

Keystone

Unified Identity & Access Control Platform

Keystone is a cloud-based solution that seamlessly integrates Identity providers and Cloud Identities for secure access management across Google Cloud Platform (GCP) and FAO Maps.

This presentation delves into the architecture, key components, and implementation details of this framework.

Carlo CANCELLIERI

Co-presenter omitted



Keystone

Features

GCP User Identification

Keystone provides tokens for secure Identification

GCP Resource access control

User can direct access a controlled list of GCP resources

GCP User and Resource Management

Keystone ensure granular permission management



Keystone

Platform Overview

Secure Access

Role-based access control for GCP and FAO Maps resources.

Centralized Management

Unified user management across Cloud identity and FAO Maps.

Simplified Integration

Streamlined integration with Google Cloud IAM and Cloud Identities.

Keystone

Key Components Overview

1

Gateway Service

FastAPI-based service for API interactions.

Enforces policies for bucket & biquery permissions.

2

Python Client Library

Simplifies interaction with the Gateway API.



Keystone

Architecture Integration



Keystone Hub

Centralize
identity management
And
policy enforcement.

Identity

Unify
Platform
And
Google Cloud
identities

Authorization

Management of
Google Cloud
And
FAO Maps specific
access permissions



Keystone



Security & Access Control

Open ID Connect Based Authentication

Secure authentication through
integrated token generation and
validation.

Access Policies

Detailed permission mapping for GCP
and FAO Maps resources.

RBAC and ReBAC

Role-Based and Relational Based Access
Control
for users and service accounts.

Keystone

API Capabilities

Gateway

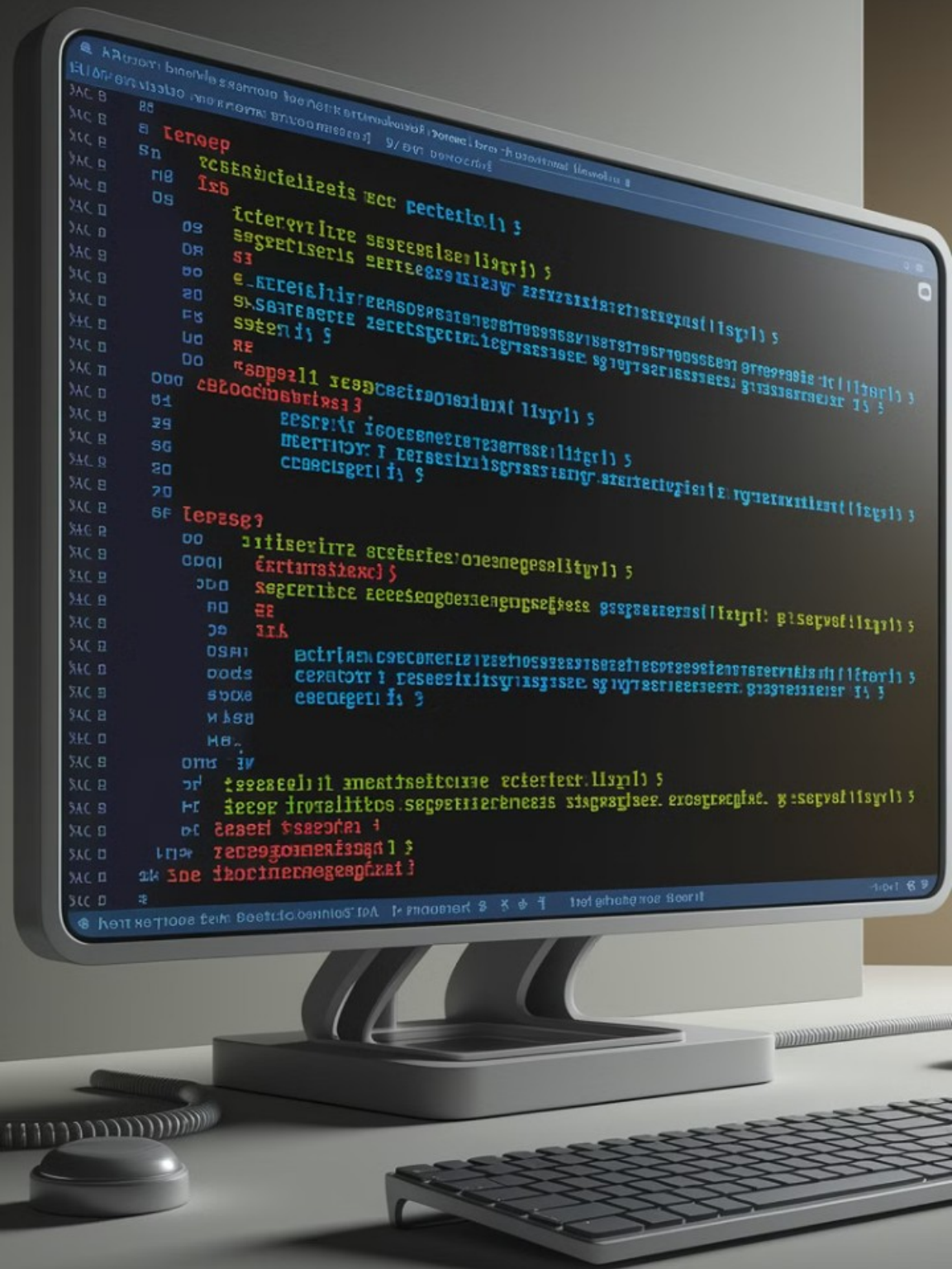
- Bucket access control
- Role verification
- Service authentication

Authentication

- Token management
- User administration
- Group controls



Keystone



Deployment & Setup

1

Prerequisites

python environment

2

Repository Setup

pip install auth-permission-client

3

```
from auth_permission_client.permissions.client import PermissionsClient

# Initialize client
client = PermissionsClient(base_url="http://your-auth-service-url")

# Check permissions (headers from your application's request context)
if client.can_read("my-bucket", headers=request.headers):
    print("You can read this bucket")
```

Administration & Configuration Walkthrough

1

Role Creation

Create administrator roles and system users.

2

Group Creation

Create groups and assign policies for fine-grained permissions.

3

Policy Assignment

Define access rules for specific resources.

Key Takeaways

Keystone is a comprehensive identity and access management platform that unifies security across GCP and FAO Maps while providing easy administration through Keycloak integration.

○ Unified Identity Platform

○ Robust Security Architecture

○ Flexible Integration

○ Streamlined Administration

Keystone demo

Authentication flow:

Thank You

Thank you for your attention. For additional information about Keystone's identity and access management platform, please contact our support team.



THANK YOU