**Система авторизации.**

Теперь в системе есть группы пользователей. Это повлекло за собой изменение структуры файла конфигурации.

Группа определяется набором команд, которые могут выполнять пользователи из этой группы. Это описывается в конфигурационном файле в тегах <rules>. Зададим группу GUEST (неавторизованный пользователь) и дадим ей разрешение на вывод формы входа:

*<rules>*

*<group name="GUEST">*

*<allow>*

*<command class="SignIn"/>*

*<command class="SignInResult"/>*

*<command class="DefaultCommand"/>*

*<command class="Msg404"/>*

*</allow>*

*</group>*

*</rules>*

В тегах <allow> прописываются все команды, разрешенные данной группе.

Если исполняемая команда отсутствует в данном списке, будет перенаправление на команду Msg404 – вывод ошибки.

Группы могут наследоваться друг от друга.

*<group name="TEST" parent="GUEST">*

*<allow>*

*<command class="AnotherCommand"/>*

*</allow>*

*</group>*

Группа TEST в данном случае имеет право выполнять любую команду, определенную в группе GUEST, и команду *AnotherCommand.*

**Как узнать группу пользователя.**

Все команды можно разделить на 2 вида:

1. **Команда с входным параметром-объектом**

Тогда группа пользователя определяется этим объектом, т.е. набором команд, которые может выполнить этот пользователь с этим объектом.

Это накладывает дополнительные условия на файл конфигурации – поменялись описания командных классов.

*<commands>*

*<command class="ShowMessageGroup" view="ShowMessageGroup" mainObj="MessageGroup">*

*<param name="mgid" required="1" objId="1"/>*

*</command>*

*<command class="SignIn" view="SignIn"/>*

*</commands>*

Здесь команда *ShowMessageGroup* совершает какие-то действие на объектом *MessageGroup*, который получает в виде id под именем параметра mgid.

То есть атрибут описания *mainObj* указывает тип объекта, с которым работает команда, а атрибут *objId* описания параметра команды указывает, что этот параметр – id этого объекта.

1. **Команда без входного параметра-объекта**

Раз команда не нуждается в объекте, то её выполнение зависит только от принадлежности пользователя к глобальной группе и не требует дополнительного описания. Пока таких группы 2: USER (авторизованный пользователь) и GUEST (неавторизованный пользователь).

**Как заставить это работать.**

Нужно, если требуется, дополнить описание своей команды и вписать её в нужную вам группу. Дать права на объект можно следующим образом:

*$acm = AccessManager::instance();*

*$acm->SetRuleObj($user\_id, $obj\_id, $group, $obj\_type);*

*$acm->SetRuleObj(1, 1, “USER”, “MessageGroup”);*

В качестве примера можете протестировать URL:

*http://localhost/message/show/group/2*

*http://localhost/message/show/group/1*

для разных пользователей.

Вход: *http://localhost/SignIn*

Выход: http://localhost/SignOut

Вопросы – в гитхаб