COUPA SOFTWARE
Access Review

Haoran Liu
Kaiyu Liu
Rundong Liu

Problems

Modern companies today usually deploy dozens of web applications and have many users involved as well.

Failing to manage users' access to resources may cause serious security problems.

Example:

Hr should not have access to research data.

Project manager should not read sales report.

Access Control

In computer security, general access control includes authorization, authentication, access approval, and audit.

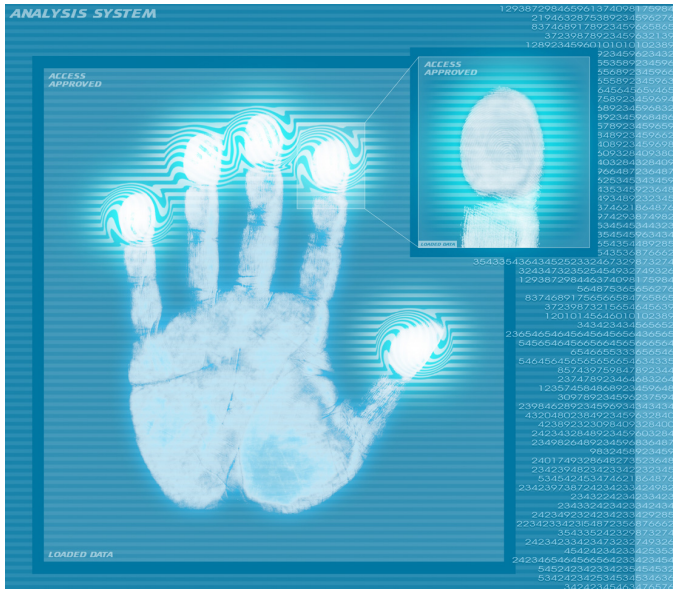
A more narrow definition of access control would cover only access approval, whereby the system makes a decision to grant or reject an access request from an already authenticated subject, based on what the subject is authorized to access.

Access Control

The access control mechanisms, which the user sees at the application level, may express a very rich and complex security policy. A modern online business could assign staff to one of dozens of different roles, each of which could initiate some subset of several hundred possible transactions in the system. Some of these (such as credit card transactions with customers) might require online authorization from a third party while others might require dual control.

Managing Access Control Is Hard

Most of the compliance programs like SOC 1, ISO 27001, etc. require at least quarterly user access reviews to verify that any terminated employees don't have accounts and the users have the correct privileges.

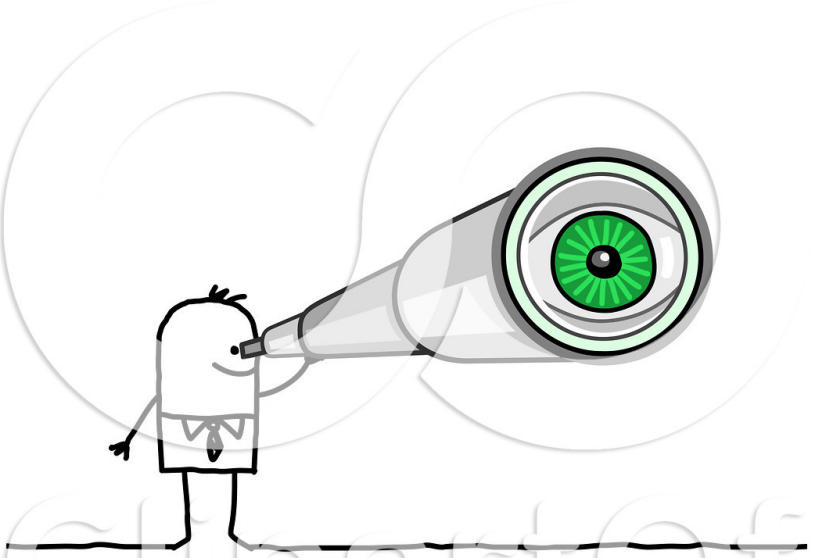


Managing Access Control Is Hard

At most companies, this is a very manual process and may take lots of effort.
Access control list is not visible to manager and not easy to manage.

Solution

Make the ACL more readable and easy to management.



Design

Implement a Django web application to make access review visible and easy.

It provides a portal for:

1) Managers to review and approve user access reviews

2) Auditors to verify user access reviews

Design

The application has a Postgres database capturing access control lists from repositories at Coupa.
It will also integrated with SaaS applications and servers using API and SSH.

Why We Choose Django?

1. Fully Featured Out Of The Box

Django is even more fully featured than most other frameworks, coming with everything you need to build a web app right out-of-the-box.

2. Lots of Packages

There are hundreds of these packages that make it easy to do things like add google maps, create complex permissions, or connect to stripe for payment processing.

3. Portability

Django includes a layer between the developer and the database called an Object Relational Mapper which makes it possible to move your whole project between most major databases by changing just 1 line of code.

4. Provider Support

Because django is a big, well established web application framework, cloud providers go out of their way to ensure it is easy and fast to deploy django apps to their platform.

5. Built-In Admin Panel

This admin panel lets non-developers create/update/delete users and any other database objects specific to your app.

6. Scalability

At its heart django is a series of components that come wired up and ready-to-go by default, but because these components are decoupled, they can be unplugged and replaced.

Why PostgreSQL

- Immunity to over-deployment
- Legendary reliability and stability
- Extensible
- Cross platform
- Designed for high volume environments
- GUI database design & administration tools

Functionality -- Manager Review

- Ability to obtain a list of existing users and their permissions on a given service
- Ability to pull users and managers from HR system

A mechanism managers can use to approve the listing user/service/permission

Functionality -- Auditors Review

Desired functionality

- Ability to provide a “differential” between what a user had during a previous review, and what they have now
- Mechanism to check the access control history, in order to ensure the access security of company.

Architecture

- Web based UI
  HTML5+CSS(Bootstrap)+JAVASCRIPT+JQUERY

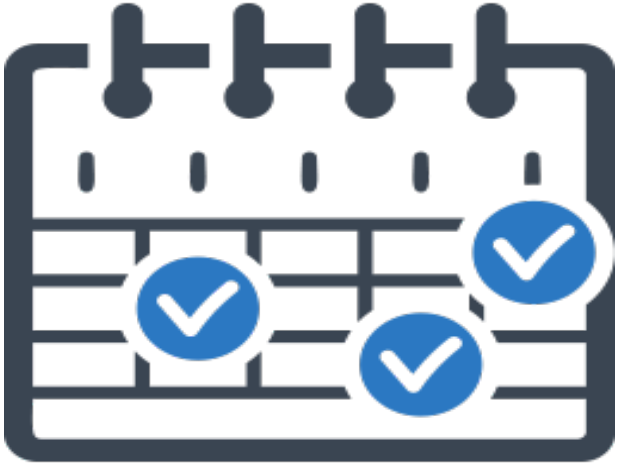
- Back end with a scalable API that can be used to integrate other services in Coupa.

Architecture

○ Specific backend integration/connectors with the following vendors/services

- Google
- AWS
- RightScale
- GitHub

Schedule



- Week 3

Data Available, and the whole architecture design

- Week 6

Realize the functionality: Manager can review the subordinate’s access list.

- Week 9

Realize the functionality: HR can audit the access history.

- Week 12

Final Demo

Q & A