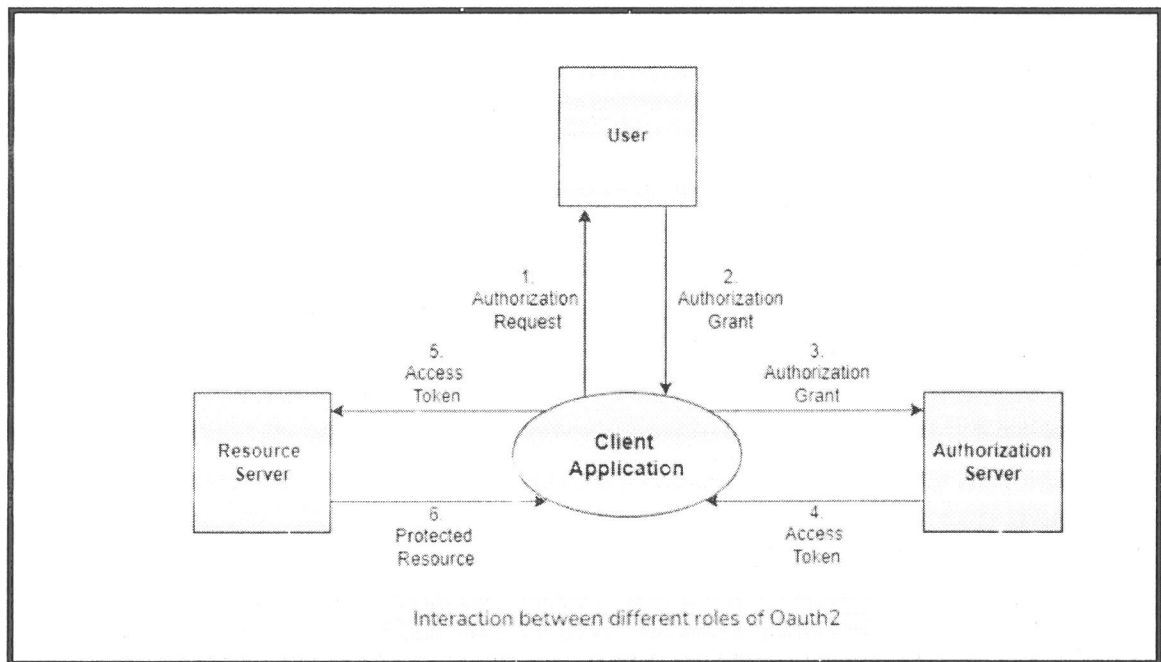


FIGURE 2.2.3: Interaction Oauth2

CHAPTER 3 : TOOLS AND TECHNOLOGY USED

3.1 LANGUAGES



3.1.1 JAVA

Java is a powerful general-purpose programming language. It is used to develop desktop and mobile applications, big data processing, embedded systems, and so on. According to Oracle, the company that owns Java, Java runs on 3 billion devices worldwide, which makes Java one of the most popular programming languages.

- Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.)
- It is one of the most popular programming language in the world
- It is easy to learn and simple to use
- It is open-source and free
- It is secure, fast and powerful
- It has a huge community support (tens of millions of developers)
- Java is an object oriented language which gives a clear structure to programs and allows code to be reused, lowering development cost

It is used for:

- Mobile applications (specially Android apps)
- Desktop applications
- Web applications
- Web servers and application servers
- Games
- Database connection
- And much more.

MULTI THREADING:

Multithreading is a process of executing multiple threads simultaneously.

A thread is a lightweight sub-process, the smallest unit of processing. Multiprocessing and multithreading, both are used to achieve multitasking.

However, we use multithreading than multiprocessing because threads use a shared memory area. They don't allocate separate memory area so saves memory, and context-switching between the threads takes less time than process.

Advantages of Java Multithreading

- 1) It **doesn't block the user** because threads are independent and you can perform multiple operations at the same time.
- 2) You **can perform many operations together, so it saves time.**
- 3) Threads are **independent**, so it doesn't affect other threads if an exception occurs in a single thread.

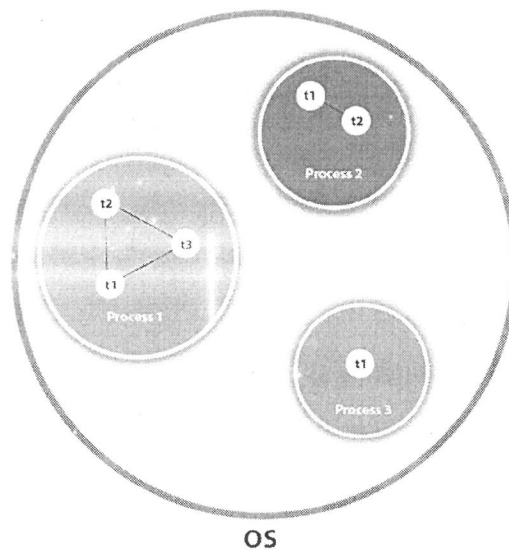


FIGURE 3.1.1: Multithreading

3.1.2 GOLANG



Golang

Go, also known as Golang, is an open-source, compiled, and statically typed programming language designed by Google. It is built to be simple, high-performing, readable, and efficient. Go is a procedural programming language. It was developed in 2007 by Robert Griesemer, Rob Pike, and Ken Thompson at Google but launched in 2009 as an open-source programming language. Programs are assembled by using packages, for efficient management of dependencies. This language also supports environment adopting patterns alike to dynamic languages. For eg., type inference (`y := 0` is a valid declaration of a variable `y` of type `float`).

ADVANTAGES

1. Flexible- It is concise, simple and easy to read.
2. Concurrency- It allows multiple process running simultaneously and effectively.
3. Quick Outcome- Its compilation time is very fast.
4. Library- It provides a rich standard library.
5. Garbage collection- It is a key feature of go. Go excels in giving a lot of control over memory allocation and has dramatically reduced latency in the most recent versions of the garbage collector.
6. It validates for the interface and type embedding.

FEATURES OF GO LANGUAGE

- **Language Design:** The designers of the language made a conscious purposeful to keep the language simple and easy to understand. The entire detailing is in a few pages and some interesting design decisions were made through *Object-Oriented* support in the language. Towards this, the language is opinionated and recommends an *idiomatic* way of achieving things. It prefers Composition over Inheritance.
- **Package Management:** Go merges modern day developer workflow of working with *Open Source projects* and includes that in the way it manages external packages.

Support is provided directly in the tooling to get external packages and publish your own packages in a set of easy commands.

- **Powerful standard library:** Go has powerful standard library, which is distributed as packages.
- **Static Typing:** Go is static typed language. So, in this compiler not just work on compiling the code successfully but also ensures on type conversions and compatibility. Because of this feature Go avoid all those problems which we face in dynamically typed languages.
- **Testing Support:** Go provides us the unit testing features by itself i.e., a simple mechanism to write your unit test parallel with your code because of this you can understand you code coverage by your own tests. And that can be easily used in generating your code documentation as an example.
- **Platform Independent:** Go language is just like Java language as it support platform independency. Due to its modular design and modularity i.e., the code is compiled and is converted into binary form which is as small as possible and hence, it requires no dependency. Its code can be compiled in any platform or any server and application you work on.

3.2 FRAMEWORKS

3.2.1 SPRINGBOOT

Spring Boot is an open-source micro framework maintained by a company called Pivotal. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application. With it, developers can get started quickly without losing time on preparing and configuring their Spring application.

Features

- A lightweight framework.
- Helps with loose coupling dependencies and testability. The modular architecture allows you to pick the parts you need and isolate them.
- Has support for both XML and annotation configuration.
- Provides abstraction on ORM software to develop the ORM persistence logic.
- Compatible with many middleware services.
- Supports the JDBC framework, which improves productivity and reduces errors.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

AUTO CONFIGURATION

Spring Boot automatically configures your application based on the dependencies you have added to the project by using **@EnableAutoConfiguration** annotation.

The entry point of the spring boot application is the class contains **@SpringBootApplication** annotation and the main method.

Spring Boot automatically scans all the components included in the project by using **@ComponentScan** annotation.

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.EnableAutoConfiguration;

@EnableAutoConfiguration
public class DemoApplication {
    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }
}
```

@SpringBootApplication automatically adds **@EnableAutoConfiguration**,

@ComponentScan and **@SpringBootConfiguration** annotation.

The **@SpringBootApplication** annotation includes all other annotations.

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class DemoApplication {
    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }
}
```

3.2.2 KEYCLOAK

Keycloak framework is an authentication and authorization server that can be used to secure your applications.

Users authenticate with Keycloak rather than individual applications. This means that your applications don't have to deal with login forms, authenticating users, and storing users. Once logged-in to Keycloak, users don't have to login again to access a different application.

This also applied to logout. Keycloak provides single-sign out, which means users only have to logout once to be logged-out of all applications that use Keycloak.

3.2.3 DOCKER

Docker is an open source containerization platform. It enables developers to package applications into containers—standardized executable components combining application source code with the operating system (OS) libraries and dependencies required to run that code in any environment.

Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allows you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you do not need to rely on what is currently installed on the host. You can easily share containers while you work, and be sure that everyone you share with gets the same container that works in the same way.

Docker provides tooling and a platform to manage the lifecycle of your containers:

- Develop your application and its supporting components using containers.
- The container becomes the unit for distributing and testing your application.
- When you're ready, deploy your application into your production environment, as a container or an orchestrated service. This works the same whether your production environment is a local data center, a cloud provider, or a hybrid of the two.

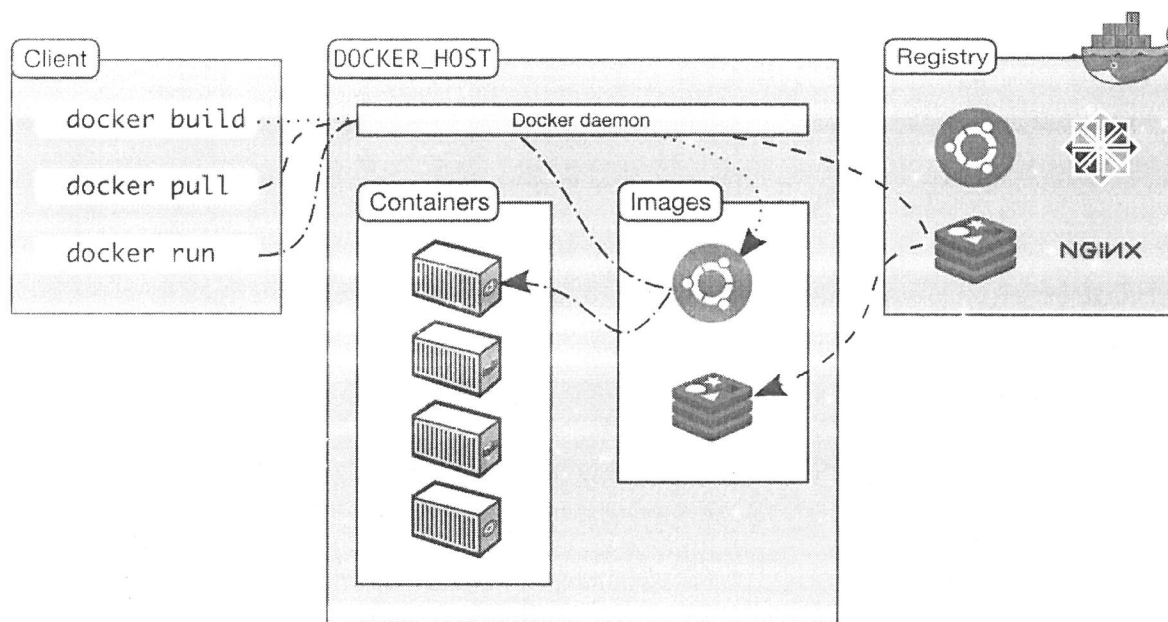


FIGURE 3.2.3: Docker Architecture

THE DOCKER DAEMON

The Docker daemon (dockerd) listens for Docker API requests and manages Docker objects such as images, containers, networks, and volumes. A daemon can also communicate with other daemons to manage Docker services.

THE DOCKER CLIENT

The Docker client (docker) is the primary way that many Docker users interact with Docker. When you use commands such as `docker run`, the client sends these commands to dockerd, which carries them out. The docker command uses the Docker API. The Docker client can communicate with more than one daemon.

3.3 REST API

A REST API (also known as RESTful API) is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services.

An API is a set of definitions and protocols for building and integrating application software. It's sometimes referred to as a contract between an information provider and an information user—establishing the content required from the consumer (the call) and the content required by the producer (the response).

REST is a set of architectural constraints, not a protocol or a standard. API developers can implement REST in a variety of ways.

When a client request is made via a RESTful API, it transfers a representation of the state of the resource to the requester or endpoint. This information, or representation, is delivered in one of several formats via HTTP: JSON (Javascript Object Notation), HTML, XLT, Python, PHP, or plain text. JSON is the most generally popular file format to use because, despite its name, it's language-agnostic, as well as readable by both humans and machines.

In order for an API to be considered RESTful, it has to conform to these criteria:

- A client-server architecture made up of clients, servers, and resources, with requests managed through HTTP.
- Stateless client-server communication, meaning no client information is stored between get requests and each request is separate and unconnected.
- Cacheable data that streamlines client-server interactions.
- A uniform interface between components so that information is transferred in a standard form. This requires that:
 - resources requested are identifiable and separate from the representations sent to the client.
 - resources can be manipulated by the client via the representation they receive because the representation contains enough information to do so.
 - self-descriptive messages returned to the client have enough information to describe how the client should process it.

- hypertext/hypermedia is available, meaning that after accessing a resource the client should be able to use hyperlinks to find all other currently available actions they can take.
- A layered system that organizes each type of server (those responsible for security, load-balancing, etc.) involved the retrieval of requested information into hierarchies, invisible to the client.
- Code-on-demand (optional): the ability to send executable code from the server to the client when requested, extending client functionality.

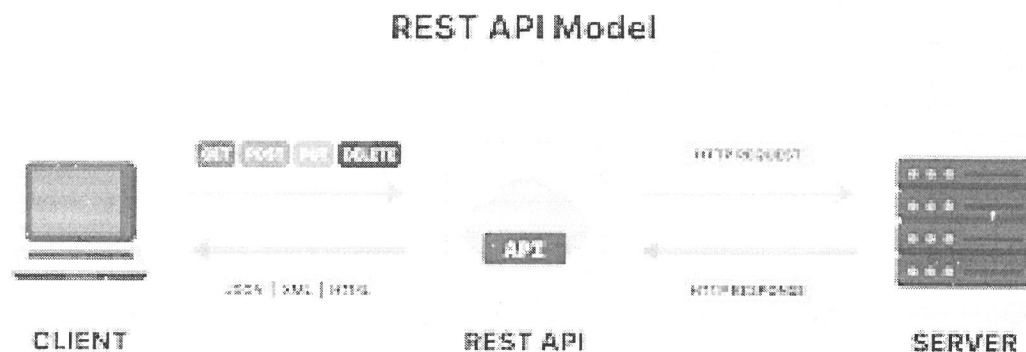


FIGURE 3.3: Rest API Model

3.4 DATABASE

3.4.1 VERTICA

Vertica is a columnar data storage platform designed to handle large volumes of data, which enables very fast query performance in traditionally intensive scenarios.

The Vertica Analytic Database consists of the following key features:

- **Columnar Storage and Execution** - column stores offer significant gains in performance, I/O, storage footprint, and efficiency when it comes to analytic workloads. With columnar storage the query only reads the columns needed to answer the query.
- **Real-time loading and querying** - with high query concurrency and the ability to simultaneously load new data into the system, Vertica can load data up to 10X faster than traditional row-store databases.
- **Advanced Database Analytics** - a set of Advanced In-Database Analytics allows you to conduct the analytics computations closer to the data. This provides immediate results from a single place without having to extract data from a separate environment.
- **Database Designer and Administration Tools** - these features allow you to tune and control Vertica with minimal administration effort.
- **Advanced Compression** - aggressive encoding and compression allows Vertica to dramatically improve analytic performance by reducing CPU, memory, and disk I/O at processing time. Vertica can reduce the original data size by up to 90%, to as little as 1/10th of its original size, without loss of information or precision.
- **Structured and Semi-Structured Data** - in addition to traditional structured database tables, Vertica provides flex tables that let you load and analyze semi-structured data such as data in JSON format.
- **Massively Parallel Processing** - a robust and scalable parallel processing solution provides active redundancy, automatic replication, failover, and recovery.

- **Deploy Anywhere** - run on physical hardware located in your own (or a co-located) data center. Or run on virtual hardware on your own virtual hosts or in the major cloud platforms (AWS, Azure, and Google Cloud).
- **Data Lake Connections** - analyze data from Apache Hadoop and Kafka plus many other systems using standard client libraries such as JDBC and ODBC.
- **Dynamically Scale Your Cluster to Meet Your Workload** - use Eon Mode to scale your database cluster up to meet increased workloads, or scale it down to save money.

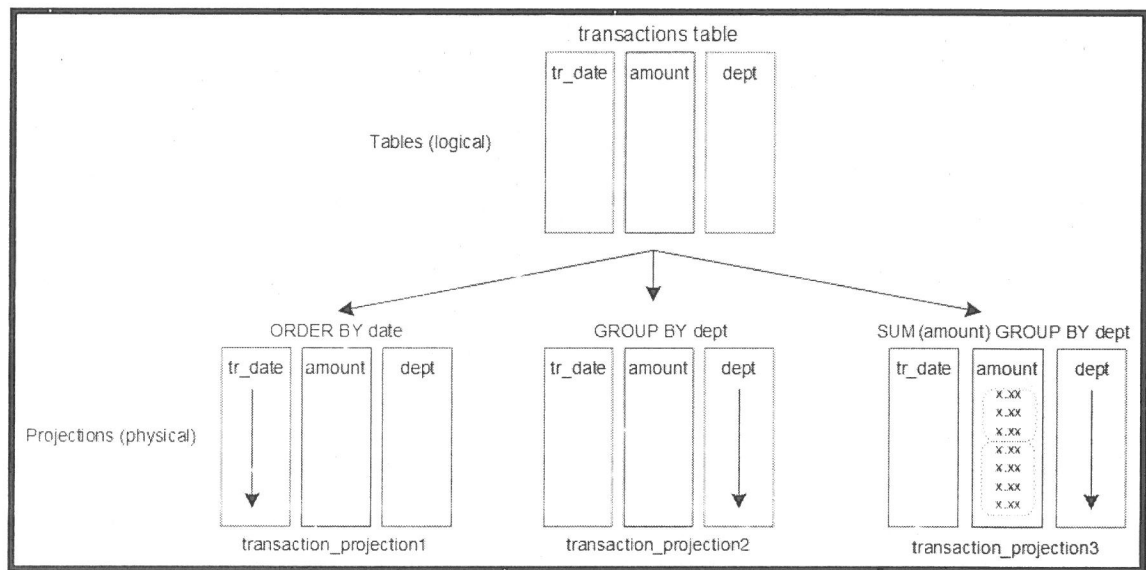


FIGURE 3.4.1: Vertica Projection

3.5 COLLABORATION PLATFORM

- Slack - Slack is a messaging app for business that connects people to the information that they need.
- GitLab - GitLab is a management platform for Git repositories that provides integrated features like continuous integration, issue tracking, team support, and wiki documentation.
- Jira -Jira Software is part of a family of products designed to help teams of all types manage work.

3.6 TOOLS

3.6.1 MOBAXTERM

MobaXterm is your ultimate toolbox for remote computing. In a single Windows application, it provides loads of functions that are tailored for programmers, webmasters, IT administrators and pretty much all users who need to handle their remote jobs in a more simple fashion. MobaXterm provides all the important remote network tools (SSH, RDP, X11, SFTP, FTP, Telnet, Rlogin, ...) to Windows desktop, in a single portable exe file which works out of the box. Some plugins can be used to add functions to MobaXterm such as Unix commands (bash, ls, cat, sed, grep, awk, rsync, ...)

3.6.2 ECLIPSE

Eclipse is an integrated development environment used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment.

3.6.3 POSTMAN

Postman is an application used for API testing. It is an HTTP client that tests HTTP requests, utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated

3.6.4 PUTTY

PuTTY is a free implementation of SSH (and telnet) for PCs running Microsoft Windows

CHAPTER 4 : CONCLUSION

At Sandvine Technologies, working as an intern gave me a lot of experience and knowledge and I am very grateful for this opportunity. Working as a professional in Sandvine has motivated me to learn new things and technologies. My contributions to the projects were appreciated and I got constant support from my teammates and mentor.

At last, I was able to complete the features given to me and it was a great experience to do internship in Sandvine Technologies.

REFERENCES

- <https://www.baeldung.com/spring-boot>
- <https://www.google.com/>
- <https://www.geeksforgeeks.org/>

Sandvine Technologies (I) Pvt. Ltd.

RMZ Ecoworld SEZ,
Building 1A, Ground Floor,
Devarabeesanahalli, ORR
Bengaluru 560103
India

t: +91 (80) 6774 3333

f: +91 (80) 6774 3232

e: info@sandvine.com

www.sandvine.com

INTERNSHIP CERTIFICATE

Date: 03/05/2022

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Deeksha Gupta has done her internship at Sandvine Technologies, Bengaluru, from 03-01-2022 to 03-05-2022 for the duration of 4 months.

As part of internship she worked in the department of Regulatory R&D at Sandvine Technologies. Assigned project was aimed at providing solutions for telecom operators and agencies. As part of the project, she was part of development of backend modules.

During her internship she has demonstrated her skills with self-motivation to learn new skills. Her performance exceeded our expectations and she was able to complete the project on time.

We wish her all the best for her upcoming career.

Sandvine Technologies

 *Varun N Kumbheshwar*

Varun N Kumbheshwar

Manager R&D

