AWOR - Fichier Lisez-moi de l'administrateur

11 mars 2007

1 Compatibilité

Cette application peut-être utilisée sur un serveur fonctionnant avec Apache, MySQL(>=4) et PHP (>=4).

PHP L'application ne nécessite pas de configuration de PHP particulière. Elle fonctionne quelque soit la valeur du paramètre de configuration register_globals, utilise les sessions via les fonctions introduites en PHP4. L'application fonctionne quelque soit la verbosité des affichages d'erreur et de warning PHP, car nous avons travaillé à éliminer tous les messages « E_NOTICE » PHP qui surviennent par exemple lors d'une comparaison avec une variable non-initialisée.

Les scripts PHP n'accèdent pas en écriture au système de fichier, sauf pour l'upload de fichiers. Le dossier « fichiers » doit être accessible en écriture par les scripts PHP sans quoi la fonctionnalité d'échange de fichier serait inutilisable.

MySQL Les scripts PHP ne modifient pas structurellement la base de donnée MySQL, il suffit donc d'un compte permettant les instructions SELECT, INSERT, UPDATE, DELETE sur les tables de l'application. Pour indiquer à l'application quel compte MySQL utiliser, veuillez vous reporter à la section 3.2.

2 Sécurité

Cette application a été développée en gardant toujours à l'esprit les problèmes de sécurités. Nous ne pouvons évidemment pas la garantir failles ni bugs, mais nous sommes fixé des règles de programmation pour éviter les failles les plus courantes. Toutes les données issue de l'utilisateur lors des traitements des pages web sont traitées pour éviter des problèmes d'insertion de code dans les requetes SQL. L'authentification des utilisateurs est systématique, vérifiée à chaque page et repose sur le système des sessions PHP. En revanche, il faut noter que le formulaire d'authentification n'est pas protégé contre l'usurpation de mots de passe car les contraintes données dans le cahier des charges ne le permettent pas, dû moins, il n'est pas possible d'utiliser un cryptage efficace (assymétrique) sans modifier le système d'authentification déléguée et sans utiliser une configuration de serveur Web spécifique. Considérez, pour résoudre ce problème, l'emploi sur serveur web sécurisé (HTTPS) et l'utilisation d'une liaison entre le serveur web et le serveur d'authentification au travers d'un réseau de confiance.

3 Installation

3.1 Paramètres de configuration

Cette application comporte deux fichiers de configuration qui sont dans le dossier « include ». Les fichiers existent dans la version distribuée et ont pour but de servir de modèle.

connect.inc.php : ce fichier contient les coordonnées du serveur MySQL, de l'utilisateur SQL et de la base que l'application utilisera.

config.inc.php : ce fichier contient tout le reste de la configuration de l'application. Le fichier est un script PHP qui initialise un tableau multi-dimentionnel. Ce choix à été fait par commodité, et pour permette de bien hiérarchiser les information, en revanche, la syntaxe est fourbe… N'oubliez pas de virgule ! Toutes les constantes sont décrites ci-dessous.

- **$adminMail :** adresse e-mail de l'administrateur. Affichée à l'utilisateur lorsque qu'une erreur interne à l'application s'est produite.
- **$automatedMail :** adresse e-mail de réponse des courriels envoyés automatiquement par l'application.
- **AUTH :** Paramètres d'authentification
  - **POP :** Paramètres pour les serveurs mail POP
    - **SERVERS :** Chaque élément de ce tableau à pour clef le nom de domaine du fournisseur de service de courriel (partie suivant le @ dans une adresse e-mail) à pour valeur un tableau associatif de la forme suivante
      - **subdomain :** adresse complete du serveur offrant le service POP
      - **port :** numéro de port TCP utilisé pour se connecter au service POP
      - **username_is_full_mail :** Booléen (true ou false) indicant si le nom d'utilisateur à utiliser pour l'authentification est l'adresse e-mail complète. C'est utile pour les serveurs POP gérant des adresses email sur plusieurs domaines (comme wanadoo / orange au moment où j'écris ce document)
  - **bypass_if_local :** Booléen indiquant si l'authentification doit être outrepassée si la connexion s'effectue depuis le serveur lui-même.
- **CSS :** Paramètres des styles de pages

- **CHOOSER_LIST :** Ce tableau associatif contient la liste des styles CSS à utiliser. Les clefs sont les libellés des thèmes à afficher et les valeurs sont les noms des fichiers CSS correspondants, sans chemin, ni extension.

MAIL :

- **TEMPLATES :** Ce tableau contient des tableaux associatifs dont les clefs seront utilisées dans les passages de paramètres HTML, et contiennent des sous-tableaux de la forme suivante :
  - **caption :** Intitulé qui sera affiché à l'utilisateur
  - **tpl_file :** Nom du fichier (avec l'extension PHP) contenant le modèle.

SUPERUSERS : Ce tableau contient la liste des login des super-utilisateurs (ceux qui ont le droit de supprimer des réunions)

UPLOAD : Paramètres concernant l'attachement de fichiers aux réunions

- **accepted_files :** Tableau des extensions qui sont autorisées.

3.2 Initialisation de la base

Tout les fichiers nécessaires sont rangés dans le dossier « install ». Vous utiliserez principalement le fichier « RAZBase.php ». Appelé au travers du serveur web, il vous permettra automatiquement de mettre en place la structure de la base de données, et optionnellement, d'insérer un jeu d'essai pour tester l'application sur votre serveur. Si vous ne souhaitez pas utiliser de compte SQL ayant le droit d'altérer la structure des tables de votre base avec ce script php, vous pouvez exécuter le script « base.sql » avec votre client SQL favori. Le jeu d'essai se trouve dans le fichier « jeu_essai.sql ».

4 Maintenance

4.1 Visualisation des objets et des fichiers

Une page d'administration est prévue à cet effet, et elle n'est accessible que par un « super-utilisateur », c'est à dire un utilisateur consigné dans le fichier de configuration principal en tant que tel, confère 3.1.

4.2 Nettoyage de la base

À partir de la page d'administration décrite dans le paragraphe précédent, vous pouvez aussi purger les fichiers attachés et les objets de la base de données devenus obsolètes. L'aide contextuelle de cette page vous expliquera comment faire. Il est vivement conseillé de faire une sauvegarde de la base de données et des fichiers joints aux réunions avant d'utiliser cet outil car l'erreur de manipulation ou de programmation est toujours possible. . .

5 Authentification dédiée - Personnalisation

AWOR utilise un système d'authentification dédiée de sorte qu'aucun mot de passe ne soit stocké, et que l'utilisateur n'ait pas besoin d'en retenir un nouveau. Le code d'authentification à été écrit de manière à être maléable et est extensible. Le coeur du système d'authentification se trouve dans le fichier « auth_dialog.php ». Vous y trouverez une section commençant par le commentaire ci-dessous.

```
/*************************************************************
** Vous pouvez ajouter d'autres mode d'authentification ici ! **
*************************************************************/
```

Comme l'indique ce commentaire, vous pourrez ajouter à cet endroit une portion de code pour utiliser un autre service d'authentification. La syntaxe est la suivante :

```
case '<nom_auth>' :
/* Eventuellement du code préléminaire */
$auth_is_ok=<fonction_ou_variable_authentification>;
if ( ! $auth_is_ok )
{ // On exécute la ligne ci-dessous que si l'authentificaiton à échouée
$errmsg=<fonction_ou_chaine_explicant_l_erreur_d_authentification>;
}
break;
```

Il vous suffit de remplacer les éléments entre chevrons dans le code ci-dessus et de le coller dans «auth_dialog.inc.php» pour pouvoir utiliser un nouveau mode d'authentification. Reste à donner à l'utilisateur la capacité de l'utiliser. Vous devrez encore, d'une part, ajouter dans la base de données le <nom_auth> à l'énumération du champ «methodeAuth» de la table «Personne», et d'autre part proposer à l'utilisateur d'utiliser ce mode d'authentification en modifiant la page «profil.php».