Handling Permissions for Custom User Object Attributes in LDAP or AD

Description

Active Directory schemas often comes with a plethora of customizations to cater to the specific requirements of an organization.

The sample code snippet below can be used in Bean Shell Permission, to retrieve the values of your custom user object attribute, to perform permission logic.

This guide is applicable for all permission types, e.g: Form Permission, Userview Category Permission, etc.

```
import java.util.Map;
import org.joget.directory.model.User;
import org.joget.plugin.ldap.model.UserLDAPImpl;
import javax.naming.directory.Attributes;

public boolean isAuthorized(User user, Map params) {

    /*
    'user' parameter is current user
    'user' parameter is of User object, UserLDAPImpl extends User
    */

    // Check if current user is not anonymous & current user belongs to AD user
    if (user != null && user instanceof UserLDAPImpl) {

        // Cast 'user' object to UserLDAPImpl
        UserLDAPImpl ldapUser = (UserLDAPImpl) user;

        // Re-use method getAttributes() to get user details
        Attributes attrs = ldapUser.getAttributes();

        //Change the attribute name here to suit your requirements
        String attributeName = "cn";
        if (attrs.get(attributeName) != null) {

            //This is how to retrieve attribute values
            System.out.println(attrs.get(attributeName).get().toString());

            /*
            Perform your permission logic for AD users here
            */
        }
    } else if (user != null && !(user instanceof UserLDAPImpl)) {
        /*
        Handle permission logic for non-AD users
        */
    } else {
        return false;
    }
}

//call isAuthorized method with injected variable
return isAuthorized(user, requestParams);
```