Multiple Organization Architecture (Multitenancy): Overview

An Auth0 Integration Guide to Architectures that must Accommodate Application Instances for Multiple Organizations or Brands

Table of contents

Introduction

In the Business-to-Business (B2B) world, you are selling to businesses: your users belong to different organizations that have signed up for your service. These users are often, and sometimes exclusively, employees of the different organizations that have signed up for your service. However they may also be customers of whatever organizations you are selling to. Whatever the situation, whether you are architecting your integration with Auth0 or you are a developer looking for help designing an integration, this document should give you a high level overview of what common use cases we see with respect to multi-tenant applications.

Most B2B applications strive to create a pleasant user experience for the employees/customers of the businesses they serve. To accomplish this most services in B2B add some branding to the service for each of the organizations that use the service. For example, let's say you work for AwesomeSaaS (a SaaS software company) and your company uses Human0 an HR application for managing benefits, etc. You would access your HR app at https://awesomesaas.human0.com, and when you log in you would see the AwesomeSaaS logo and the login experience would be customized to use AwesomeSaaS colors.

When architecting your integration with Auth0, the first thing you will need to consider is whether or not your customers (we'll call them organizations) will allow users from other organizations to log into their instance of the application. We need to know whether or not those users are shared between organizations or isolated to one particular organization.

Let's introduce a couple of examples of applications that will help highlight the differences.

Travel0 is a fictitious company that offers travel agency and other related services online. When navigating this document, think of yourself as an employee of Travel0. Travel0 has several applications, but for the purpose of this exercise we'll focus on the two applications that are marketed directly to organizations:

- **Travel0 Corporate Booking**: This application provides organizations with an online application where their employees can log into the application and book work related travel. Organizations that are customers of this application include:
    - **Hoekstra & Associates**: a small law office with just a couple of employees. They do not have an IT department and don't have the time or capacity to learn how to setup a corporate Identity Provider (IDP).
    - **Gupta & Smith Law**: a larger law office, but they also do not have an IT department and don't have the time or capacity to learn how to setup a corporate IDP.
    - **MetaHexa Bank**: a large finance organization. They provide banking and insurance services and have their own IDP.

    - **Many Student University (MSU)**: a large university with several campuses. Each campus has its own IDP.

- **Travel0 Adventure Management**: This application allows organizations to create and market adventures (white water rafting, horseback riding, zip line, etc). It allows guides (who are freelance, or employee of some 3rd party travel/event organization) to sign up for or be scheduled to lead adventures. Organizations that are customers of this application include:
    - **AdventureZ**: a large tour/event guide. They have their own IDP that they use for their employees. They rarely, if ever need freelancers because they just have enough guides on staff - some of which only work during the busy times. They also facilitate their guides' ability to do freelance work for other companies.
    - **Rocky Mountain High Adventures**: a new group coming into the market for the first time. Just the co-founders run tours, and they mostly reach out to freelancers for help during busy times.
    - **Suzie's Rafting and Ziplines.** This company has been around for a long time. They have a staff of guides that handle most of their events, but will also reach out to freelancers when busy.

Terminology

Let's take a step back for a moment and clarify some terminology. This is important for this document because many of the words used in this document can be overloaded to mean many things. We will be diligent in our use of terms so that we can keep the meaning specific to the intended use in this document in an attempt to avoid confusion. Please take a moment to read through each definition so that it is clear which companies are filling which roles when reading through the examples. This terminology will be consistent through all of the other documents associated with this overview as well.

Application Tenant: we will avoid using this term as much as possible to avoid confusion, but where necessary it will refer to a tenant in *your* application as opposed to the Auth0 Tenant. Instead of using this term, we will use "Organization" or "Organization Instance."
Auth0 Tenant (Authorization Server): the Auth0 tenant that you create in Auth0. It is your Authorization Server and represents a user domain.
Employee: a person who is part of your company. They likely have an account in your Identity Provider (IdP). They may need admin access to organization instances. NOTE: your customers may have users who are also employees, but we will refer to those as Organization Users as we don't know if they are employees or not. We will only refer to Employees of your company.
Identity Provider (IdP): a service that manages authentication of users and optionally user profile information and credentials for an organization, company or group; or the service may delegate the credential validation and profile management to another IdP. Example IdPs are: your Auth0 tenant, your Azure AD instance, Google, Facebook, etc.

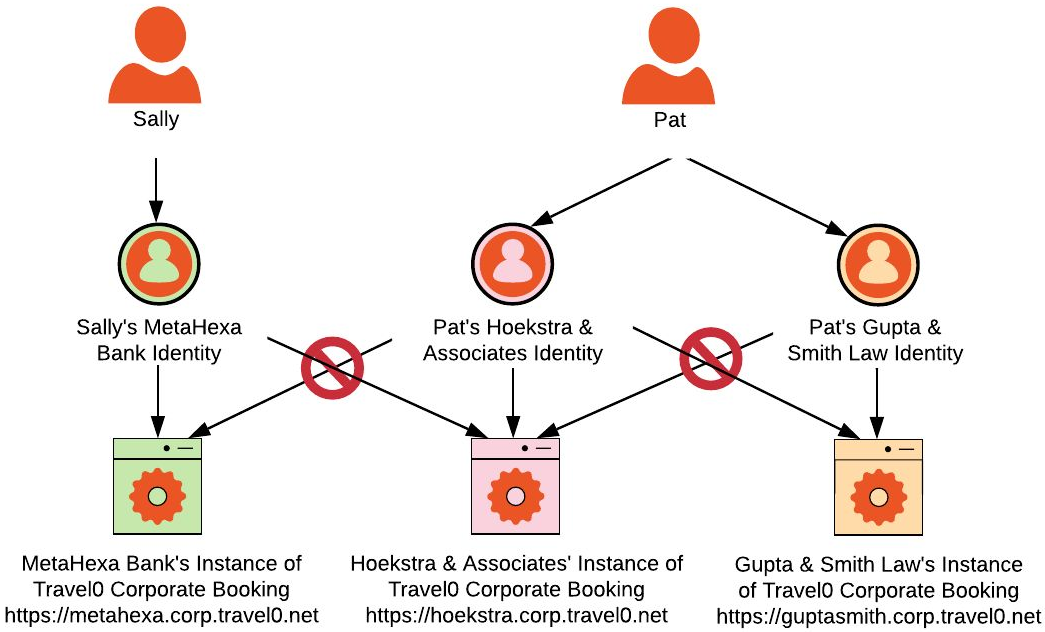
Organization: a company that is a customer of yours. If you refer to organization instances of your applications as tenants, we will refer to them as organizations to avoid confusing the term with the Auth0 tenant. This is a replacement for the term Application Tenant to avoid confusion.
Organization User: the person who is logging into the application as a member of one of your organizations.

To Share or Not to Share

Now that we know what each word means, let's get back to discussing how we determine which type of application you are providing. This will require that we first take a look at some specific user cases and determine which way we want to go with each case.

An organization should map directly to one of your business customers/partners. There are two different approaches regarding how to store your organization users; pay close attention to the users that need access to more than one organization, these users will help to determine which approach more closely maps to your company's requirements.

- **Isolated to the organization:** every user belongs to exactly one organization. In this use case, it would not make sense for a user to be a part of more than one organization, and even if they were, we would rather they create a separate "identity/user" for each organization. Using Travel0 Corporate Booking as an example, the diagram below shows how this would look; for more information regarding this type of scenario see the integration guidance entitled Users Isolated by Organization.



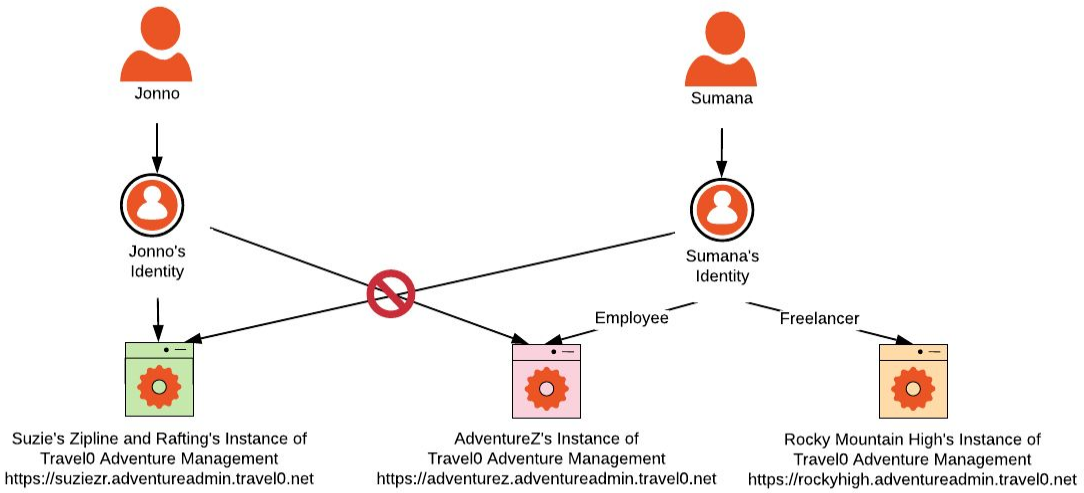
Sally is a typical user in this environment. Sally is an employee of MetaHexa Bank and she can only access the MetaHexa Bank's instance of Travel0 Corporate Booking.

Pat is atypical for this type of environment, either Pat doesn't exist for your company or is a rare user. We are including Pat as an example of the decision that is being made when isolating users to their organization. Pat is a freelance paralegal and does some work for both Heokstra & Associates and Gupta & Smith Law.

Here is where your environment will dictate the decision you need to make: if you want users to be isolated to their organization you're making the decision that Pat must create two separate users, one for accessing Heokstra & Associates's instance of Travel0 Corporate Booking and a separate user for accessing Gupta & Smith Law's instance of Travel0 Corporate Booking. This makes sense in this scenario, and probably reduces accidental error situations by forcing Pat to create two separate personas, one for each law firm. So that when Pat books travel it requires a separate login to the specific organization instance in order to make the booking.

- **Shared between organizations:** In this scenario we no longer tie the user's credentials directly to the organization they belong to or have access to. Users now have two options for how they log in: a) they create credentials in your company's identity store (username/password database in your Auth0 tenant) rather than any identity store specifically allocated in your Auth0 tenant for their organization or b) they log in using their own organization's IdP. Once a user has an identity, we then give them permission to access the organizations that they should have access to. This could mean access to

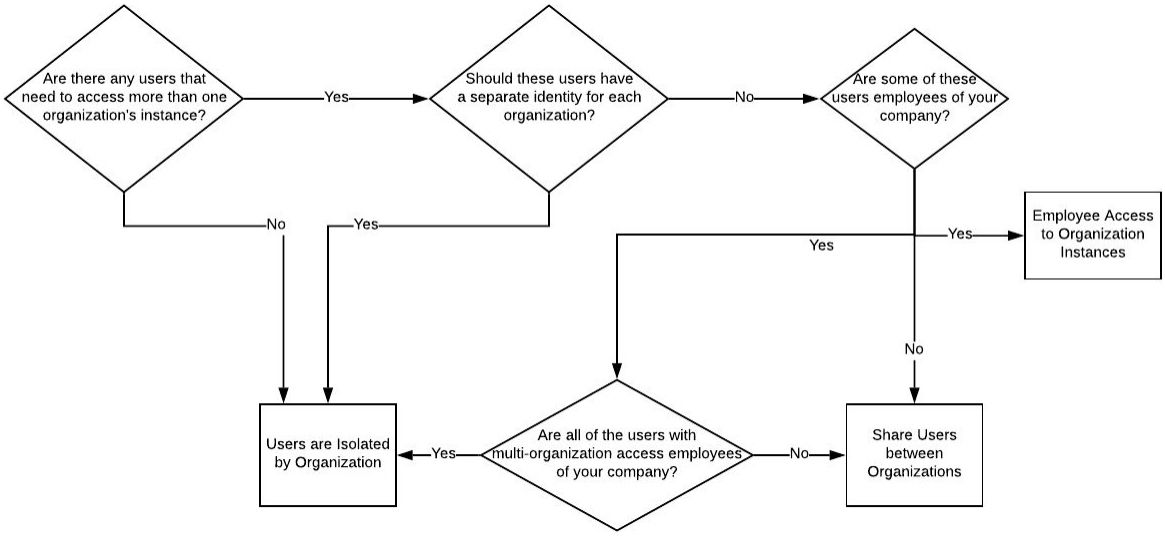
just one organization, or it may mean they have access to more than one organization. Users will need to understand when prompted to log in that they can use those same credentials to access each organization's instance. Using Travel0 Corporate Booking as an example, the diagram below shows how this would look.



Jonno is a typical user. Jonno is an employee of Suzie's Rafting and Ziplines. Jonno is only able log into Suzie's Instance of Travel0 Adventure Management to create and guide adventures. Jonno's credentials are either stored in a Travel0 database connection or in Suzie's Zipline and Rafting's IdP depending on whether Suzie wants to setup the connection or not.

Sumana is an employee of AdventureZ, but AdventureZ also coordinates freelance opportunities for the smaller guide companies during high peak times. Sumana has been invited by Rocky Mountain High Adventures to freelance. Sumana is authorized to log into both AdventureZ and Rocky Mountain's instances of Travel0 Adventure Management. However, since she has never been invited to guide for Suzie's Rafting and Ziplines, she is not authorized to access that instance. Sumana needs to have the same identity for both organizations because the guide system involves a rating system. Sumana's ratings need to carry over and be combined between organizations. Sumana's credentials, like Jonno's, are either stored in a Travel0 database connection or in AdventureZ's IdP depending on whether Suzie wants to setup the connection or not. It has no bearing on the organization instances she has access to.

Now that we have defined two different approaches to user isolation, let's walk through how to make this decision. Here are the questions you need to ask yourself to determine which approach you will need to take:



*Even if only a small percentage of your users belong to more than one organization, you need to structure your system so users **can** belong to more than one if you want to support it for **any** of your users.*

Users are Isolated by Organization

Each organization has its own set of users and users can not and should not be able to access other organizations. If they attempt to, they should be rejected as unauthorized (see Users Isolated by Organization), keep in mind that you can choose to force your users create a separate account for each organization even if they belong to more than one as a person, they would be considered two different users.

Share Users between Organizations

A user may belong to more than one organization and it would be convenient if that user did not have to have a separate identity/account as they navigate from one organization to another. Organizations can still use their own IDP in shared user scenarios. We will have more details on this in a future document.

Employee Access to Organization Instances

Do your employees need to be able to log in to the organization's instances? If so, you will need to have an enterprise connection to your IDP and a way to redirect your employees to that IDP (possibly a special login URL that redirects to Auth0 using connection=<your IDP>). We will have more details on this in a future document.