

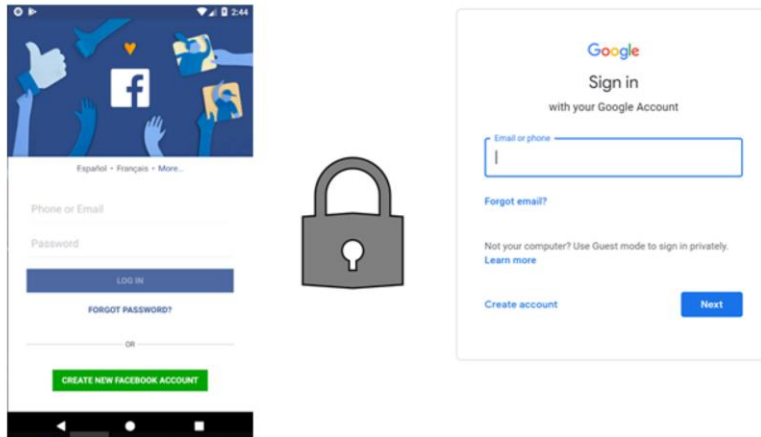
GeneXus[™]
The power of doing.

GeneXus Access Manager - Introduction

Security

GeneXus 16

Security



As we already know, most applications today call for a security schema so that only allowed users may have access, in addition to the authorization or restrictions for access to the various parts of an app, according to the permits assigned to each user.

GeneXus Access Manager - Introduction GeneXus

Security

LOGIN

admin

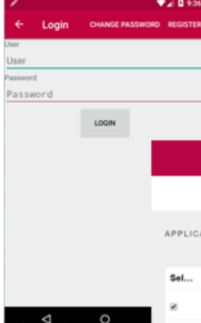
password


Keep me logged in

Remember Me


LOG IN

FORGOT YOUR PASSWORD?
or create an account





Authorization



Authentication

USERS ROLES SETTINGS

Administrator

Add Permission

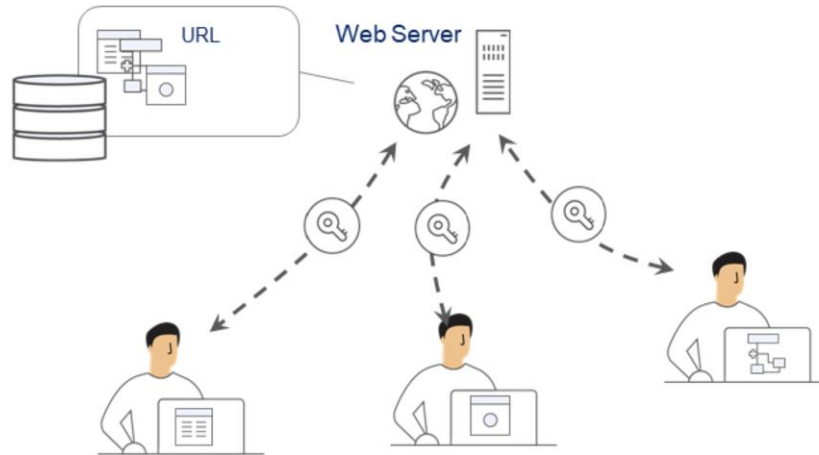
Try a Search

← BACK + ADD SELECTED

APP.	GAM Backend	ROLE	BackedUser
Sel...	Permission name	Description	Permissions options
<input checked="" type="checkbox"/>	gamexamplechangerespository_Execute	Change Working Repository	Allow
<input type="checkbox"/>	gamexamplechangeyourpassword_Execute	Change Password	Allow
<input checked="" type="checkbox"/>	gamexamplewwwapplications_Execute	Application	Restricted
<input checked="" type="checkbox"/>	gamexamplewwwauthypes_Execute	Authentication Types	Deny
<input type="checkbox"/>	gamexamplewwwconnections_Execute	Connections	Allow

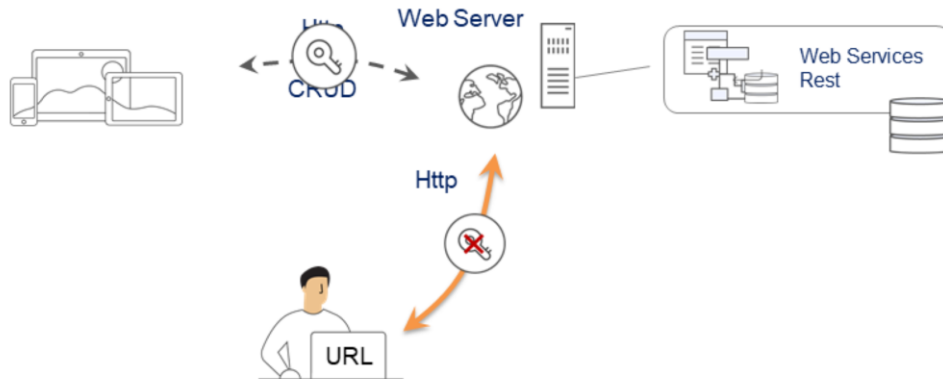
This implies ensuring that all users who access are duly authenticated (that is, the user must be who she/he says she/he is) and authorized (that is, once the user is authenticated, she/he is allowed access to specific parts of the application, or not).

Security in Web Applications



For the case of Web apps, because they have several access points, any object accessible through URL must very verify authentication permits.
This means that each object must include the security check to perform the corresponding verification.

Security in Smart Devices Applications



For the case of apps for Smart Devices, because these applications are distributed, part of them is executed on the device itself and the business layer of the app is solved through Rest services with an access URL, so they are exposed to unwanted accesses.

Just like in the case of web apps, what we must do is verify that only duly authenticated and authorized users access the application, thus avoiding the execution of users who are non-compliant with that condition.

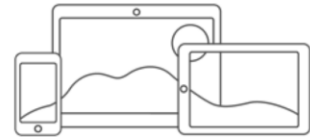
Integrated Security Solution


GeneXus Access Manager



Authentication

Authorization



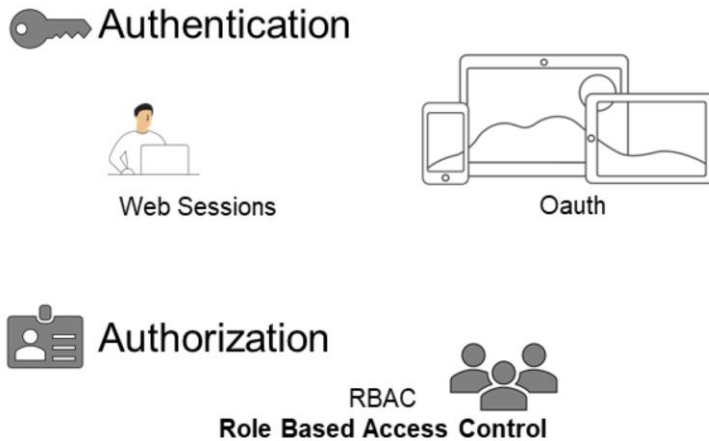
To fulfill all such needs, GeneXus provides us with a security module known as GeneXus Access Manager (GAM), which solves authentication and authorization functionalities for both web and smart device apps.

The GAM was developed by GeneXus so it is easily integrated with the app's KB. It enables a centralized solution for everything relative to security. The purpose is for the security solution to be used as declaratively as possible within the application, without creating additional complexities.

The GAM also provides a backend that enables the definition of users, permits, security policies and access to objects, among other things.

Additionally, it offers an API to access many of these functionalities in a programmatic manner.

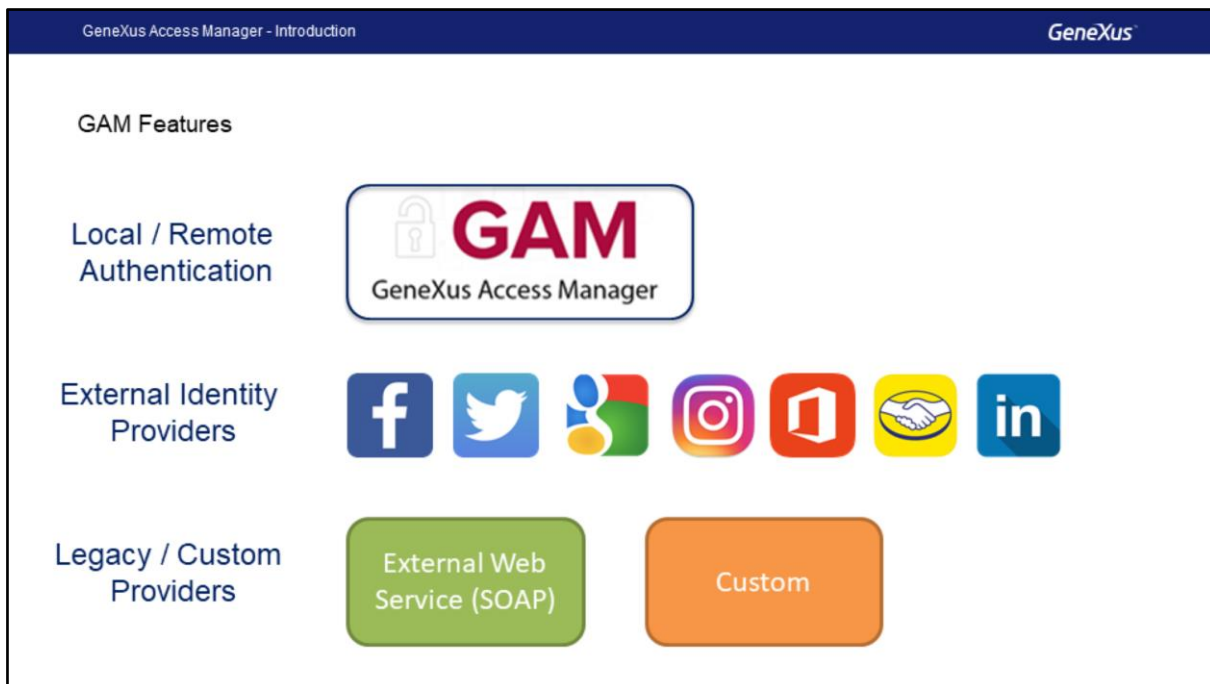
GAM Features



What is used to internally solve the Authentication is:

- **Web sessions** for web applications security
- **Oauth** to solve security in the case of SD apps

For Authorizations, the implementation is based on Roles using the **Role Based Access Control** model with which methods, properties and everything necessary to handle the app's authorization are encapsulated.



The GAM provides different Authentication Types:

Local authentication using GAM, where users and all their credentials are stored in a proprietary database, or also in a **Remote** manner, because an application that uses GAM may become an identity provider, and in such case, other apps with GAM may be remotely connected to this server to obtain authentication from there.

We may also use other external identity providers. These provide authentication based on the **Oauth 2.0** protocol, like **Facebook, Twitter and Google, Instagram, Office 365, Mercado Libre, or LinkedIn**, where standard authentication mechanisms are used, all based on the protocol implemented by these applications. In this case, there is no need to define local users.

It is often necessary to integrate our app with other applications with which we have to exchange information, and it becomes necessary to ensure that users are authenticated through an external authentication of the app.

One way to perform external authentication is to use a **SOAP web service** provided by the other app and to set up the GAM for it to consume that web service.

The other might provide an external program to solve the authentication, but not exactly a

web service. In such case, we set up the GAM to accept authentication of the Custom type.

GeneXus Access Manager - Introduction GeneXus

GAM Features

The screenshot displays the GeneXus Access Manager interface. On the left, there is a 'Speakers' list with columns for Id, Full Name, Image, Company Name, Country Id, and Country. The main area shows a 'Permissions' table with columns for Permission name, Description, Permissions options, and Inherited. Below this is a 'Speaker' form with fields for Id, Name, Surname, Full Name, Image, and CVMini. On the right, there is a mobile app interface titled 'Work With Speaker' showing a list of speakers and a detailed profile for Armin Bachmann.

Permission name	Description	Permissions options	Inherited
gamexamplechangeyourpassword_Execute	Change Password	Allow	DELETE
gamexamplewvapplications_Execute	Application	Allow	DELETE
gamexamplewvauthypes_Execute	Authentication Types	Restricted	DELETE

Id	Full Name	Image	Company Name	Country Id	Country
12	García, Alejandro		GeneXus	5	Uruguay
15	Cardozo, Amanda		Century 21 Karskydtel Prop.	5	Uruguay
23	Bachmann, Armin		GeneXus	5	Uruguay
2	Gonda, Breggan		GeneXus	5	Uruguay

Speaker Form:

Id:

Name:

Surname:

Full Name:

Image:

CVMini:

Mobile App Profile:

Bachmann, Armin
 Mr. Bachmann is a graduate of Computer Science Engineering from the School of Engineering of the University of the Republic of Uruguay, and member of GeneXus support team since 1999.
 GeneXus

With the Authorization we define execution permits of objects and permits for operating modes in transactions.

The definition is done by granting each objects permits for each role. The effective permits on the object will depend on the role assigned to the user.

This validation is done on the following web objects:

- Web Panels
- Web Components with property URL Access=Yes
- Transactions

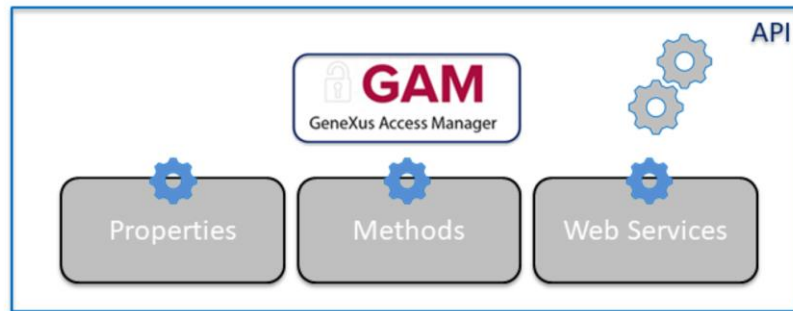
Additionally, a verification is made on permits for the Insert, Update, Delete and Display modes of web transactions.

And for smart devices of the objects:

- Work With for Smart Devices
- Panels for Smart Devices.

And the Insert, Update and Delete actions on the Work Withs for Smart Devices.

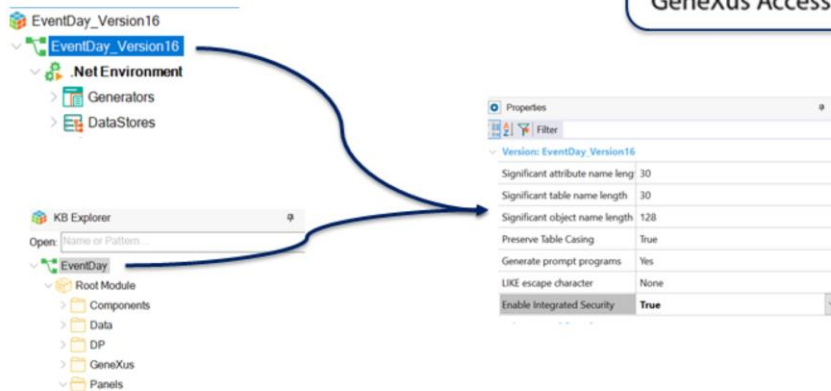
GAM Features



The GAM also shows an API (Application Program Interface) to access its properties and methods in case we need to do it from our app and a series of web services that may be used from other apps.

This topic will be dealt with at the advanced level.

Enable Integrated Security



In order to enable the GAM, we must go to the KB's active version level and set up the **Enable Integrated Security** property to True.

In the Trial version, it is in the first node of KB Explorer under the name of the KB.

GeneXus Access Manager - Introduction GeneXus

Integrated Security Level

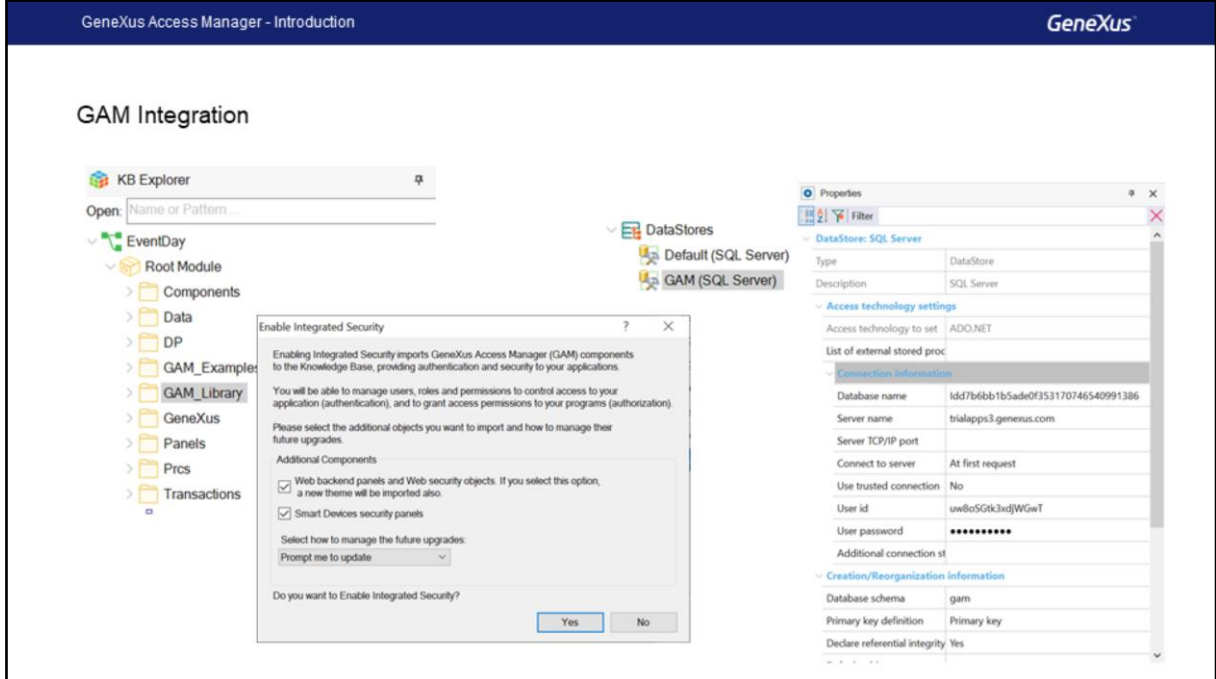
The screenshot displays the GeneXus Access Manager interface. On the left, a tree view shows the project structure under 'EventDay_Version16', including 'Net Environment', 'Generators', and 'DataStores'. Below it, 'KB Explorer' shows a tree view with 'EventDay' containing 'Root Module', 'Components', 'Data', 'DP', 'GeneXus', and 'Panels'. In the center, a properties window for 'EventDay_Version16' shows 'Enable Integrated Security' set to 'True' and 'Integrated Security Level' set to 'Authentication'. Below this, another properties window for 'Company' shows 'Integrated Security Level' set to 'Authentication'. On the right, a 'Properties' window for 'Company' shows various attributes like 'Name', 'Description', 'Module/Folder', 'Business Component', 'Qualified Name', and 'Object Visibility'. A logo for 'GAM GeneXus Access Manager' is also visible in the top right.

Once we have enabled the GAM, we will see another property called **Integrated Security Level** that allows us to indicate the default value of the security of objects in the KB.

This property is also at the object level, so it is possible to customize the way in which security will be implemented in that object.

There are three possible values:

- **None:** indicates that the object will be public, that is, it will not have security.
- **Authentication:** indicates that only authenticated users will be able to execute it.
- **Authorization:** indicates that users, in addition to being authenticated must also be authorized to execute that object, that is, they must have the role appropriate for executing it.



Once our security properties have been set up, the GAM objects will be automatically imported in the KB, and then we will have to do a Rebuild All with that. When we do that, a dialog box will open up telling us that the GAM module will be installed in our KB, with the solution ready for both web and Smart Devices.

GAM is also prepared to execute on a database independent from the application's database if required. In such case, we need not be concerned with this structure because it has a Schema of its own, associated with an Independent Data Store in the KB, making the full configuration independent.

Additionally, GAM will initialize and maintain the whole database fully updated.

DEMO: Integrate GAM into Knowledge Base

Now we will go to GeneXus to use GAM.

GeneXus™

Videos	training.genexus.com
Documentation	wiki.genexus.com
Certifications	training.genexus.com/certifications