

# Mise en oeuvre de la sécurité des menus sous Forms Builder

par [Souleymane Danem](#) ()

Date de publication : 21 Juillet 2008

Dernière mise à jour :

Comme tous les outils de développement rapide d'applications, Forms Builder offre la possibilité d'activer et / ou de désactiver des éléments de menu en fonction des rôles de l'utilisateur qui se connecte à une application.

L'objectif de cet article est de mettre au point un guide complet et pédagogique pour l'implémentation de la sécurité des menus dans Forms Builder.

Prérequis : niveau 1 en administration oracle, initiation au développement sous Forms Builder

I - Introduction.....	3
II - Prerequis.....	3
III - Configuration.....	3
IV - Création d'une table.....	4
V - Mise en place des rôles et privilèges.....	4
VI - Création des utilisateurs.....	4
VI - Création des formulaires et du menu.....	5
VIII - Association des rôles au menu.....	7
IX - Mise en place de la vue « FRM50_ENABLED_ROLES ».....	9

## I - Introduction

Lorsqu'il s'agit de mettre en oeuvre les notions de sécurité dans les applications, il devient vite fastidieux de gérer les droits d'accès et de modification aux données via le code.

Pour se faire, le noyau Oracle inclu un dispositif appelé : Rôle. Celui-ci permet de définir un profil, de lui associer les droits désirés sur les objets, puis d'associer ce ou ces rôles à un utilisateur.

Oracle Forms permet également d'utiliser ces rôles pour mettre en oeuvre la sécurité au niveau des menus.

Selon le(s) rôle(s) affecté(s) à l'utilisateur connecté (au sens compte Oracle), Forms gère automatiquement l'affichage ou le masquage des options du menu.

## II - Prerequis

On suppose que les utilisateurs de ce tutorial dispose des sources pour installer le noyau Oracle et les outils de développement (Developer/2000, Developer 6i, Developer 9i, #).

Pour l'installation des produits Oracle, voir sur [www.oracle.com](http://www.oracle.com), [www.developpez.com](http://www.developpez.com).

Télécharger le code source mise en oeuvre dans ce tutoriel : [asecmenu.zip](#)

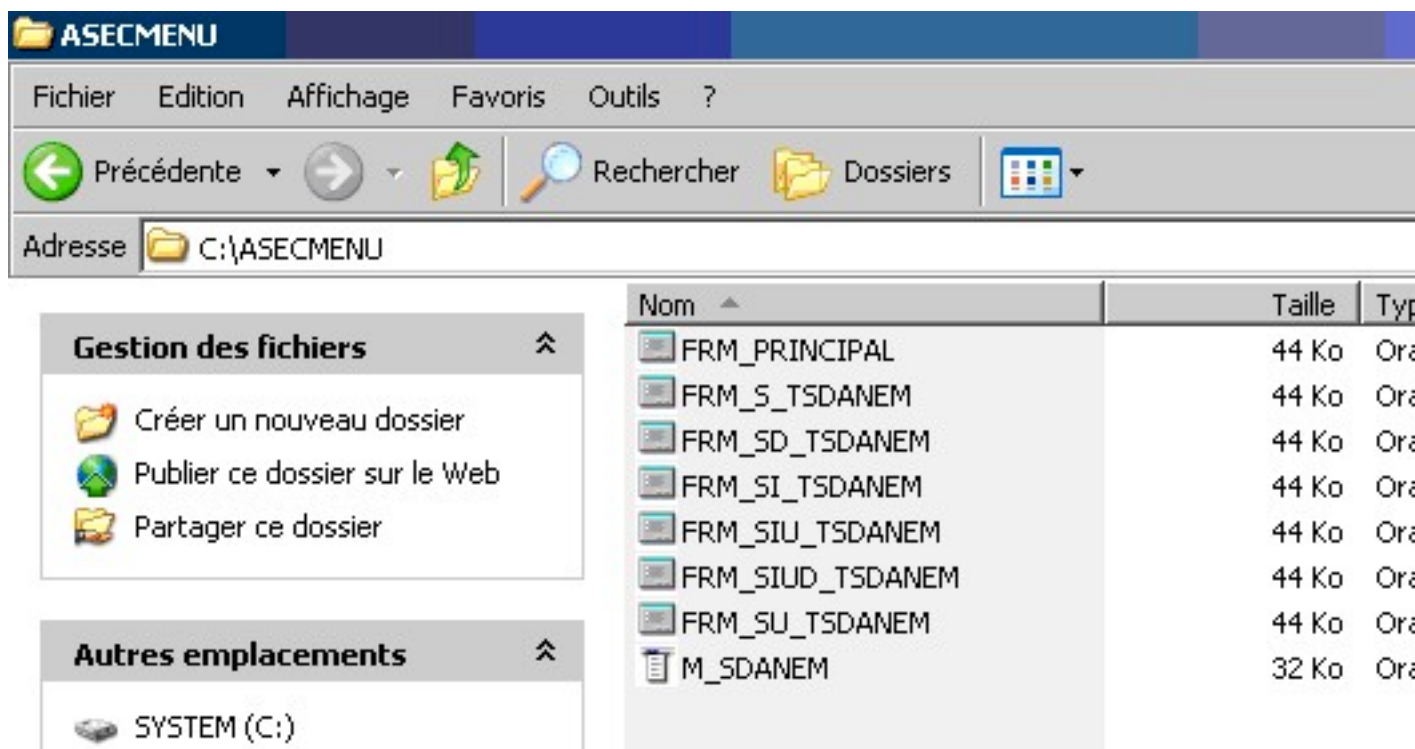
## III - Configuration

Par simplification, les chemins des différents fichiers de l'application ont été " codés en dur ".

Pour les initiés au développement sous Forms Builder, il sera aisé :

- D'ouvrir et de modifier les fichiers \*.fmb et \*.mmb ;
- De paramétrer la base de registre ou les fichiers de configuration nécessaires.

Il faut donc créer le répertoire " C:\ASECMENU " pour y mettre les fichiers.



## IV - Création d'une table

Commençons par créer la table TSDANEM qui sera utilisée pour mettre en place :

- Un menu sur lequel on appliquera la sécurité des menus ;
- Des formulaires qui seront appelés avec des utilisateurs différents à partir du menu.

Soit la table TSDANEM définie comme suit :

```
CREATE TABLE TSDANEM
(code_sd  number primary key,
 nom_sd   varchar2(25) not null,
 pre_sd   varchar2(50) not null,
 age_sd   number(3) constraint ck_age check(age_sd >15),
 mail_sd  varchar2(50)
) ;
```

## V - Mise en place des rôles et privilèges

La seconde étape pour la mise en oeuvre de la sécurité des menus consiste à créer des rôles au niveau de la base de données Oracle. Ces rôles seront affectés aux différents utilisateurs qui seront créés dans l'étape 3.

Ci-dessous le script de création des rôles :

```
CREATE ROLE R_S_TSDANEM ;           -- Permet seulement des SELECT sur TSDANEM
CREATE ROLE R_SI_TSDANEM ;         -- Permet seulement des INSERT sur TSDANEM
CREATE ROLE R_SU_TSDANEM ;         -- Permet seulement des UPDATE sur TSDANEM
CREATE ROLE R_SD_TSDANEM ;         -- Permet seulement des DELETE sur TSDANEM
CREATE ROLE R_SIU_TSDANEM ;
CREATE ROLE R_SIUD_TSDANEM ;
CREATE ROLE R_CON ;
```

Après avoir mis en place les rôles nécessaires, attribuons les privilèges objet (sur la table TSDANEM) aux rôles créés plus haut.

```
GRANT select ON TSDANEM to R_S_TSDANEM ;
GRANT select, insert ON TSDANEM to R_SI_TSDANEM ;
GRANT select, update ON TSDANEM to R_SU_TSDANEM ;
GRANT select, delete ON TSDANEM TO R_SD_TSDANEM ;
GRANT select, insert, update ON TSDANEM to R_SIU_TSDANEM ;
GRANT select, insert, update, delete ON TSDANEM to R_SIUD_TSDANEM ;
GRANT create session to R_CON ;
```

## VI - Création des utilisateurs

A présent, il faut créer les utilisateurs nécessaires. Les utilisateurs, avec des rôles différents, se connecteront à la mini application constituée des formulaires et du menu.

Soit les utilisateurs ci-dessous :

```
CREATE USER usd_s IDENTIFIED BY psd_s ;
CREATE USER usd_si IDENTIFIED BY psd_si ;
```

```
CREATE USER usd_su IDENTIFIED BY psd_su ;
CREATE USER usd_sd IDENTIFIED BY psd_sd ;
CREATE USER usd_sui IDENTIFIED BY psd_sui ;
CREATE USER usd_suid IDENTIFIED BY psd_siud ;
```

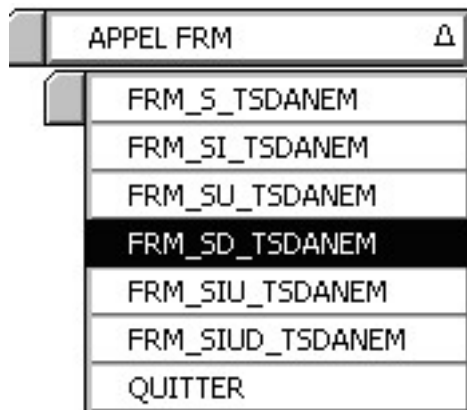
Après avoir créé les utilisateurs, on leur attribue les rôles adéquats :

```
GRANT R_S_TSDANEM to usd_s ;
GRANT R_SI_TSDANEM to usd_si ;
GRANT R_SU_TSDANEM to usd_su ;
GRANT R_SD_TSDANEM to usd_sd ;
GRANT R_SIU_TSDANEM to usd_sui ;
GRANT R_SIUD_TSDANEM to usd_suid ;
GRANT R_CON to usd_s, usd_si, usd_su, usd_sd, usd_sui, usd_suid ;
```

## VI - Création des formulaires et du menu

Après avoir créé la table, mis en place les rôles et créé les utilisateurs sous SQL\*Plus, passons à Forms Builder pour réaliser les formulaires et le menu adéquats.

Soit le menu ci-dessous :



```
Le rôle R_S_TSDANEM est sélectionné pour l'élément de menu FRM_S_TSDANEM
Le rôle R_SI_TSDANEM est sélectionné pour l'élément de menu FRM_SI_TSDANEM
Le rôle R_SU_TSDANEM est sélectionné pour l'élément de menu FRM_SU_TSDANEM
```

Les codes classiques d'appel des formulaires sont associés aux éléments du menu. Pour le cas qui nous concerne, prenons par exemple :

```
OPEN_FORM ('nom_formulaire') ;
EXIT_FORM pour l'élément " Quitter " ;
```

Les formulaires suivants (voir l'écran de la page suivante) seront élaborés :

```
FRM_S_TSDANEM          -- Nom du formulaire pour afficher les données de la table
FRM_SI_TSDANEM        -- Nom du formulaire pour insérer des données dans la table
```

```
FRM_SU_TSDANEM  
FRM_SD_TSDANEM  
FRM_SIU_TSDANEM  
FRM_SIUD_TSDANEM  
FRM_PRINCIPAL
```

Chaque formulaire est basé sur la table TSDANEM.

er - [Navigateur d'objet]

Change Navigateur Programme Outils Window Help

## Forms

PRINCIPAL

TSDANEM

\_TSDANEM

TSDANEM

I\_TSDANEM

ID\_TSDANEM

\_TSDANEM

EM

## Liens attachés

MENU1

## Éléments

APPEL\_FRM\_MENU

## Éléments

FRM\_S\_TSDANEM

FRM\_SI\_TSDANEM

FRM\_SU\_TSDANEM

FRM\_SD\_TSDANEM

FRM\_SIU\_TSDANEM

FRM\_SIUD\_TSDANEM

QUITTER

## Objets

êtres

de programme

de propriété

visuels

PL/SQL

d'objets

grés

base de données

## VIII - Association des rôles au menu

C'est l'une des étapes les plus importantes. Il faut procéder comme suit et de façon minutieuse  
Attribuer tous les rôles créés plus haut au module M\_SDANEM en utilisant la propriété " Rôles module ".  
Cette opération est effectuée avant d'attribuer un accès rôle aux différents éléments du menu (voir l'écran de la page suivante) :

- Il faut double-cliquer sur la propriété " Rôle module " pour ouvrir la fenêtre ;
- Ensuite, il faut saisir les noms de rôles ;

Positionner la propriété de module Menu " Sécurité " à Oui:

Menu : M_SDANEM	
<b>général</b>	
Module	M_SDANEM
Information de référence	
Commentaires	
<b>fonctionnel</b>	
Menu principal	MENU1
Arborescence de menu	
Chemical path de fichier de menu	
Chemical path de démarrage	
Intégrer bibliothèque avec application Forms	Oui
<b>propriété menu</b>	
Sécurité	Oui
Rôles module	

Pour chaque élément de menu, attribuer des accès rôle en utilisant la propriété " Rôle élément "

Positionner la propriété d'élément de menu " Afficher sans droit d'accès " à Non.

Cela permet de rendre invisible un élément de menu en fonction des rôles de l'utilisateur connecté :

Builder - C:\ASECMENU\M\_SDANEM.mmb - [Palette de propriétés]

Propriété Programme Outils Window Help

M\_S\_TSDANEM

	FRM_S_TSDANEM
référence	
	Oui
	FRM_S_TSDANEM
de menu	Simple
enu magique	Aucun
utons associé à l'option de menu	
mande	PL/SQL
nt Menu	
rier	
enu	Oui
barre d'outils de menu horizontale	Non
barre d'outils de menu verticale	Non
enu	Non
d'icône	
u	
	Suite...
droit d'accès	Non

**M\_SDANEM: Options de menu - autorisa...**

Rôles ayant accès à cette option :

- R\_S\_TSDANEM
- R\_SI\_TSDANEM
- R\_SU\_TSDANEM
- R\_SD\_TSDANEM
- R\_SIU\_TSDANEM
- R\_SIUD\_TSDANEM

Compiler (CTRL +T) le module M\_SDANEM pour générer l'exécutable \*.mmx

Associer le module M\_SDANEM à tous les formulaires :

er - C:\ASECMENU\FRM_PRINCIPAL.fmb - [Palette de propriétés]	
ropriété Programme Outils Window Help	
: FRM_PRINCIPAL	
	FRM_PRINCIPAL
ence	
le	
	FRM_PRINCIPAL
de	F_PRINCIP
	Fichier
	<code>c:\asecmenu\m_sdanem</code>
gueur obligatoire	Non
u	

Compiler (CTRL + T) tous les modules Forms créés dans la partie 4

### IX - Mise en place de la vue « FRM50\_ENABLED\_ROLES »

La vue FRM50\_ENABLED\_ROLES doit exister dans la base de données pour que la sécurité des menus puisse fonctionner correctement.

Un administrateur de bases de données (connecté comme SYSTEM) doit exécuter le script FRMxxSEC.SQL (qui se trouve dans le répertoire d'installation de Forms) avec :

- xx = 45 pour Forms 4.5
- xx = 60 pour Forms 6i
- xx = 90 pour Forms 9i
- ...

Ensuite, il faut exécuter la requête ci-dessous (qui n'existe pas toujours dans les scripts du genre FRMxxSEQ.SQL) :

```
GRANT select ON FRM50_ENABLED_ROLES TO public ;
```

ou

```
GRANT select ON FRM50_ENABLED_ROLES TO &nom_user ;
```

Cependant, la seconde requête offre plus de sécurité. Chacune des deux requêtes doit être lancée avec l'utilisateur SYSTEM.