CASO #2

- Documentacion: https://developer.okta.com/docs/reference/rest/
- Tutorial sincronizacion OTKA - Postman: https://www.youtube.com/watch?v=u1Fqh4KneXI

Pasos a seguir

1. Importar las colecciones de Groups, Users y Factors:

- https://developer.okta.com/docs/reference/postman-collections/

2. Crear los 3 grupos con la colección "Groups": (Marketing, Sales y C-Level)

- Crear grupo: Group Operations > Add Group

3. Crear 4 Usuarios con la colección "Users":

- Creamos el usuario perteneciente a un grupo: Create User > Create User in Group
- Le asignamos una contraseña a los usuarios: Credential Operations > Set Password

Usuarios: (nombre - contraseña)

- Yareth Morataya - minchurila2908
- Amy Morataya - lucas2908
- Patricia Sandoval - cuco2908
- Ricardo Morataya - minchu2908

Grupos y Usuarios:

- Marketing: Yareth Morataya
- Sales: Amy Morataya y Patricia Sandoval
- C-Level: Ricardo Morataya

4. Habilitar MFA

- Documentación: https://developer.okta.com/docs/guides/mfa/ga/main/
- Se implementa el autenticador de Google

Para testear el MFA vamos a crear un usuario con "Create User without Credentials"

- Usuario: Lucas Duarte - ubersam28@gmail.com
- ID Usuario: 00uj2eog2iEGxU7Zx5d7

Ahora vamos a Enroll the factor con la sub-colección "Factor Lifecycle Operations" que se encuentra en la colección "Factors"

- Seleccionamos "Enroll Google Authenticator Factor" y en la URL reemplazamos {userId} por nuestro ID Usuario para testing

- ID Factor: uftj2ep1yqm0U6wel5d7
- Link proporcionado para obtener código QR: https://dev-62745605.okta.com/api/v1/users/00uj2eog2iEGxU7Zx5d7/factors/uftj2ep1yqm0U6wel5d7/qr/20111sHG3FkkTL1LIjf41utfVuqhNhn9OIkl-vHtUFu-LwL6DB60t5Y

Se realizan los pasos del apartado "Activate the factor":

- Se obtiene este ID: uftj2ep1yqm0U6wel5d7

5. Asignación de permisos

- Vamos a crear una cuenta en: https://auth0.com

- Creamos una Authorization Store, en este caso "GummyBears"

- Nos dirigimos al FGA Dashboard y realizamos el Started Guide, diseñamos el modelo y creamos las reglas de tuplas

- Modelo creado:

Authorization Model

```
type user
  relations
    # Define which groups the user belongs to
    define member_of: [Sales, Marketing, C-Level]
    # Define various permissions the user can have
    define has_permission: [payment_method_permission, contact_info_permission, sales_columns_permission, sales_target_progress_permission]
type Sales
  relations
    # Define members of the Sales group
    define Amy_Morataya: [user]
    define Patricia_Sandoval: [user]
type Marketing
  relations
    # Define members of the Marketing group
    define Yareth_Morataya: [user]
type C-Level
  relations
    # Define members of the C-Level group
    define Ricardo_Morataya: [user]
type payment_method
  relations
    # Define which users have access to this payment method
    define has_method: [user]
type payment_method_permission
  relations
    # Define which users can use which payment methods
    define can_use_payment_method: [user, payment_method]
type CreditCard
```

```
  relations
    # Define which users can use this credit card
    define has_method: [user]
type PayPal
  relations
    # Define which users can use PayPal
    define has_method: [user]
type contact_info_permission
  relations
    # Define which users can modify contact information
    define can_modify_contact_info: [user]
type sales_columns_permission
  relations
    # Define which users can view sales columns
    define can_view_sales_columns: [user]
type sales_target_progress_permission
  relations
    # Define which users can view the sales target progress card
    define can_view_sales_target_progress_card: [user]
```