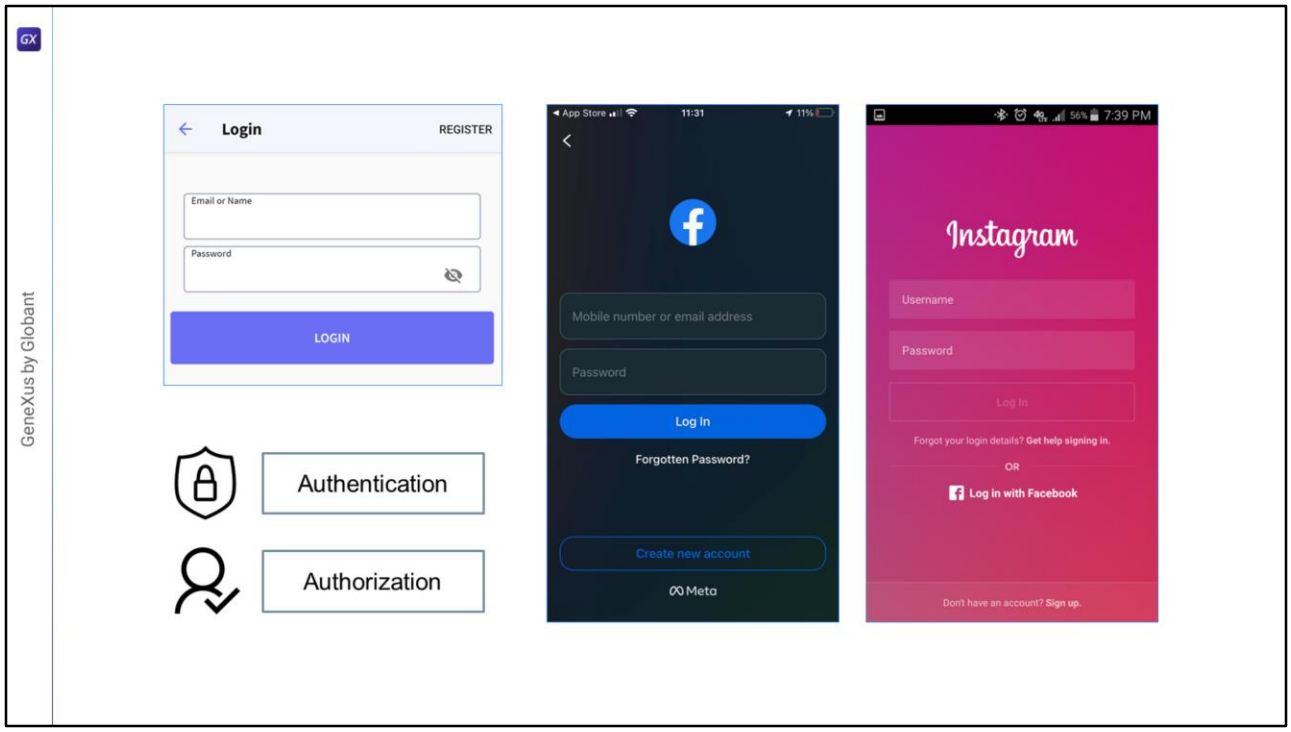
GeneXus by Globant

GeneXus Access Manager

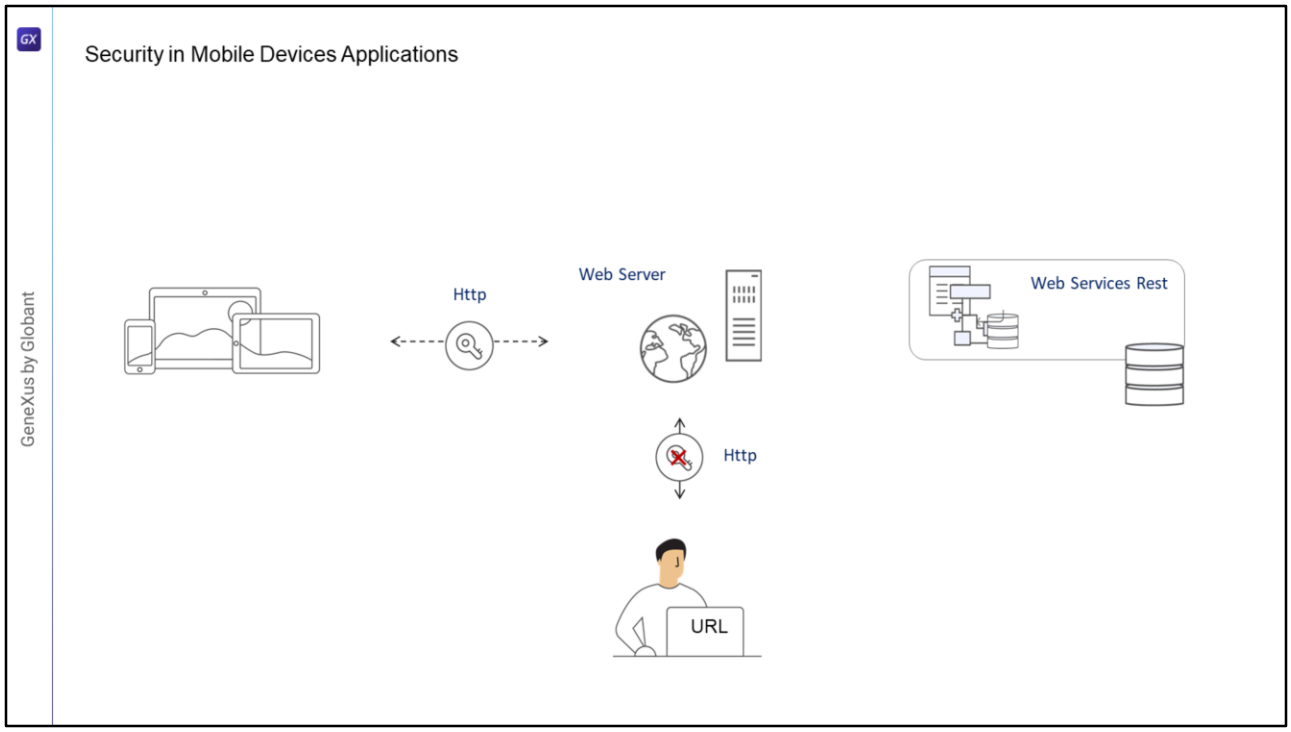
Introduction

Diego Marranghello



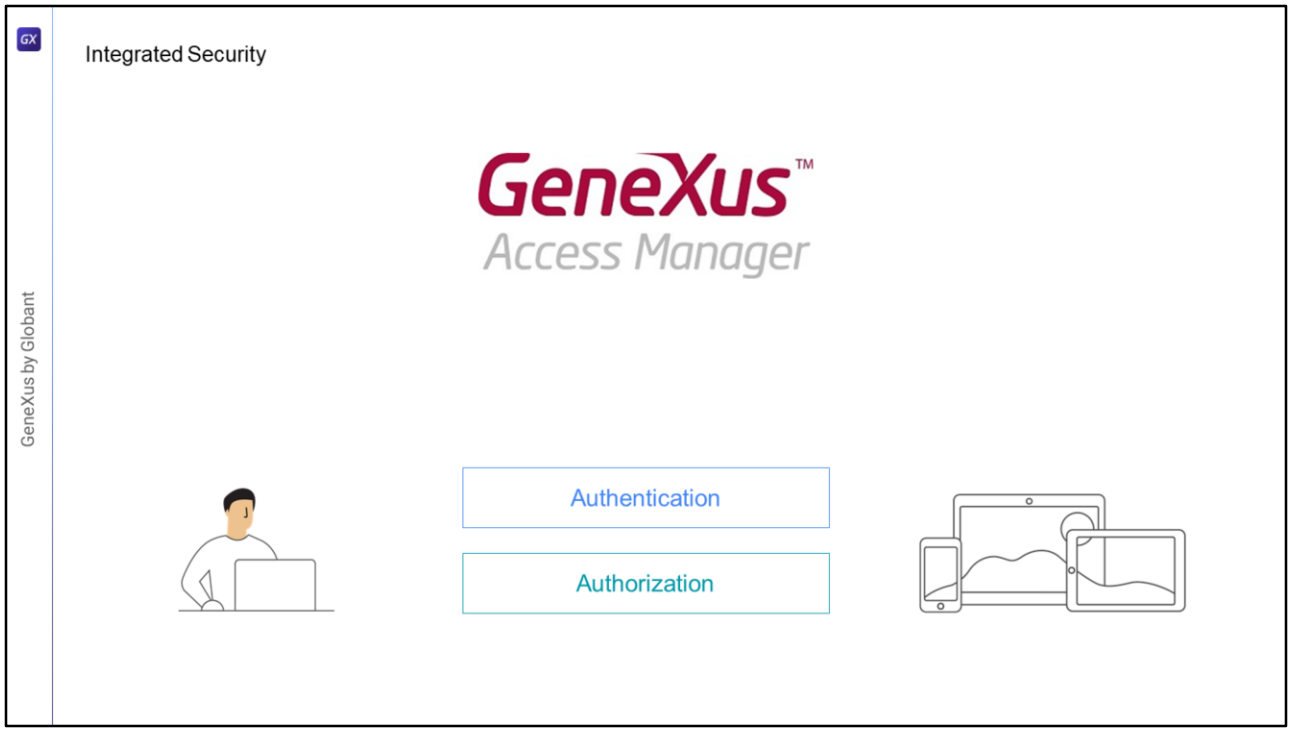
Como ya sabemos , la gran mayoría de las aplicaciones modernas necesitan un esquema de seguridad, para que solo puedan ingresar los usuarios permitidos y también autorizar o restringir el acceso a partes de la aplicación, según los permisos asignados al usuario.

Esto significa asegurar que todos los usuarios que ingresen estén debidamente autenticados y autorizados.



En el caso de la aplicaciones para dispositivos móviles, al ser aplicaciones distribuidas, una parte de ellas se ejecuta en el propio dispositivo, y la capa de negocios de la aplicación se resuelve a través de servicios Rest que tienen una URL de acceso, por lo que están expuestos a accesos indeseados.

Al igual que para las aplicaciones web, lo que se hace es verificar que solamente usuarios debidamente autenticados y autorizados puedan acceder a la aplicación, evitando la ejecución de usuarios que no cumplan con esto.

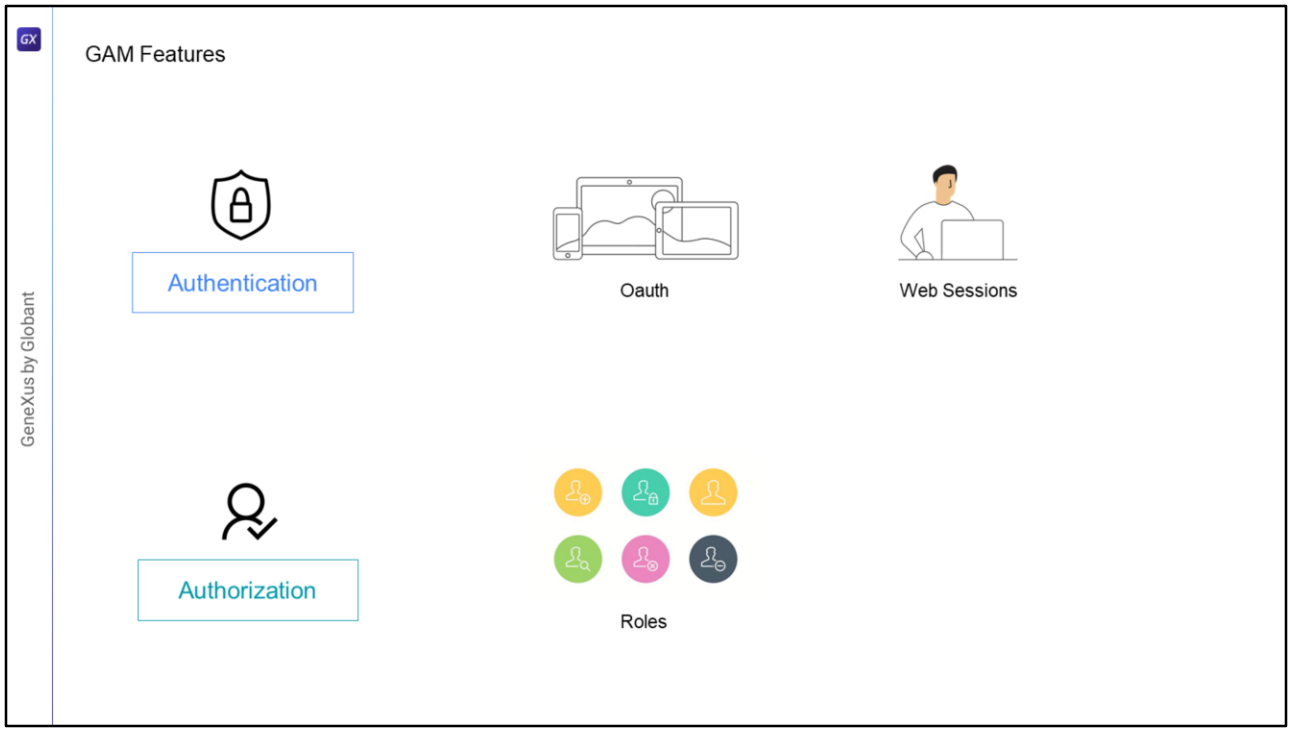


El módulo de seguridad GeneXus Access Manager (GAM) resuelve las funcionalidades de autenticación y autorización, tanto para aplicaciones Web como para aplicaciones para dispositivos móviles.

El GAM está desarrollado en GeneXus por lo que se integra fácilmente a la KB de la aplicación y permite resolver de manera centralizada todo lo referente a la Seguridad de la misma. El objetivo es que la solución de Seguridad se utilice lo más declarativamente posible dentro de la aplicación, sin crear complejidad adicional.

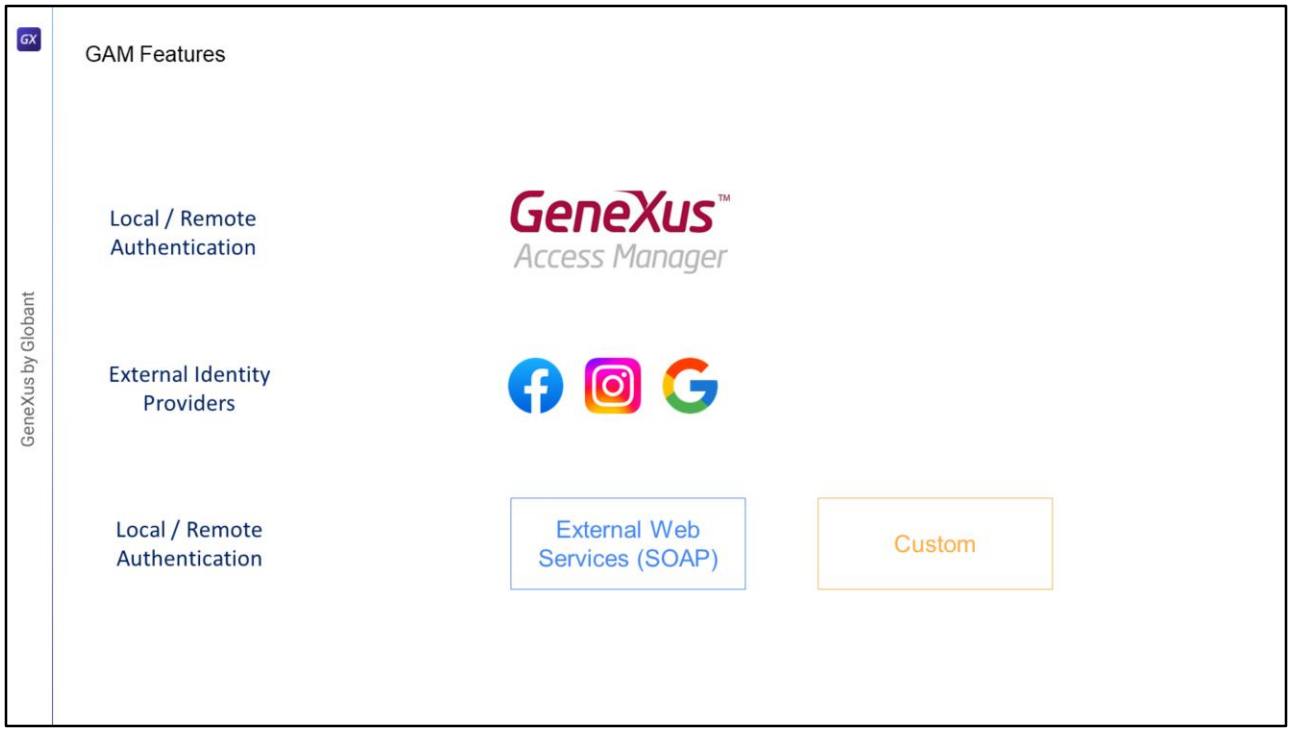
El GAM también provee un backoffice que permite definir usuarios, permisos, políticas de seguridad y acceso a objetos, entre otras cosas.

Además provee una API para poder acceder a muchas de estas funcionalidades en forma programática.



Para resolver la Autenticación en dispositivos móviles, internamente se usa Oauth, a diferencia de las aplicaciones Web donde se utilizan Web Sessions.

En el caso de la Autorización, su implementación está basada en Roles



El GAM provee diferentes Tipos de Autenticación, los tipos disponibles son:

Autenticación local usando GAM donde los usuarios y todas sus credenciales son almacenados en una base de datos de la cual somos propietarios, o también en forma Remota, ya que una aplicación que use GAM puede ser convertirse en proveedor de identidades y en este caso, otras aplicaciones con GAM pueden conectarse remotamente a este server y obtener la autenticación desde allí.

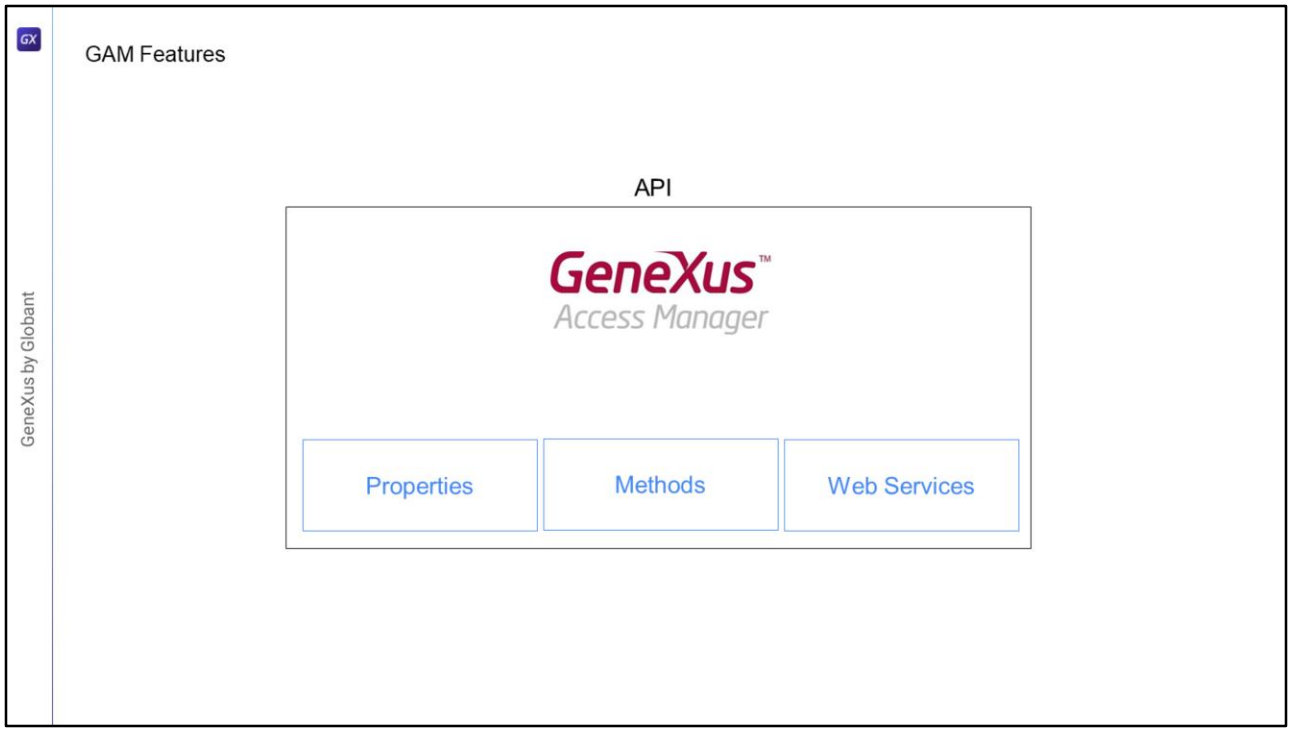
Podemos utilizar también a otros proveedores de identidad externos, estos proveen una autenticación basada en el protocolo Oauth 2.0 como Facebook, Instagram, Google, etc. En este caso no hay necesidad de definir usuarios locales.

En ocasiones es necesario integrar nuestra aplicación con otras, mediante una autenticación externa a la aplicación.
Una forma de autenticación externa es utilizar un web service SOAP que provee la otra aplicación y configurar al GAM para que consuma de ese web service.

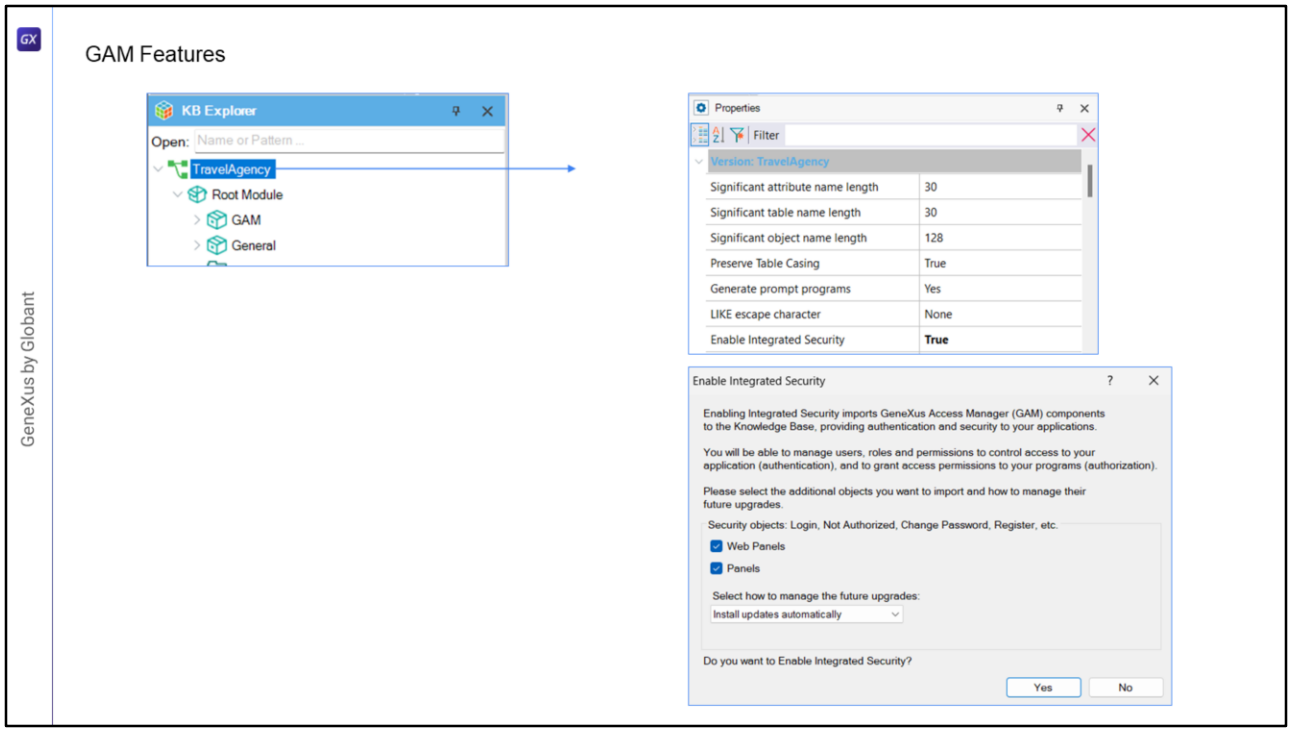
Puede ser que la otra aplicación provea un programa externo para resolver la autenticación, pero que no necesariamente sea un web service. En ese caso es posible configurar el GAM para aceptar una autenticación del tipo Custom.

Con la Autorización, definimos los permisos y ejecución de los objetos y de los modos de operación de las transacciones.

La definición se hace otorgando para cada objeto, permisos a cada rol y en función de cuál sea el rol que tenga asignado el usuario, serán los permisos efectivos sobre el objeto, como por ejemplo en dispositivos móviles: los objetos WorkWith y Panel

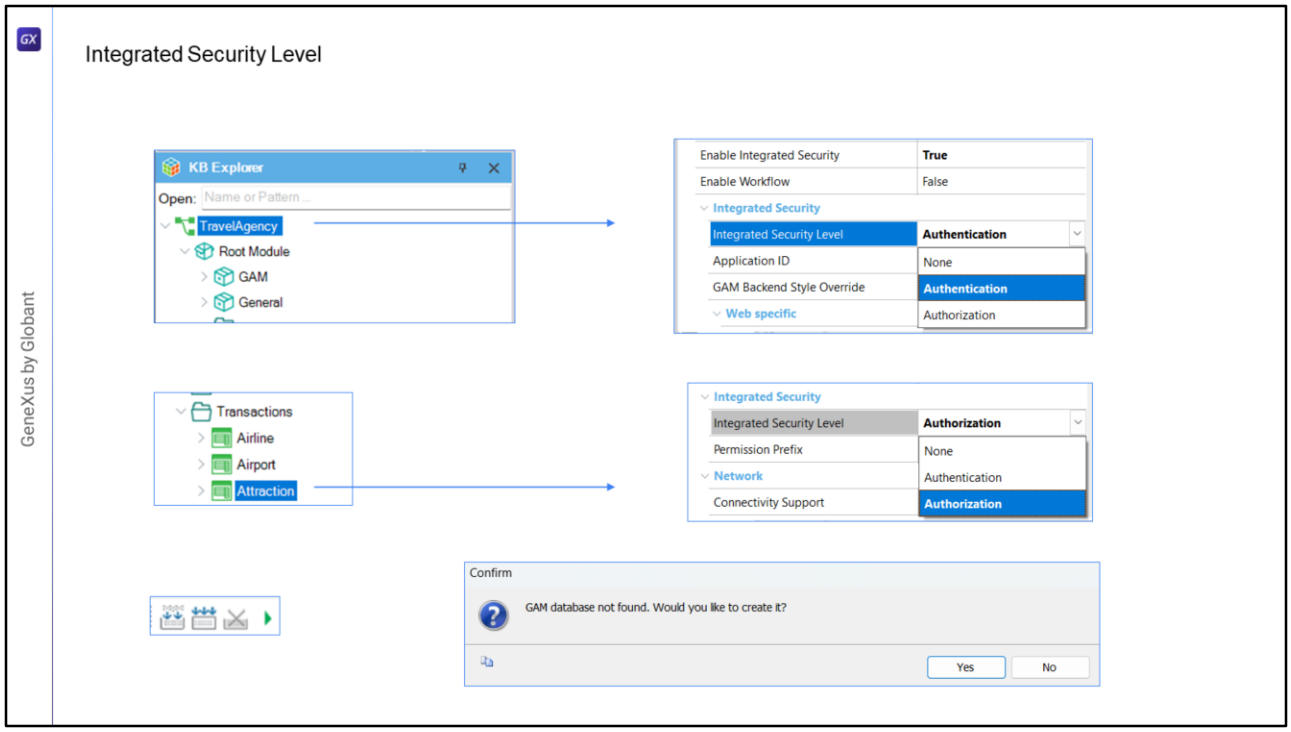


El GAM también expone una API para acceder a sus propiedades y métodos en caso de que sea necesario hacerlo desde nuestra aplicación y una serie de servicios Web que pueden ser utilizados desde otras aplicaciones. No entraremos en detalles sobre esta funcionalidad en este video.



Para habilitar el GAM se debe ir a la versión activa de la KB y configurar la propiedad Enable Integrated Security con el valor True.

Al hacerlo, se abrirá un cuadro de diálogo que nos avisa que se instalará el módulo GAM en nuestra KB, con la solución lista tanto para web como para dispositivos móviles.



Una vez habilitado GAM, veremos otra propiedad llamada Integrated Security Level que permite indicar el valor por defecto de la seguridad de los objetos de la KB.

Esta propiedad se encuentra también a nivel de cada objeto, por lo que será posible personalizar la seguridad de cada objeto.

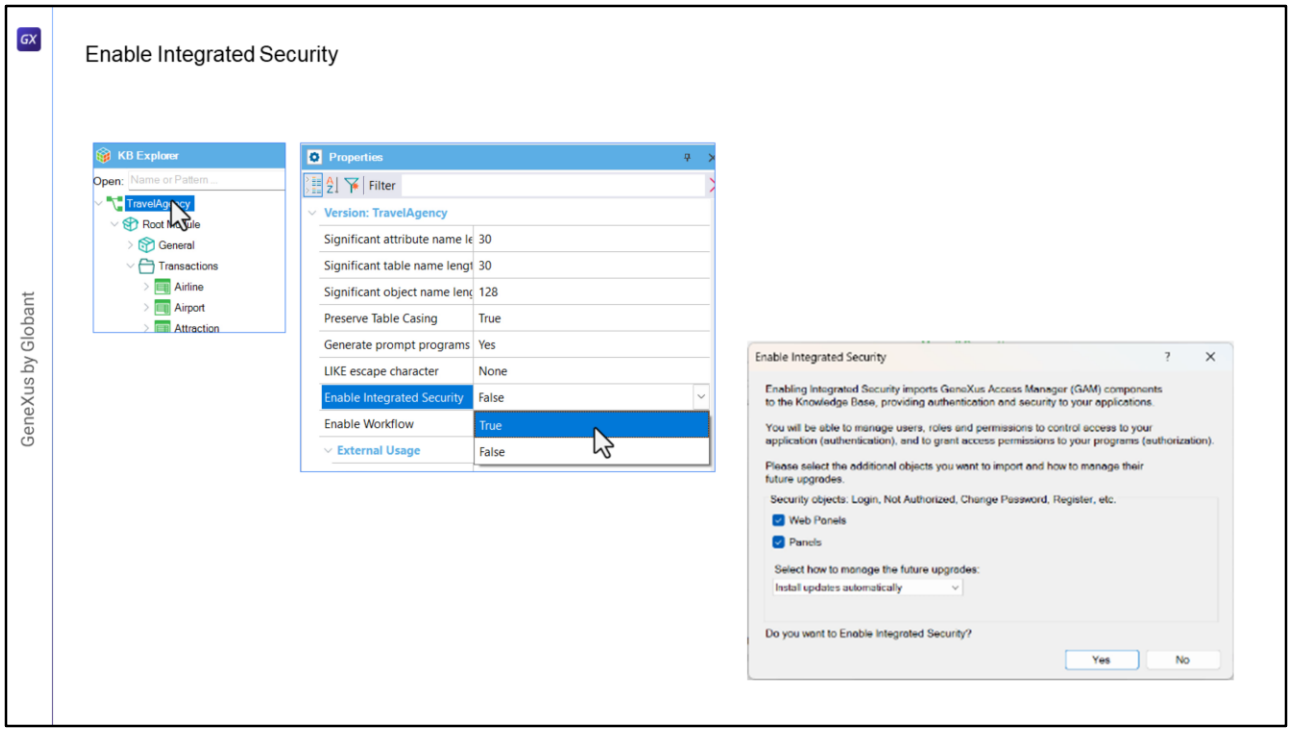
Hay tres valores posibles:

None: indica que el objeto será publico, es decir no tendrá seguridad.
Authentication: indica que sólo usuarios autenticados podrán ejecutarlo.
Authorization: indica que el usuario además de haberse autenticado, tendrá que estar autorizado para ejecutar dicho objeto, es decir tener el rol adecuado para ejecutarlo.

Una vez que tengamos estas propiedades de seguridad configuradas se van a importar en forma automática los objetos de GAM en la KB y luego deberemos hacer un Rebuild All de la misma.

GAM solicitará crear una base de datos independiente de la base de datos de la aplicación. Estará asociada a un Data Store Independiente en la KB, con lo cual toda su configuración es independiente.

Veamos todo esto en un ejemplo.

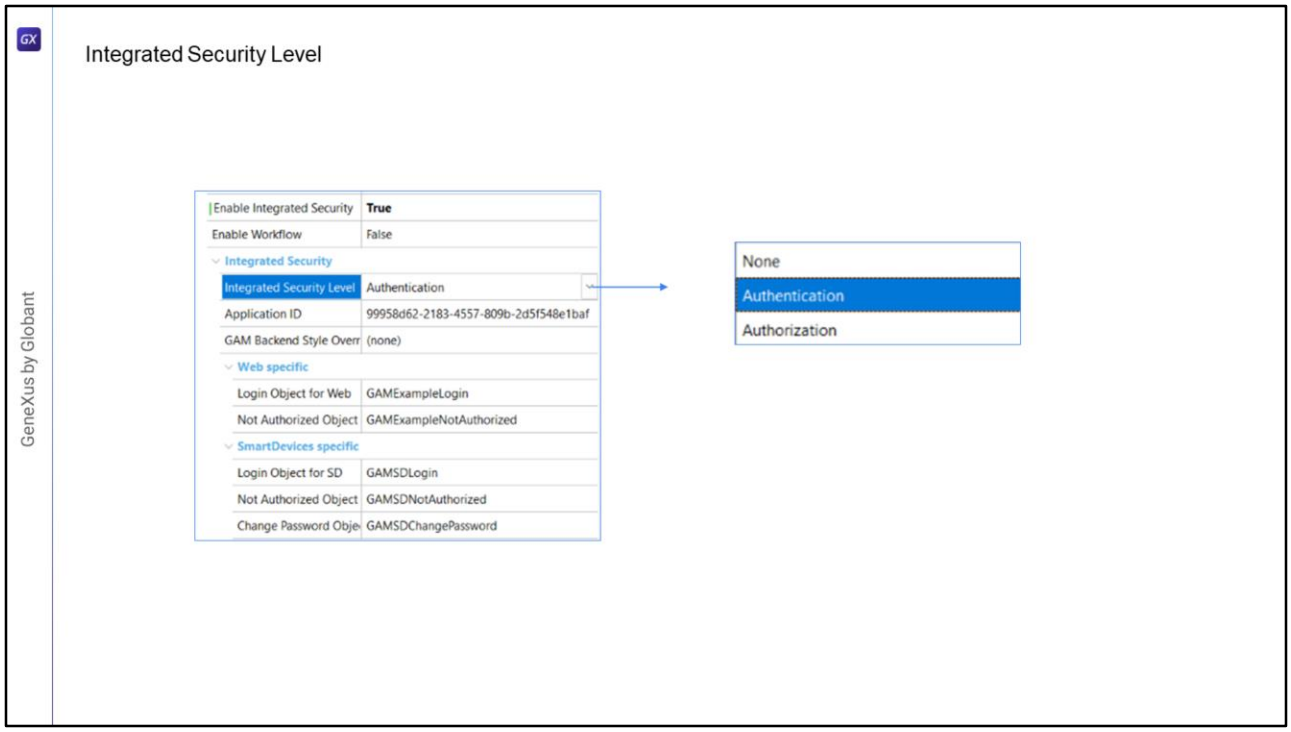


Tenemos creada parte de una aplicación para una agencia de viajes. Vamos a ir a las propiedades de la Base de Conocimientos, hacemos click sobre TravelAgency, el nombre de la KB, y vamos a habilitar GAM poniendo en la propiedad Enable Integrated Security el valor True.

Aquí podemos indicar si deseamos que se integre en los objetos utilizados para aplicaciones Web, y/o en caso que tengamos algún objeto generado para dispositivo móvil, como el caso de los Paneles o el WorkWith usado para mobile, podremos indicar que se integre a este tipo de aplicaciones, en este caso como nos interesa aplicarlo a una aplicación mobile deberá estar seleccionado.

Con este combo podemos elegir cómo deseamos que se actualice este modulo, puede ser automático, podemos elegir que nos pregunte o que nunca se actualice.
Confirmamos.

Aquí comienzan a importarse los objetos de GAM.



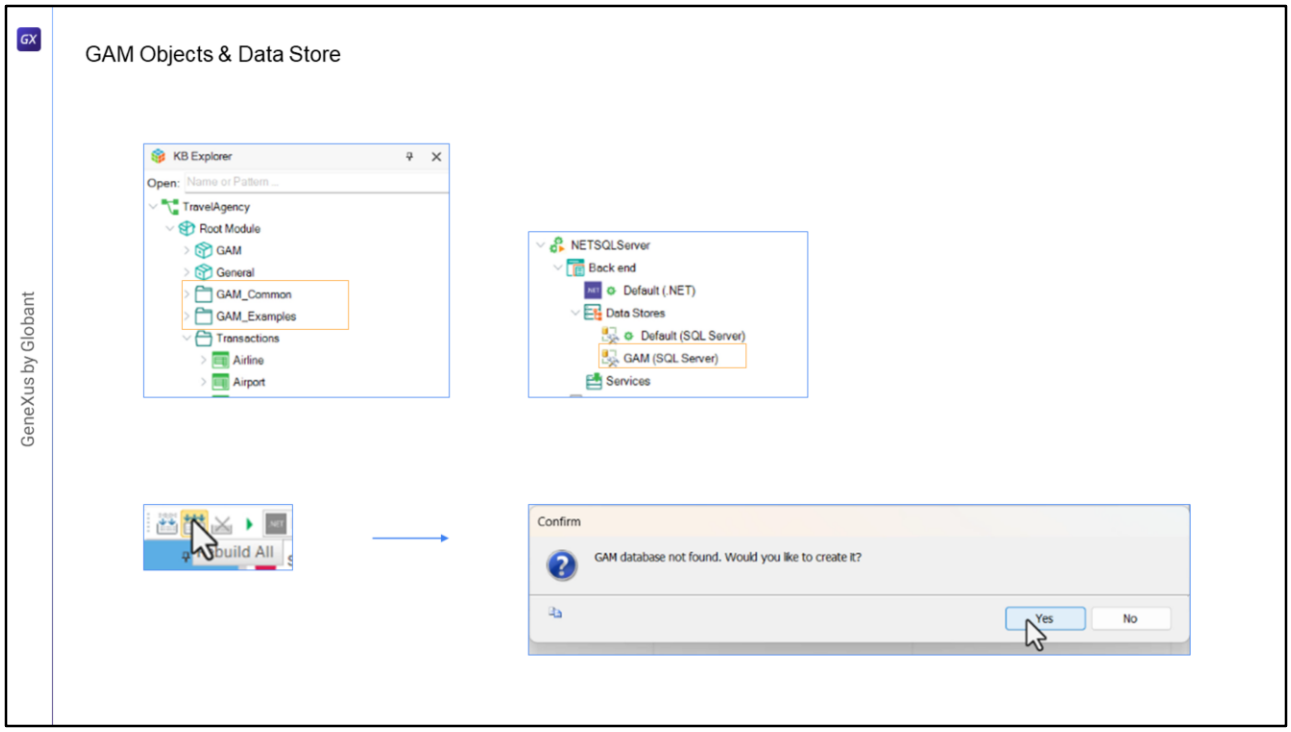
Ahora tenemos disponible la propiedad Integrated Security Level.

En esta propiedad podemos indicar si deseamos habilitar sólo autenticación, que es el valor por defecto, si deseamos autorización, o si no deseamos tener seguridad.

Dejaremos en Authenticaton, lo que hace que para acceder a cualquier objeto que permita seguridad, nos pedirá credenciales de acceso.

Además se asigna un Application ID que se utilizara en el repositorio de GAM para identificar a la aplicación.

Ahora ya tenemos las propiedades donde indicamos los objetos de login, uno en caso de error de autorización y otro para cambiar la contraseña.

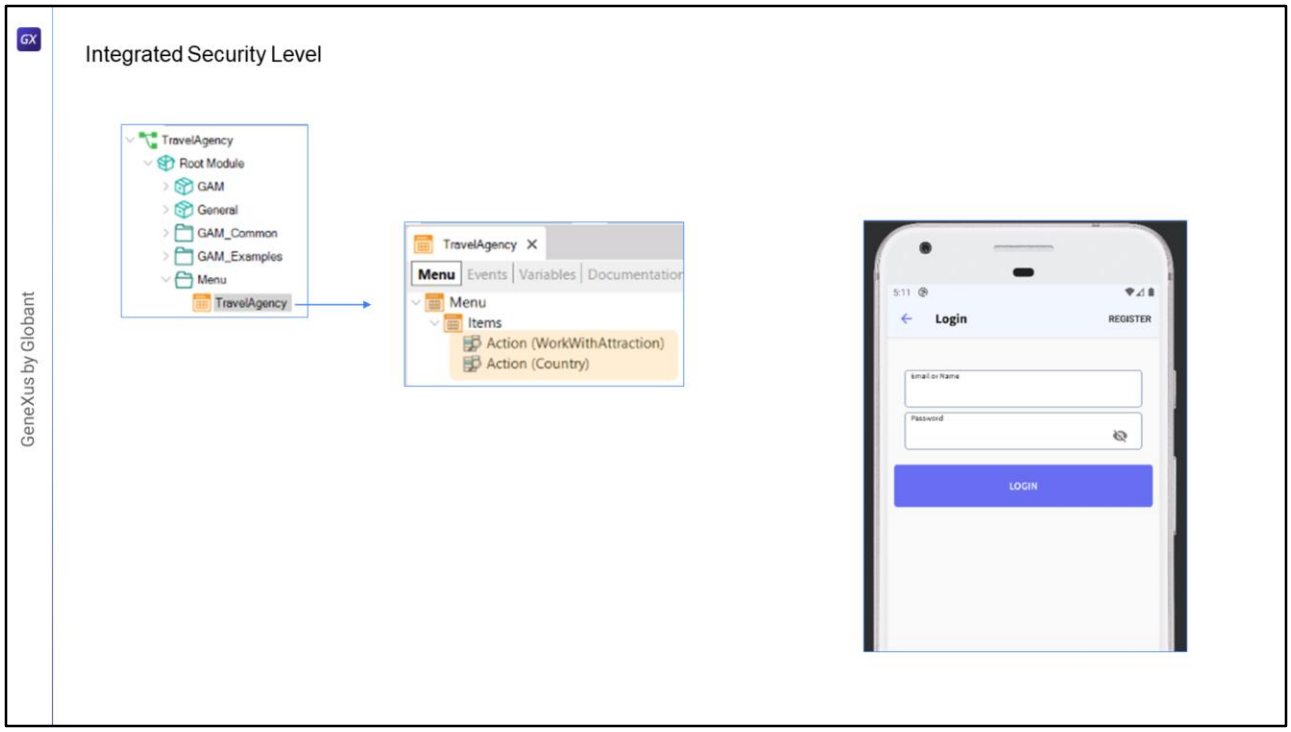


En la KB ya podemos ver que se crearon algunos folders en el root module.

Además tenemos un nuevo Data Store, GAM, con la información de esa conexión.

Bien, al terminar el proceso de importación, vamos a hacer un Rebuild All.

GeneXus nos indica que la base de datos de GAM no se encontró y si deseamos crearla.
vamos a poner que si.
Se crea la base de datos con todas las tablas y luego se inicializa.



En nuestra aplicación tenemos un objeto Menu de nombre TravelAgency, el cual tenemos declarado como startup object, y que tiene estos objetos como ítems para que podamos acceder.

Al ejecutar, lo primero que vemos es la pantalla de login, ya que configuramos que para toda la aplicación tuviera como seguridad Authenticaton.
Y si deseamos ingresar sin usuario nos da un error. Todo esto nos provee GAM en forma automática.

Para ingresar vamos a usar un usuario que se crea por defecto: “admin”, con la contraseña “admin123”.
Y ahí sí nos permite acceder a nuestra aplicación.

Con esto terminamos la introducción al objeto GAM para dispositivos móviles.


GeneXus by Globant
GeneXus™
by Globant
training.genexus.com