

Parte 4 - Investigación

¿Qué es una IDP?

Una IDP, Identity Provider, es un servicio encargado de crear, almacenar y gestionar las identidades digitales de los usuarios en aplicaciones o sistemas. Autentica cualquier entidad conectada a un sistema, como humanos, computadoras y otros dispositivos, y proporciona los datos necesario para que pueda acceder a un servicio o recurso de manera segura.

USOS:

- Autenticación de Usuarios:
Verifica la identidad de los usuarios al acceder a aplicaciones o servicios.
- Gestión de Acceso:
Otorga diferentes permisos para acceder a recursos dependiendo el usuario.
- Autenticación Multifactor:
Asegura el acceso al requerir múltiples formas de verificación, menos riesgo de acceso no autorizado.
- Cumplimiento de acceso para Proveedores y Clientes:
Gestiona, no solo empleados, pero también proveedores, socios o clientes.
- Autenticación adaptativa:
Utiliza métodos de autenticación basados en contexto y riesgos, equilibrando la seguridad con la experiencia del usuario.
- Gestión de acceso a APIs:
Protege el acceso a las APIs.
- Portales de autoservicio para usuarios:
Permite a los usuarios gestionar sus perfiles y contraseñas.
- Inicio de sesión único (SSO):
Optimiza el acceso permitiendo a los usuarios ingresar a múltiples aplicaciones con un solo conjunto de credenciales.

Dockerfile

Dockerfile: Es un archivo de texto que contiene instrucciones para construir una imagen de Docker.

- **Requerimientos:**
 1. Tener Docker instalado en la máquina.
 2. Definir una aplicación base, es decir, qu;e aplicación queremos colocar en un contenedor.
 3. Identificar las bibliotecas, configuraciones o herramientas que necesitemos en nuestra aplicación.
 4. Preparar las configuraciones específicas de la aplicación si es necesario.
 5. Identificar los puertos que tu aplicación usa para comunicarse.

- **Imagen Base:**

Esta imagen depende del lenguaje que use tu aplicación. Algunos ejemplos son:

- Node.js `node:16`
- Python `python:3.9`
- Java `openjdk:11-jdk`
- Base de Datos `mysql:latest`

- **Puertos a Exponer:**

Depende de los puertos que utilice la aplicación, esta información se puede consultar en la documentación de esta. Por ejemplo:

- Node.js `3000`
- Apps web `80` o `443`
- Base de Datos `3306`

- **Archivo de Configuración:**

Es un archivo donde se definen parámetros o opciones de configuración que determinan el comportamiento de ese sistema, el cual es utilizado por una aplicación. En lugar de cambiar el código fuente de una aplicación, estos archivos permiten que los usuarios ajusten los aspectos de los funcionamientos sin tener que modificarlos de forma directa.

Code-Server

Es una versión de Visual Studio Code que se ejecuta en un servidor y permite acceder al editor de código a través de un navegador web.

Docker Compose:

Es una herramienta que complementa Docker, que permite definir y gestionar aplicaciones de múltiples contenedores. Permite describir la configuración del entorno de desarrollo en un archivo, se especifican servicios, volúmenes y redes necesarias para tu aplicación.

Un **Servicio** representa un contenedor o un conjunto de contenedores que ejecutan una misma imagen de Docker, es una unidad lógica que representa cómo ejecutar una aplicación dentro de un contenedor.

Un **Volumen** es un mecanismo para compartir datos entre contenedores y permite que los datos de la aplicación no se pierdan cuando el contenedor se elimina o reinicia.

un **Puerto** es un punto de acceso lógico o un número de identificación que permite que los dispositivos o servidores se comuniquen con otros. Es una forma en la que aplicaciones accedan a los recursos de red que están disponibles en otra máquina.