Authentication

Nicolas Adrién | GeneXus Training

Web

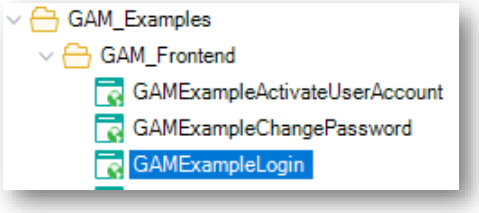
Mobile

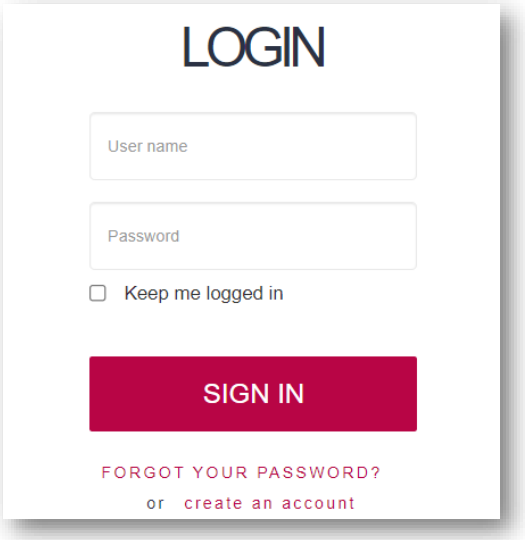
Como dissemos em vídeos anteriores, a autenticação é o ato ou processo de confirmar que algo (ou alguém) é quem diz ser.

Todos os cenários de autenticação GAM incluem a possibilidade de inserir um nome de usuário e uma senha e validar estes dados em uma base de dados existente.

Os modos aos quais pode ser aplicado o GAM dependem do ambiente para o qual são desenvolvidas ou implementadas as aplicações. As opções disponíveis são Web e Móvel.

GAM Login Method

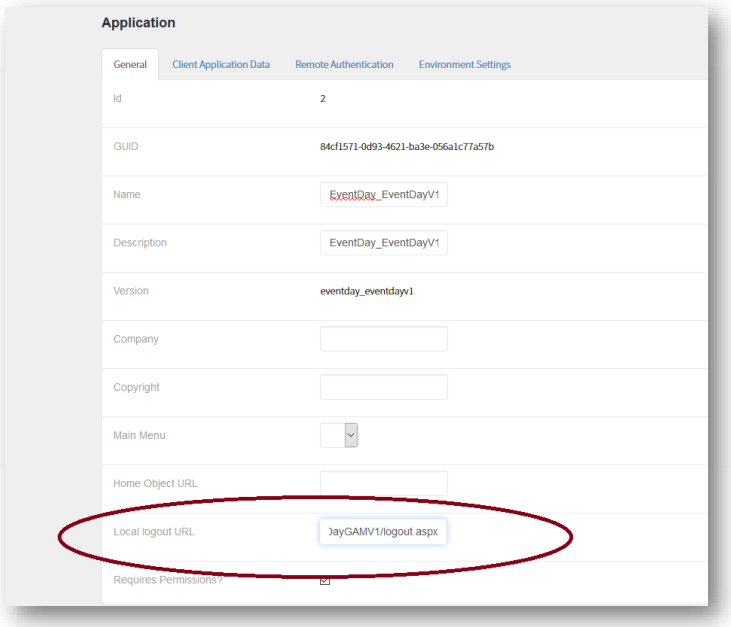




Vejamos o método de Login do GAM.
Como dissemos antes, GAM oferece objetos de exemplo que podemos utilizar como guia.

Em particular, para o método de GAM existe o objeto **GAMExampleLogin**, que realiza a autenticação através do GAM.
Nele é utilizado o método de início de sessão de **GAMRepository**, que é um objeto externo que faz parte da biblioteca GAM, sobre o qual não entraremos em detalhes no momento.

Logout method



Em relação ao encerramento de sessão, uma possível implementação pode ser a seguinte.

A primeira instrução carrega no repositório da base de dados GAM os dados do usuário, e em caso de erros estes são recebidos no SDT *&Errors*.
A segunda instrução transfere o fluxo para o Web Panel **GAMExampleLogin**. Isso é tudo.

O encerramento de sessão da aplicação é configurado utilizando o backoffice web do GAM na configuração da aplicação (ou mediante programação, utilizando a API do GAM).
Sua finalidade é determinar a URL do objeto a ser redirecionado após a execução do Logout nas aplicações SSO.

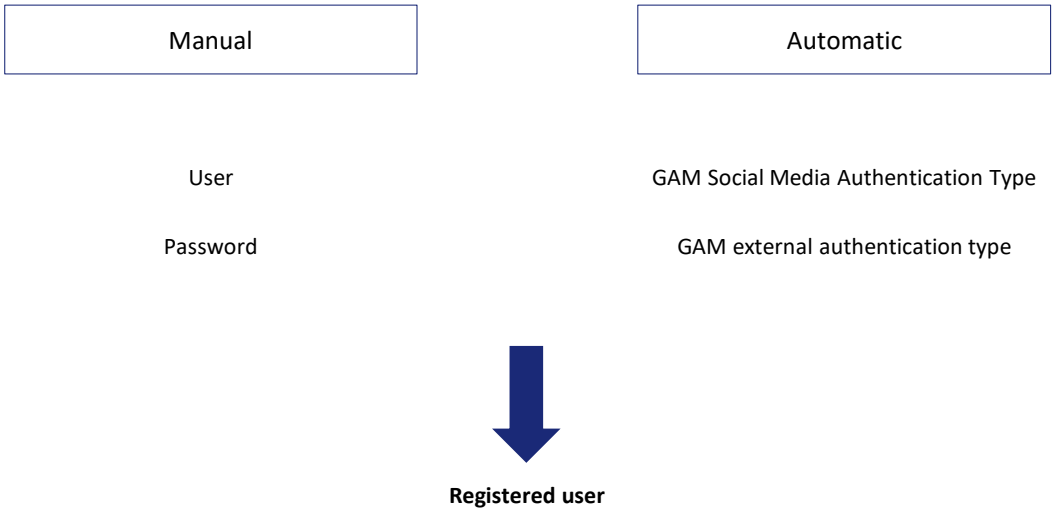
Change Password

```
If &UserPasswordNew = &UserPasswordNewConf
    &isOK = GAMRepository.UpdateUserToChangePassword(&UserPassword, &UserPasswordNew, &Errors)
    If &isOK
        If &Errors.Count > 0
            Msg("Your new password has been saved successfully!.")
            Do 'DisplayMessages'
        Else
            If GAMRepository.IsRemoteAuthentication(&IDP_State)
                //Redirect to remote authentication
                GAMRepository.RedirectToRemoteAuthentication(&IDP_State)
            Else
                &URL = GAMRepository.GetLastErrorsURL()
                If &URL.IsEmpty()
                    GAMHome()
                Else
                    Link(&URL)
                Endif
            Endif
        Endif
    Else
        Do 'DisplayMessages'
    Endif
Else
    Msg("The new password and confirmation do not match.")
Endif
```

Como já mencionamos a existência dos exemplos de GAM, ele também fornece um para a alteração de senha.

O mecanismo dele é bastante simples de entender e consiste em utilizar os métodos fornecidos pelo GAM, como **UpdateUserToChangePassword**.
Não vamos entrar em detalhes sobre a implementação deste método, mas posteriormente à sua chamada só é controlado se houve erros ou não no processo. No caso de que não, também verifica se a autenticação foi remota, com o objetivo de finalizar o processo de Login nas aplicações externas se for o caso, e se não, é redirecionado para home do GAM.

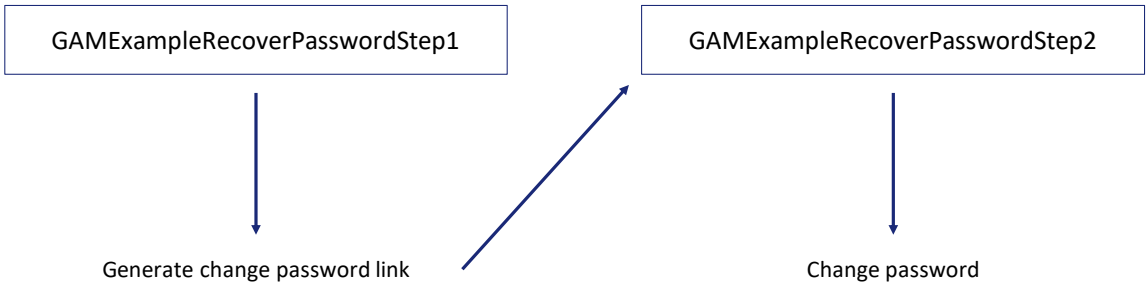
Register User



Em relação ao registro de usuário, existem duas formas de realizá-lo:
A primeira é manual, onde um usuário GeneXus se registra no GAM e ali cria um usuário e senha para ter acesso à aplicação. Isto ocorre no caso de *Tipo de Autenticação GAM Local*.

A segunda forma é automática e é o caso de *Tipo de autenticação do GAM por redes sociais ou autenticação externa*, onde o registro de usuário é realizado quando o usuário entra na aplicação pela primeira vez.

Forgot Password

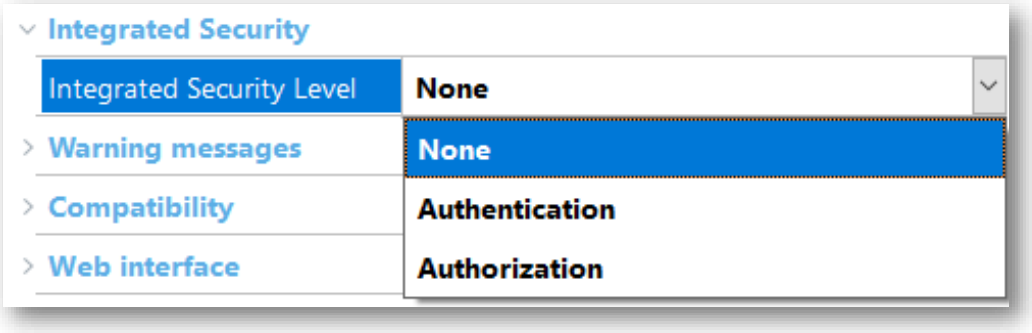


Com a Biblioteca de exemplos GAM são distribuídos dois objetos: **GAMExampleRecoverPasswordStep1** e **GAMExampleRecoverPasswordStep2**, que fornecem uma solução para o caso de esquecimento de senha. A ideia é que o usuário GeneXus complete estas amostras de acordo com suas necessidades.

A ideia básica destes objetos é que, com o primeiro, o usuário da aplicação tenha a possibilidade de inserir seu nome de usuário ou e-mail com o objetivo de alterar sua senha através dele.
Como a ideia é preservar a confidencialidade do usuário, é enviado para seu e-mail um link com destino ao segundo objeto web, pelo qual finalmente poderá alterar a senha.

Na Wiki de GeneXus podem ser conhecidos mais detalhes sobre a implementação disso.

Enforcement



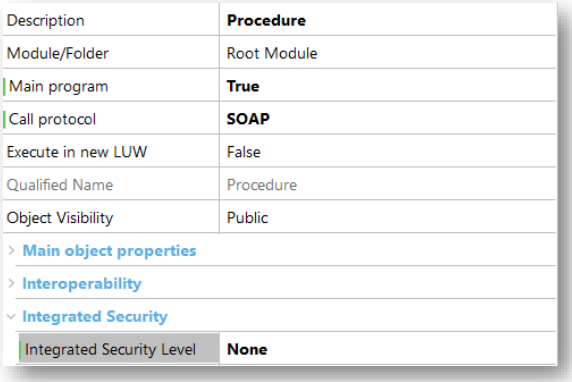
Em todos os webpanels que exigem autenticação, GAM a verifica automaticamente sem necessidade de solicitar ao usuário que faça login o tempo todo.
A forma mais fácil de configurar e definir isto, é configurando a propriedade **Integrated Security Level** com valor **Authentication** nas propriedades dos objetos que queremos que apenas sejam acessados autenticados.

Ao ativar esta propriedade com esse valor se fará cumprir a segurança.
No caso de objetos web, a verificação também é realizada em cada chamada AJAX que é executada. Este é o valor predeterminado no nível da versão.

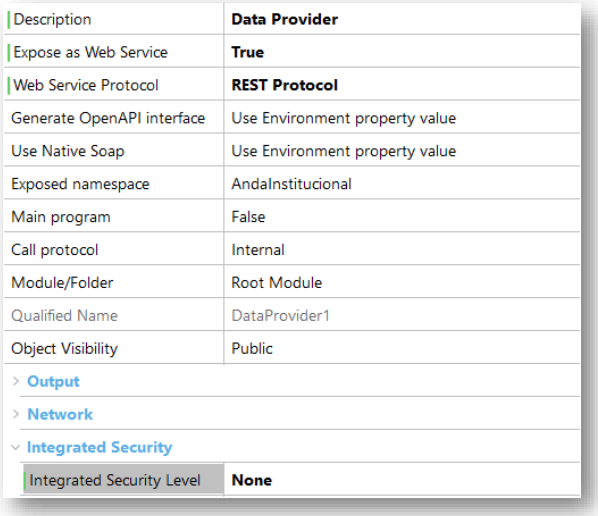
Se o usuário não está autenticado, será exibido um Objeto de início de sessão para a propriedade Web ou um Objeto de início de sessão para a propriedade SD (dependendo da aplicação) para permitir que o usuário se autentique e acesse a aplicação.

No caso de escolher a opção **Authorization**, não só será verificado se o usuário está autenticado no sistema ou não, como também será verificado se possui permissões para acessar o referido objeto.

Enforcement



Description	**Procedure**
Module/Folder	Root Module
Main program	**True**
Call protocol	**SOAP**
Execute in new LUW	False
Qualified Name	Procedure
Object Visibility	Public
Main object properties	
Interoperability	
Integrated Security	
Integrated Security Level	**None**



Description	**Data Provider**
Expose as Web Service	**True**
Web Service Protocol	**REST Protocol**
Generate OpenAPI interface	Use Environment property value
Use Native Soap	Use Environment property value
Exposed namespace	AndaInstitucional
Main program	False
Call protocol	Internal
Module/Folder	Root Module
Qualified Name	DataProvider1
Object Visibility	Public
Output	
Network	
Integrated Security	
Integrated Security Level	**None**

Se a propriedade Integrated Security Level for definida no nível da versão, os procedimentos ou Data Providers do tipo Web Service, assumirão também esse valor e pode ser que não é o que se busca.

Para definir um valor específico ou simplesmente desabilitar a propriedade, é feito como qualquer outro objeto a partir do menu Propriedades sobre o objeto.

Enforcement

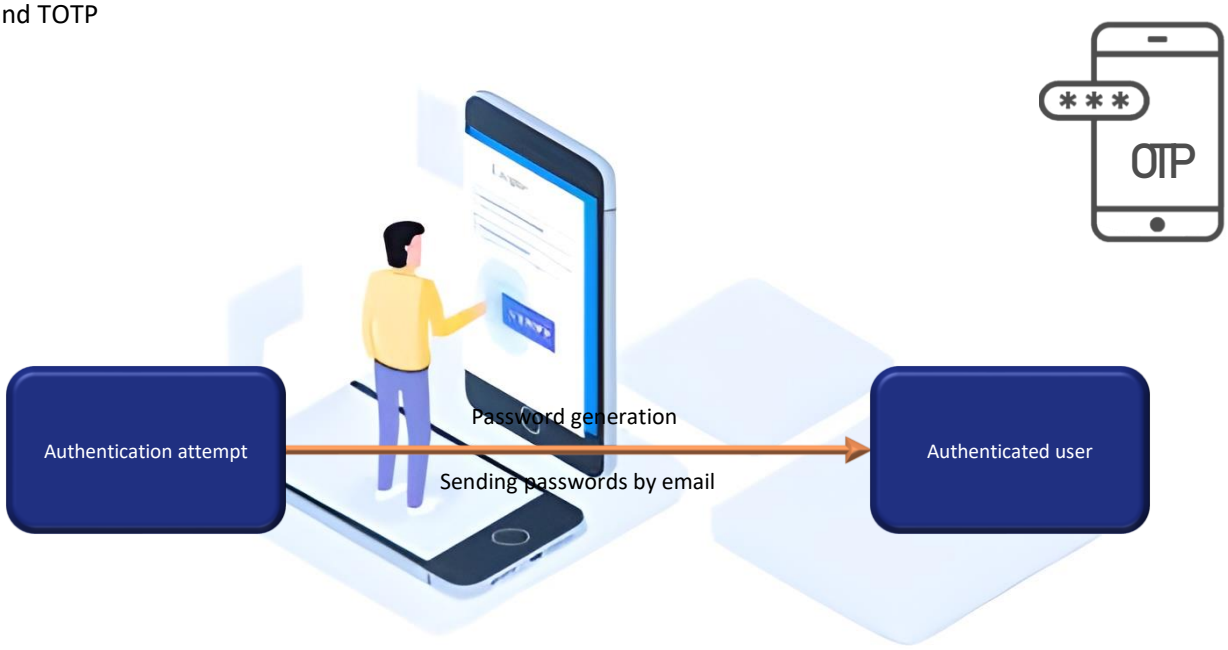
```
&SessionValid = GAMSession.IsValid(&Session, &Errors)
If &SessionValid and not &Session.IsAnonymous
    ...
```

Session	GAMSession, GeneXusSecurity

Outra forma de fazer isto, mas de forma manual, podemos encontrá-la nos objetos de exemplos de GAM, onde é feito com o código que vemos em tela.

Nos diferentes External Objects que traz incorporado GAM, temos **GAMSession**, que possui um método para a verificação de sessão.
Além de verificar o resultado booleano deste método, é possível utilizar uma *variável do tipo GAMSession*, com a qual podemos utilizar os diferentes métodos disponíveis do External Object.
Neste caso em particular, é utilizado **IsAnonymous** para verificar se na sessão está autenticado ou não.

OTP and TOTP



Fornecer acesso seguro a aplicações e software baseado na nuvem é um desafio constante para as empresas de todos os setores.
Uma das maneiras pelas quais tem sido combatido o roubo de senhas e outros tipos de ataques cibernéticos é através do uso de senhas de uso único (ou OTP, como é sua sigla).

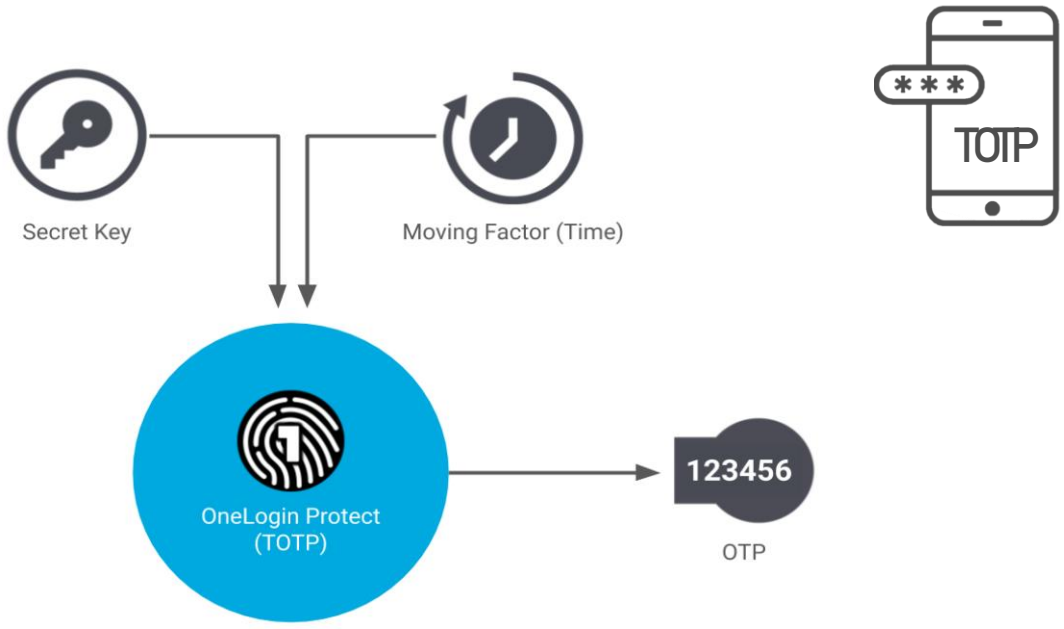
OTP é uma forma de autenticação multifator onde para cada autenticação, é possível gerar uma senha temporária e enviá-la por e-mail ou SMS para o usuário especificado no formulário de início de sessão.

Como funciona?
Quando um usuário se autentica em uma aplicação web, é gerada uma senha e é enviado um e-mail/SMS com a senha gerada. Por padrão, GAM envia senhas por e-mail; no entanto, é possível personalizar como são enviados os códigos aos usuários (por exemplo, enviando as senhas por SMS).
Estas mesmas etapas se aplicam às aplicações móveis.

A senha gerada automaticamente não deve estar expirada e pode ser utilizada para **um único** início de sessão.
No entanto, OTP deve ter uma data de expiração por motivos de segurança, e é o administrador da aplicação que pode definir esse tempo.

Para permitir que um usuário utilize OTP com GAM, o usuário deve existir nele, estando registrado previamente, e deve ter sido validada a autenticidade do método utilizado para receber a OTP, seja o e-mail ou SMS para verificar se realmente é do usuário registrado.

OTP and TOTP



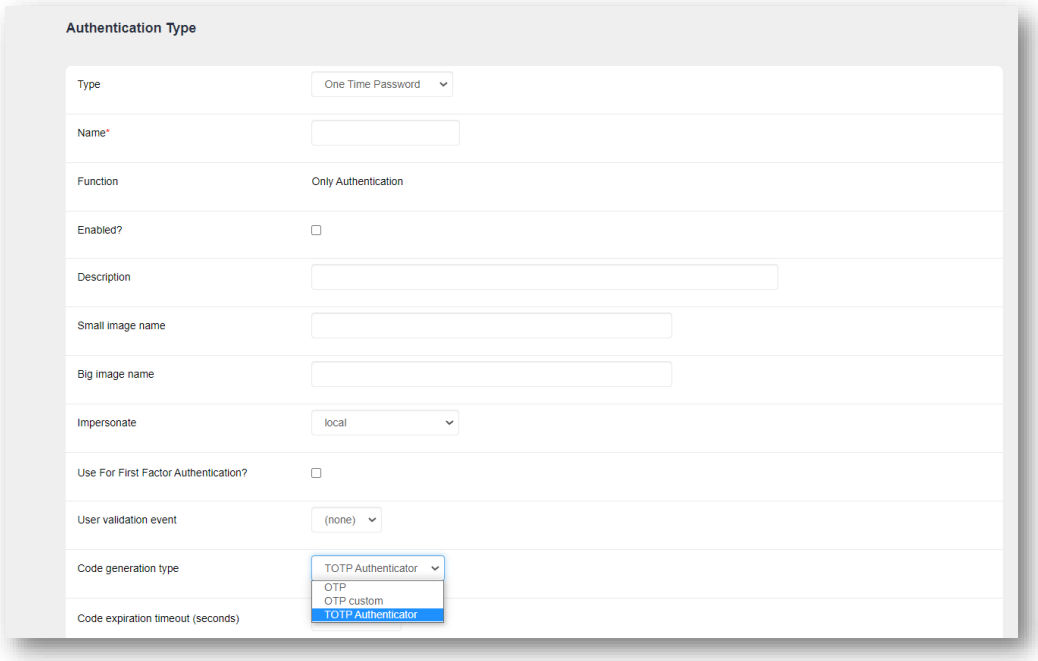
A outra maneira pela qual tem sido combatido o roubo de senhas e outros tipos de ataques cibernéticos é por meio do uso de senhas de uso único baseadas no tempo (ou TOTP, como é sua sigla).

A senha de uso único baseada no tempo é um algoritmo que gera chaves de senha de uso único (OTP) que utilizam a hora atual como fonte de unicidade. Portanto, no GAM, TOTP se encontra como um tipo de geração de Código OTP.

Este método de autenticação oferece a vantagem de que não precisam lembrar de uma senha, pois é gerado um novo código cada vez que desejam iniciar sessão. Além disso, adiciona outro nível de segurança porque o código é válido por um curto período de tempo.

No caso em que alguém tente autenticar-se com um nome de usuário que não lhe pertence, este método adiciona outro nível de dificuldade, pois os usuários precisam de uma aplicação em seu celular para obter estes códigos.

OTP and TOTP



A ativação e configuração destes métodos no GAM pode ser feita através da opção de menu Authentication Type no Backoffice do GAM.

training.genexus.com
wiki.genexus.com
training.genexus.com/certifications