



Roles in the e-Commerce Framework

Applying roles to customers and back-end users can be done a number of ways in the e-Commerce Framework. We will examine how to accomplish this for both the front-end and back-end in this article.

I. Customer Roles

Setting roles for customers is done via the back-end. To see the roles screen you must log into Commerce Manager and go to: **Customers → Customer Roles**

Here you can view the default roles as well as create your own custom roles:

A. Creating the Role

A screenshot of the 'Roles' management interface. The top left corner features the Media Chase logo and the tagline 'Change to Your Company'. The main heading is 'Roles' with the subtitle 'Define list of roles for customers'. Below this is a breadcrumb trail: 'Home > Customers > Customer Roles'. A 'Name:' text box is followed by an 'add role' button. A table below lists existing roles: 'Everyone', 'Customer', 'Administrator', and 'Registered'. On the left side, there is a navigation menu with 'Select a View' (showing 'Customers' and 'Customer Roles'), and 'Actions' (showing 'New Customer' and 'Submit Product Feedback').

The Roles List

Figure 1

To create a new role you give it a name in the text box and click the “add role” button. Your new role will be immediately added to the list. That’s it for the roles screen!

B. Assigning the Role

Now we will see how to apply these roles to a customer. We must first navigate to the customer screen by clicking on the customer link in the left menu or using the top menu:

Customers → Customers

Either way, you will be brought to the following screen which displays, in order of date joined, each customer of the store:

Customers
Find customers

[Home](#) > [Customers](#) > Customers

[New Customer](#) | [Export to Excel](#) | [Export to Xml](#)

First Name Last Name
 Email Company

Email	Full Name	Joined
christine@noemail.com	Christine Calamug	Today, 5:29 PM
tim@mediachase.com	Timothy OBrien	Today, 5:28 PM
jandoe@noemail.com	Jane Doe	8/28/2006 4:59:23 PM
anonymous@yourcompany.com		8/16/2006 10:06:59 PM

Page Size:

The Customer List
Figure 2

In the following screen we will take a look at the customer Jane Doe, by clicking on her full name. This will bring us to the “Customer Details” screen:

Customer Details
View customer info

[Home](#) > [Customers](#) > Customer Details

[New Customer](#) | [Edit Customer](#) | [Login as Customer](#)

Login: Jane
Email: jandoe@noemail.com
Status: Account enabled

Customer Addresses
[Add new address](#)

1. Address: [Modify] Jane Doe 123 Street Los Angeles, CA 90046 United States 123-456-7890	2. Address: [Modify] Jane Doe 5554 6th Street Los Angeles, CA 90046 United States 123-456-7890
--	---

Orders

<input type="checkbox"/>	OrderId	Person	Total	Status	Completed
Page Size: <input type="text" value="10"/>					
Perform batch order processing on selected orders: <input type="text" value="[any]"/> <input type="button" value="Run batch"/>					

The Customer Details “view” Screen
Figure 3

Here you can see the address of the customer, orders, account status, login name, and email. What we are interested in is the “Edit Customer” link at the top of the screen. Clicking on it will bring us to the following screen:

Customer Details

Modify customer info

[Home](#) > [Customers](#) > Edit Customer

Basic Information

Login:

Email:

Password:

Roles:

Customer
Administrator
Registered

Discount: ▼

Account Disabled

Attributes

Key:

Value:

Key	Value	Options

Last updated 8/30/2006 5:23:24 PM

Created 8/28/2006 4:59:23 PM

Return to customers list

The Customer Details “edit” screen
Figure 4

Here we can see how Roles are assigned. In Jan Doe’s case, she has been assigned to the Customer and Registered roles. You can assign a role by highlighting the role by clicking on it. To add multiple roles hold down the ctrl button and click.

C. Applying the Role

Now that we have seen how roles are created and assigned we must now apply these roles. There are two ways of applying these rules. One is through the back-end (CommerceManager) and another is through the PublicStore web.config file. We will deal with each of these methods starting with Commerce Manager.

1. Apply Roles in Commerce Manager

For our example we have the following Catalog which includes a “Fashion and Apparel” category. We want to deny anyone who is a customer from seeing this category, which would be highly unusual since you would want to obviously sell your product, but for the sake of this tutorial we will use this category.

Catalog → Site Structure

Name	Size	Modified	Options
..			
Consumer Electronics		8/16/2006 10:09:10 PM	
Customer Support		8/16/2006 10:09:19 PM	
Daily Specials		8/16/2006 10:09:18 PM	
Elite Customer Club		8/31/2006 10:28:36 AM	
Fashion & Apparel		8/16/2006 10:09:15 PM	
Filter by Brands		8/16/2006 10:09:17 PM	
Filter by Price		8/16/2006 10:09:18 PM	
Puppies		8/16/2006 10:09:17 PM	
Software		8/16/2006 10:09:15 PM	

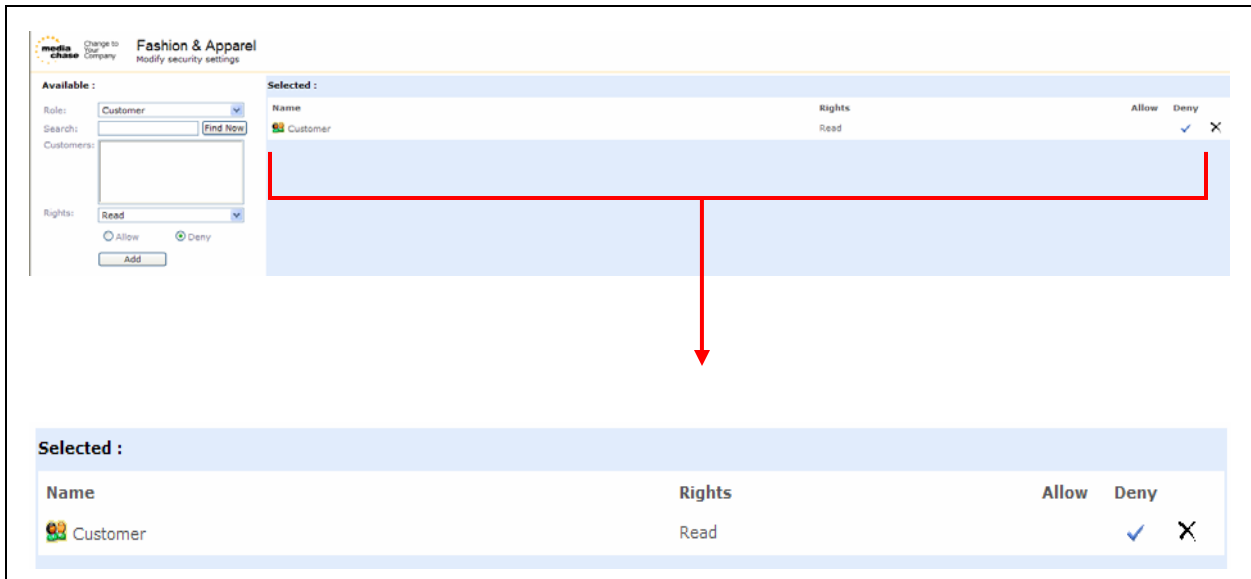


Name
..
Consumer Electronics
Customer Support
Daily Specials
Elite Customer Club
Fashion & Apparel
Filter by Brands
Filter by Price
Puppies
Software

The first step is to apply the “customer” role to the “Fashion and Apparel” category.

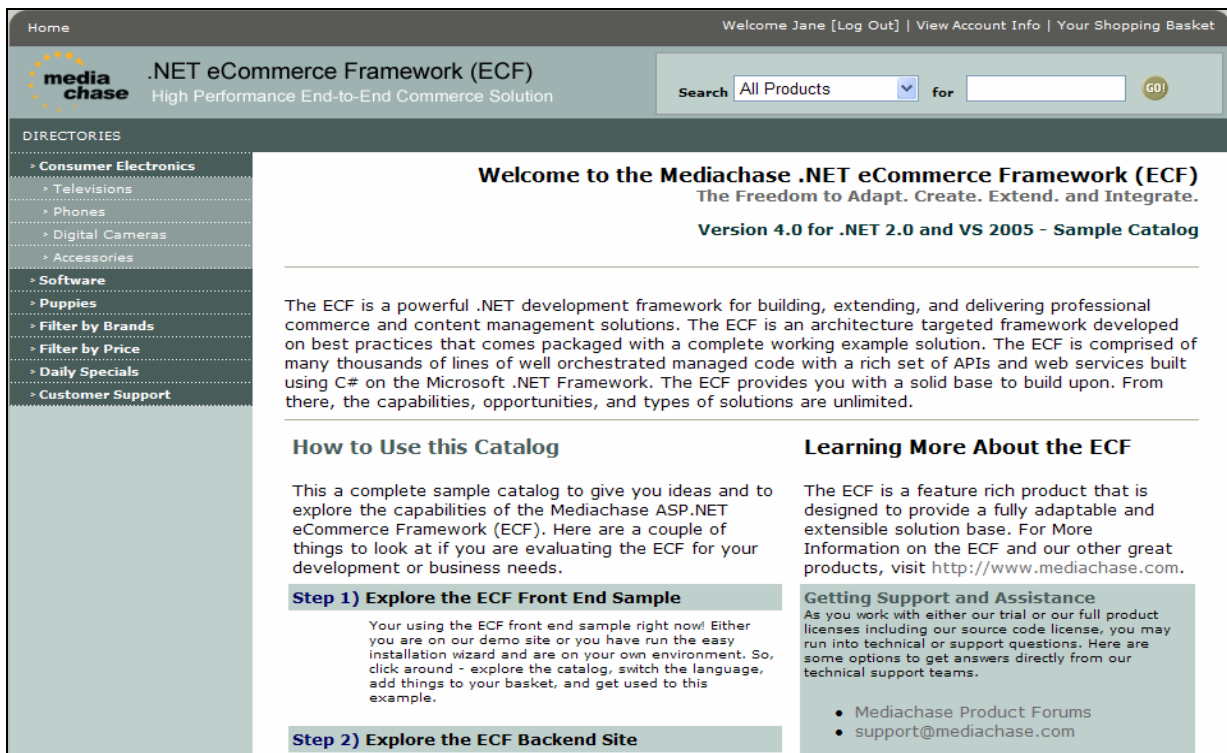
We do this by clicking the category checkbox as seen in figure 5c, then clicking the Access button as seen in figure 5b. This will bring you into the edit security screen for the “Fashion and Apparel” category as seen below.

On this screen you can assign the roles which you want to deny and the roles which you will allow to see this category. In our case, I am going to deny all customers:



Modify Security Settings
Figure 6

Here is our store when we sign in as Jane Doe, who is in the customer role, as you will notice the Fashion and Apparel class is no longer visible due to the fact that Jane Doe is a customer and those with the customer role are denied read rights to the Fashion and Apparel category.



2. Applying the Role in Public Store web.config file

We can also use the Public Store web.config file to apply a role. Below we are disallowing a customer to view their address information, again something you probably would not do, but for the sake of this demonstration we are using it as an example:

```
<location path="Profile/AccountAddress.aspx">
  <system.web>
    <authorization>
      <deny users="?" />
      <deny roles="Customers" />
    </authorization>
  </system.web>
</location>
```

As you can see we are denying anyone who has the role of “Customers” the right to see the AccountAddress.aspx page. You can view your public store web.config file and see the security section where you would apply these settings. Next, we will deal with roles for back-end users.

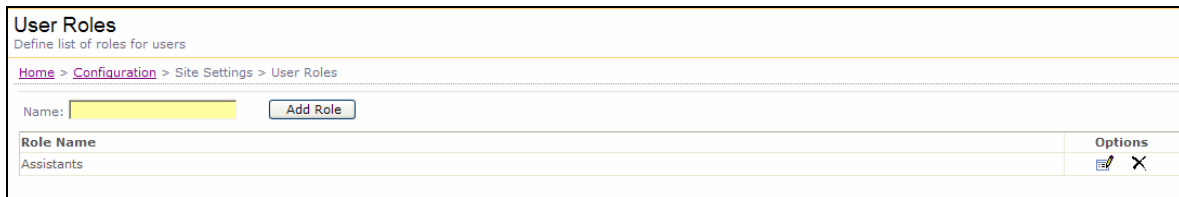
I. User Roles



In the back-end users can apply roles in both the web.sitemap file and also the web.config file. The difference is that in the web.sitemap file defines which menu choices are available to users while the web.config file actually defines rights for certain pages.

Creating the Role

First, we must create the role in the back-end. Navigate to: **Configuration → site settings → user roles**

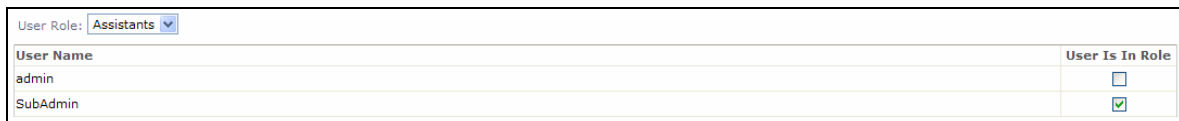
Here you will see the following screen:



Role Name	Options
Assistants	 

Back-end user roles
Figure 8

We have created the role “Assistants” to create this role yourself, just give it a name in the text box and click the “Add Role” button. To apply this role to back-end users click the edit icon under the options area. The screen you will be brought to is the following:



User Name	User Is In Role
admin	<input type="checkbox"/>
SubAdmin	<input checked="" type="checkbox"/>

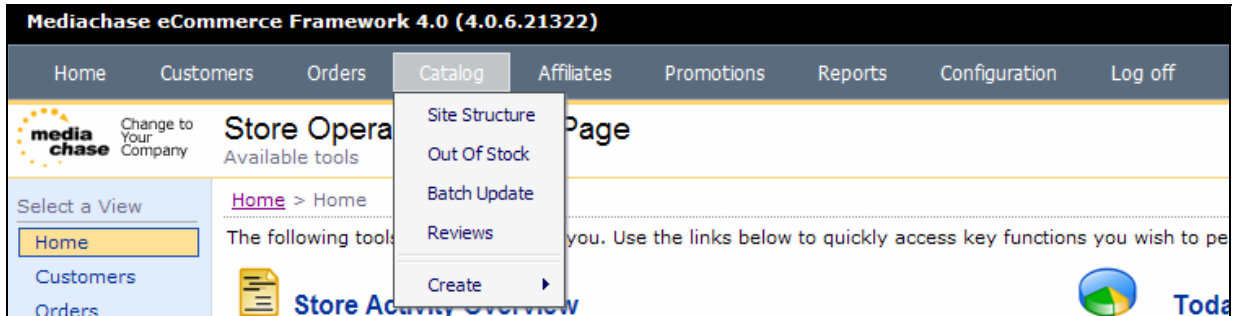
Manager users in a specified role
Figure 9

You can see who is in this role and who isn't, as well as apply or remove the role from a back-end user. Here we have applied the “Assistants” role only to the SubAdmin user. Next we will see how to apply this role within the web.sitemap file.

Apply the Role in web.sitemap

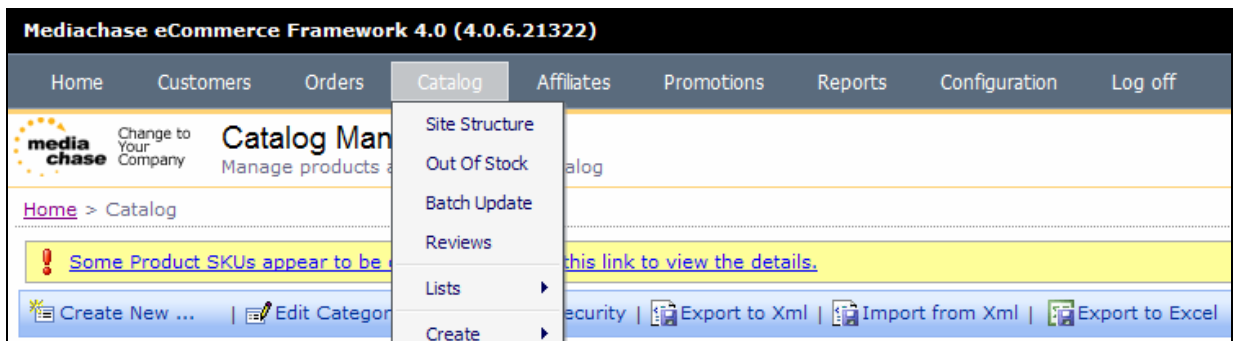
```
<siteMapNode LookId="ContainItemLook" title="$resources:
CommerceManager, SM_LISTS, Lists" roles="Assistants">
  <siteMapNode title="$resources: CommerceManager, SM_ALL_PRODUCTS,
  All Products" url="~/Products.aspx"></siteMapNode>
  <siteMapNode title="$resources: CommerceManager, SM_ALL_SKUS, All
  Skus" url="~/Skus.aspx"></siteMapNode>
  <siteMapNode title="$resources: CommerceManager,
  SM_ALL_VIRTUAL_PAGES, All Virtual Pages"
  url="~/Pages.aspx"></siteMapNode>
  <siteMapNode title="$resources: CommerceManager, SM_ALL_DOWNLOADS,
  All Downloads" url="~/Downloads.aspx"></siteMapNode>
</siteMapNode>
```

In the above code in the sitemap file we have assigned only the Assistants role to the lists node in the menu. This means that this option will now disappear from the menu for all those users who are not “Assistants”. For example, here we have signed in with the “admin” user who is not part of the “Assistants” role. You will notice that “Lists” is not visible in the drop-down, whereas in figure 12, where we are signed in as “subadmin” lists is visible. This is working just as we stated in our web.sitemap.



Signing in with admin user

Figure 10



Signing in with subadmin user

Figure 11

The downfall to using the sitemap file is that it only removes the option from the list, but does not restrict other users from actually navigating to the page if they know the url. For example, the admin user can still get to skus, products, and downloads by simply typing in the absolute url in the address bar.

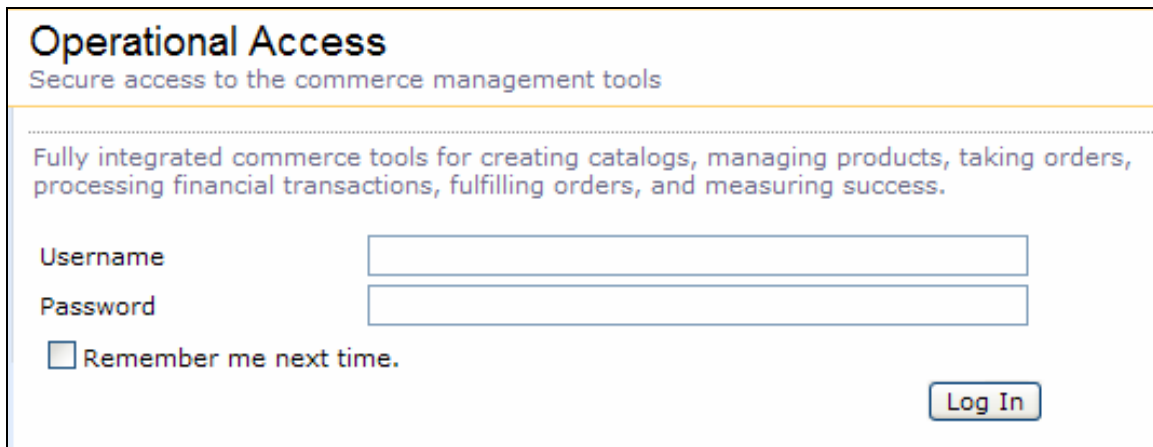
Apply the Role in web.config

As in the public store, one could apply the role in the web.config. This way is better if your aim is to deny rights to the page completely, rather than just “appear” to restrict viewing rights such as with the web.sitemap file.

Below we have applied the roles in two ways we have denied Assistants to view either the Categorynew.aspx page and the Productnew.aspx page. That is, they will be unable to create new products.

```
<location path="CategoryNew.aspx">
  <system.web>
    <authorization>
      <deny roles="Assistants" />
    </authorization>
  </system.web>
</location>
<location path="ProductNew.aspx">
  <system.web>
    <authorization>
      <allow users="*" />
      <deny roles="Assistants" />
    </authorization>
  </system.web>
</location>
```

Two things in particular are worth noting here. The first is although the result looks the same, the item(s) are removed from the menu as was the case with the web.config file, the user is actually prevented from navigating to the page via other means (i.e. absolute url). If the user tries they will be brought to the following screen indicating that only authorized users are allowed:



Operational Access
Secure access to the commerce management tools

Fully integrated commerce tools for creating catalogs, managing products, taking orders, processing financial transactions, fulfilling orders, and measuring success.

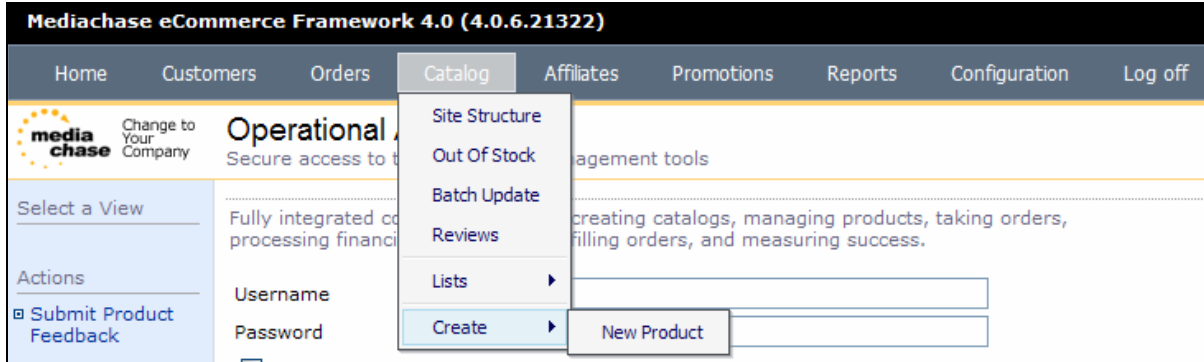
Username

Password

Remember me next time.

Operational Access Screen
Figure 12

Secondly, you will notice that Productsnew.aspx as remained as an option:



**View with roles applied
Figure 13**

This is due to the fact that `<allow users = "*" />` takes precedence over disallowing a individual user.

Conclusion

This concludes our article on roles in both the front-end and back-end of the e-Commerce Framework. We'd love to hear feedback from you on how our articles have helped you or what you think we may improve. You can email me at tim@mediachase.com if you have questions about this article or suggestions for improvement.