

WORKING WITH DBXL PERMISSIONS

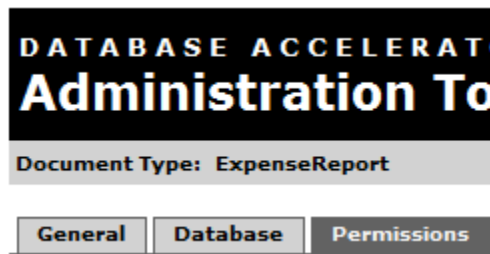
The Expense Report form is storing employee and manager data. We want to keep Expense Report data private, so we have thought up the following permissions schema:

- Everyone will be able to submit a new Expense Report
- Managers will then be able to Read/Write
- The original submitter will only be able to Read after the Expense Report has been submitted.

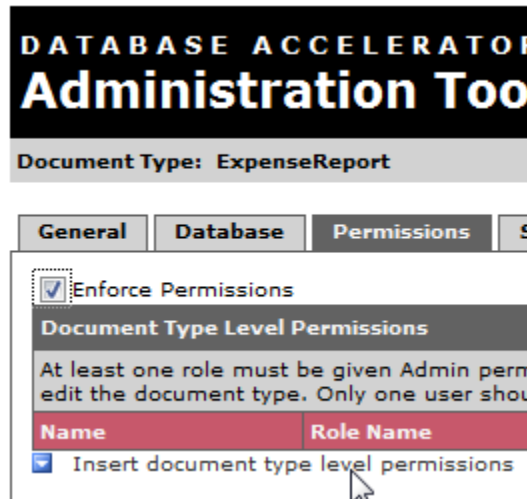
This lab will show you how to set up DBXL permissions to satisfy this requirement.

ALLOW EVERYONE TO SUBMIT A NEW EXPENSE REPORT

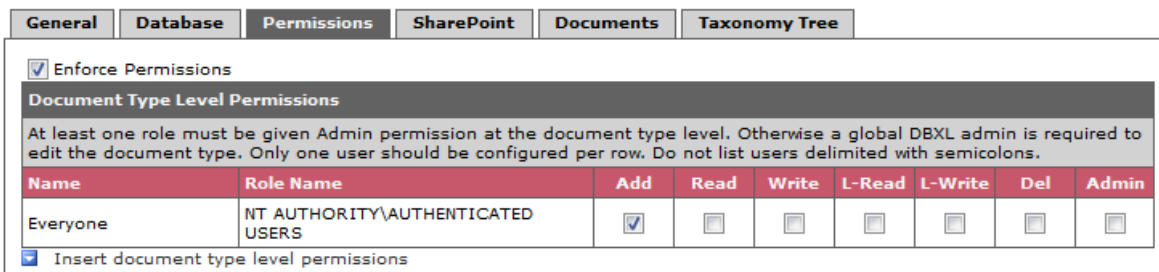
1. Open the DBXL Administration Tool.
2. Click **Edit** for the ExpenseReport document type.
3. Switch to the Permissions tab.



4. Check the **Enforce Permissions** checkbox.
5. Click on **Insert document type level permissions**.



6. Under the **Name** column, enter "Everyone" (without the quotes). This is just a label, so it could be anything you'd like.
7. Under **Role Name**, enter NT AUTHORITY\AUTHENTICATED USERS.
8. Check only the **Add** checkbox. Uncheck the others.



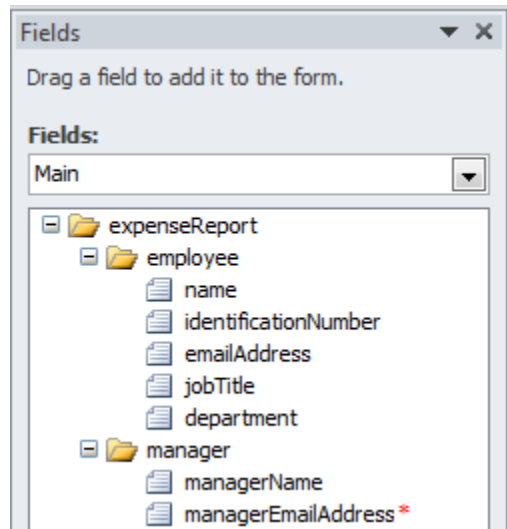
9. Click **Save** and then **OK** in the confirmation dialog.

Congratulations. You have granted all users (that are authenticated on your domain) access to submit a new Expense Report.

MODIFY THE FORM

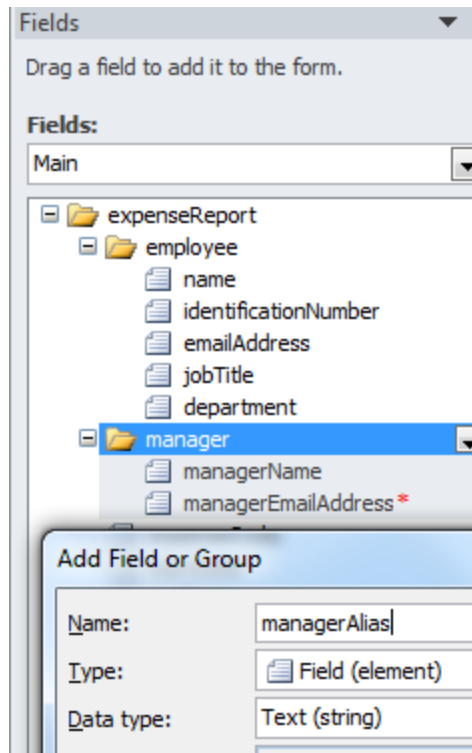
One problem is discovered when we look at our form schema: we are not storing the manager's nor the employee's alias.

Qdabra DBXL Training: Module 6



But it's no problem: we already have an Active Directory data connection, so all we need is a new node and some rules.

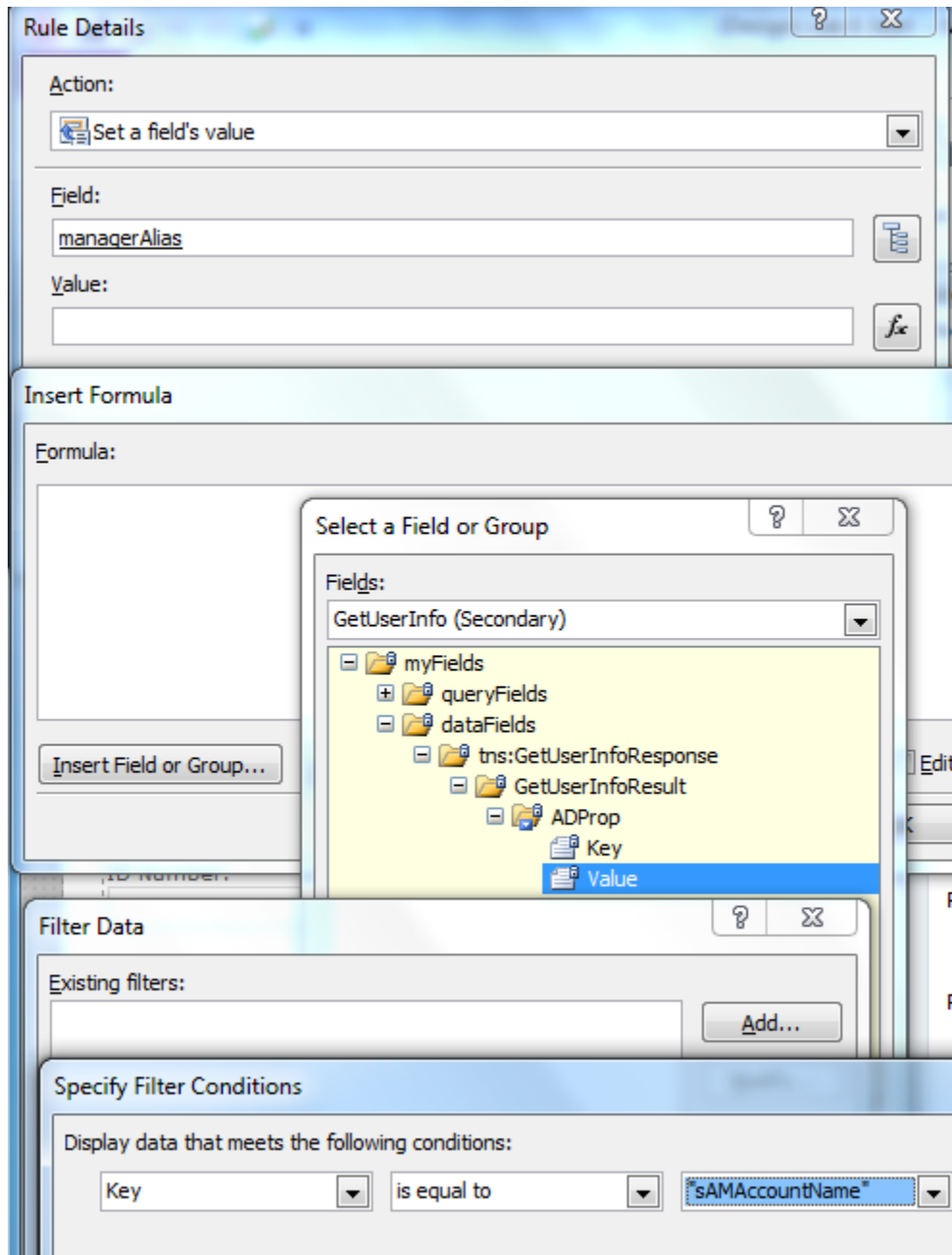
10. Right click on the manager group and select **Add**.
11. Call the new node **managerAlias** and click **OK**.



Qdabra DBXL Training: Module 6

12. Go to **Data > Form Load**.
13. Select the Manager rule.
14. Select **Add > Set a field's value**.
15. For the **Field**, select the new **managerAlias** node.
16. For the **Value**, click **fx**.
17. Click **Insert field or group**.
18. Select the **Value** node from the GetUserInfo data connection, but don't click **OK**. Click **Filter Data** instead.
19. In the **Filter Data** window, click **Add**.
20. In the first dropdown, select **Key**.
21. In the middle dropdown select "is equal too".
22. In the third dropdown, select **Type text** and enter **sAMAccountName**.

Qdabra DBXL Training: Module 6



23. Click OK to close every dialog window.

Are you able to repeat the same steps for the Employee on your own? Try it! If you get stuck, follow these steps:

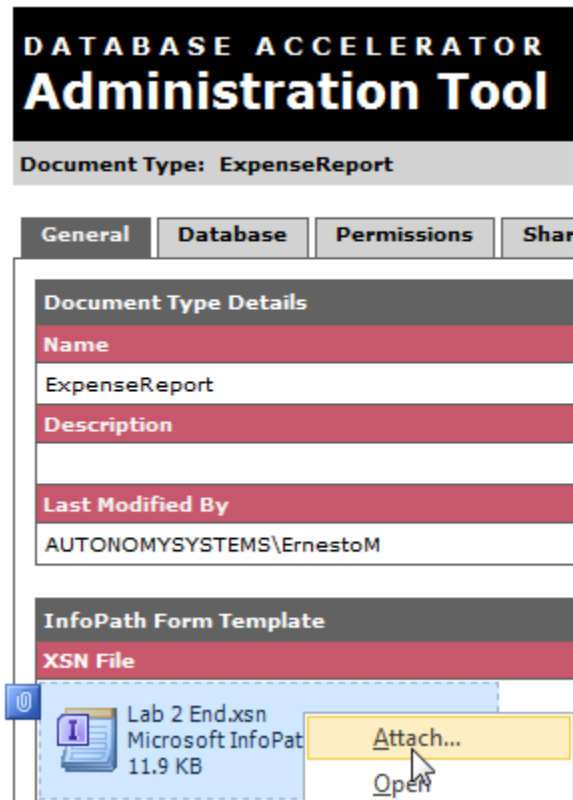
Qdabra DBXL Training: Module 6

24. Right click on the employee group and select **Add**.
25. Call the new node **employeeAlias** and click **OK**.
26. Go to **Data > Form Load**.
27. Select the Employee rule.
28. Select **Add > Set a field's value**.
29. For the **Field**, select the new **employeeAlias** node.
30. For the **Value**, click **fx**.
31. Click **Insert field or group**.
32. Select the **Value** node from the GetMyInfo data connection, but don't click **OK**. Click **Filter Data** instead.
33. In the **Filter Data** window, click **Add**.
34. In the first dropdown, select **Key**.
35. In the middle dropdown select "is equal too".
36. In the third dropdown, select **Type text** and enter **sAMAccountName**.
37. Click OK to close every dialog window.
38. Save the form.

Bonus: If you want to test this change, you can temporarily display the new nodes in the canvas, then preview to make sure that the field is getting populated correctly.

UPDATE THE FORM IN DAT

39. Back in DAT, switch to the **General** tab.
40. Right click on the File Attachment control and select Attach.



41. Attach the latest form template saved a few steps ago.
42. Click **Save** and then click **OK** in the confirmation dialog.

SET PERMISSIONS FOR THE EMPLOYEE

43. In DAT, switch to the **Permissions** tab.
44. Click **Insert document level permissions**.
45. Under **Name**, enter **Employee**. This is just a label, so it can be anything you'd like.
46. Under **Membership XPath**, enter the xpath for the employee field in the form template. It is **/my:expenseReport/my:employee/my:employeeAlias**. You can open the XSN in design mode to double check.
47. Check the **Read** checkbox only.

SET PERMISSIONS FOR THE MANAGER

48. Click **Insert document level permissions**.
49. Under **Name**, enter **Manager**. Again, this is just a label, so it can be anything you'd like.

Qdabra DBXL Training: Module 6

50. Under **Membership XPath**, enter the xpath for the field in the form template. It is ***/my:expenseReport/my:manager/my:managerAlias***. You can open the XSN in design mode to double check.
51. Check the **Writer** checkbox. You'll notice that the **Read** and **L-Write** checkboxes will also be checked.

Document Level Permissions						
Name	Membership XPath	Read	Write	L-Read	L-Write	Del
Employee	/my:expenseReport/my:employee/my:employeeAlias	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manager	/my:expenseReport/my:manager/my:managerAlias	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Insert document level permissions

52. Click **Save**, and then click **OK** in the confirmation dialog.

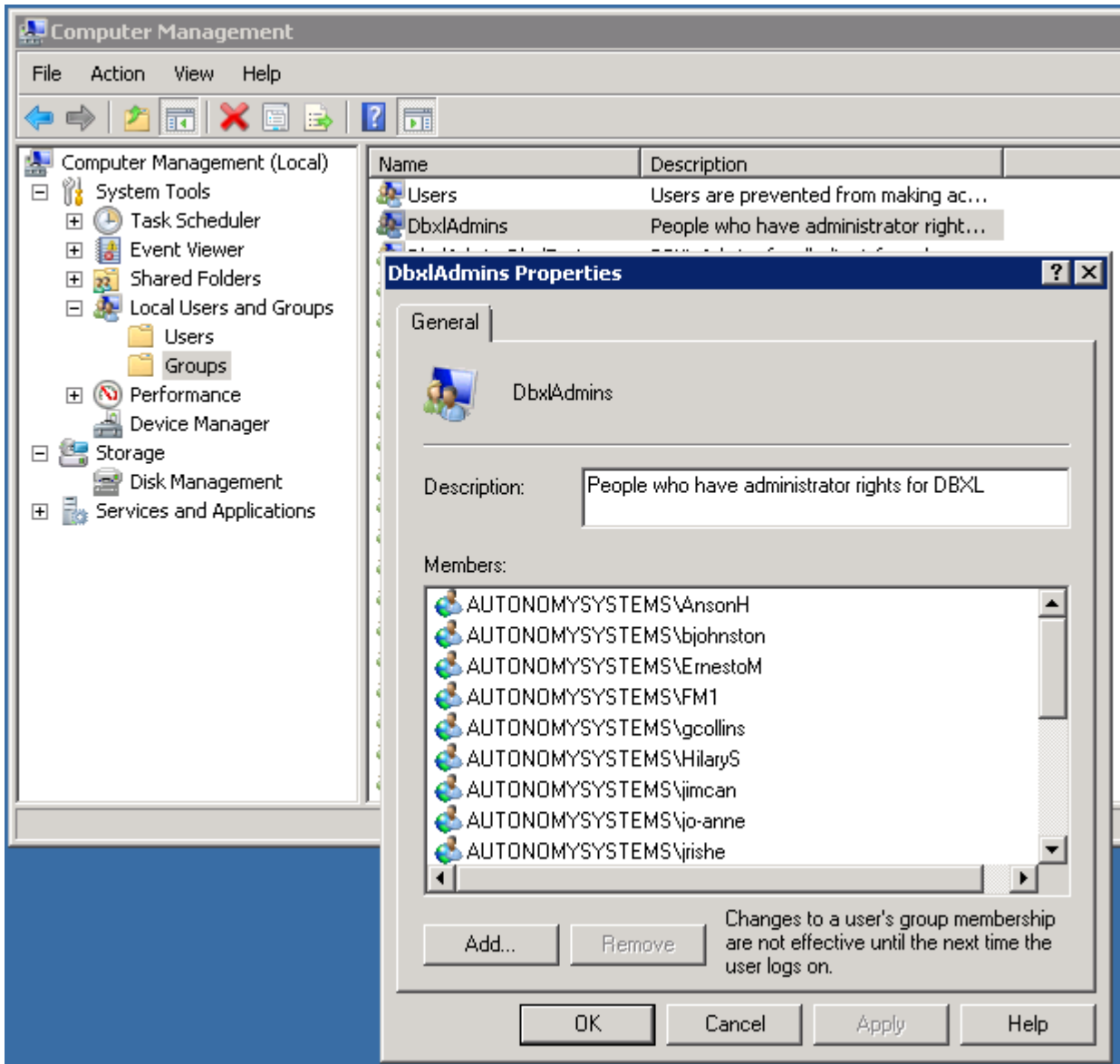
You're done! You've set up the scenario we described at the beginning of this lab.

HOW COULD YOU TEST PERMISSIONS?

If you are executing the steps in this lab, you are most likely a DBXL Administrator. This probably means that you have permissions to all document types and documents. To test the permissions that you have just set up, you will need to:

- a) Reduce your own permissions. To accomplish this:
 - i. On the machine where DBXL is installed, open the DBXL web.config file, located in *C:\inetpub\wwwroot\QdabraWebService* for default installations of DBXL.
 - ii. Locate the following two lines:


```
<add key="AdminGroupAlias" value="<GROUP NAME>" />
          <add key="DbxlAdminAlias" value="<GROUP NAME>" />
```
 - iii. Now that you know the group that is granted DBXL Administrator permissions, open **Computer Management** and locate that group.
 - iv. Remove yourself from the group.



b) Access from another user's account. However, ensure that the user is not a DBXL Administrator.

TEST

1. Open the form by using the **Open** link in the **General** tab.
2. Fill out the form. As you fill out the form, think about which users will be granted Reader and Writer permissions.
3. Reopen the form as the **Reader**, and then try to **Submit** the form, you will see the following:

Qdabra DBXL Training: Module 6

The error message says:

***InfoPath cannot submit the form.
An error occurred while the form was being submitted.
The SOAP response indicates that an error occurred on the server:
System.Web.Services.Protocols.SoapException: Server was unable to process request. --->
System.Exception: Exception Code: I0033
Exception Message: Access denied
at Qdabra.Dbxl.DbxlDocumentServiceEx.CheckAndThrowException(StatusInfo info)
at
Qdabra.Dbxl.DbxlDocumentServiceEx.Qdabra.Dbxl.Services.IDbxlDocumentServiceEx.SubmitDocu
ment(String docTypeName, String xml, String name, String author, String description, Int32&
docId, String& refId)
--- End of inner exception stack trace ---***

For further testing, try to open the document as a user who is neither the reader nor the writer.

APPENDIX

UNDERSTANDING DBXL PERMISSIONS

DBXL ADMIN PERMISSIONS

DBXL Admin Permissions are initially defined when DBXL is installed, though you can modify them later by editing the web.config file on the server, located in *C:\inetpub\wwwroot\QdabraWebService* for default installations of DBXL.

The main purpose of DbxlAdmin is to perform broadly-scoped tasks like:

- adding and removing licenses
- adding, modifying and deleting document types
- reshred documents
- Import/Export documents

However, no additional document permissions are granted (Read/Write/Delete) to this user. That is, the DbxlAdmin doesn't have access to some of the finer-grained operations like performing actions on individual documents.

GLOBAL LEVEL PERMISSIONS

Global Level Permissions are also initially defined during installation. The groups or users who are set at the Global Level will have access to all the document type configurations and all the documents. There are three sets of permissions that can be set at the Global Level:

- **Admin:** The members of this group have access to manage the Document Types as well as all document permissions for the document types they manage.
- **Writer:** The members of this group can view and edit all documents, but they do not have access to view or edit the Document Type Configurations.
- **Reader:** The members of this group are only allowed to view all documents. They do not have access to edit the documents and also do not have access to view or edit the Document Type Configurations.

MODIFYING GLOBAL LEVEL PERMISSIONS

Global Level Permissions can be modified by editing the value for the **ReaderGroupAlias**, **WriterGroupAlias** and **AdminGroupAlias** keys in web.config, located in *C:\inetpub\wwwroot\QdabraWebService* for default

installations of DBXL. You can turn off Global Level Permissions by switching the value of the **CheckUserPermissions** key from *true* to *false*. The **BrowserGroupAlias** only allows opening the XSNs.

```
<!-- Global security settings -->
<add key="CheckUserPermissions" value="true" />
<!--
If CheckUserPermissions is false, the group aliases below
are used as defaults for new document templates if
no permissions are specified
-->
<add key="BrowserGroupAlias" value="NT Authority\Authenticated Users" />
<add key="ReaderGroupAlias" value="" />
<add key="WriterGroupAlias" value="" />
<add key="AdminGroupAlias" value="BUILTIN\Administrators" />
<add key="DbxlAdminAlias" value="BUILTIN\Administrators" />
```

DOCUMENT TYPE LEVEL PERMISSIONS

Document Type level permissions are granted in the Permissions tab of DAT **for each Document Type**.

Note that the **Enforce Permissions** checkbox must be checked for the permissions in this tab to be enforced. You are able to enter as many users or groups as desired, by clicking the **Insert document type level permissions** link. This will insert a new row where you will need to enter a name (an identifier of your choosing) and the user or group (Role Name). Then, for each of these rows, you will be able to grant the following permissions: **Add, Read, Write, L-Read, L-Write, Del** and **Admin**.

In the example below, the managers group has Admin permissions over the Document Type, while the Readers group only has Read access.

Note the following:

- Checking the Admin checkbox will check the other six checkboxes.
- Checking the Write checkbox will check the Read and L-Write checkboxes.
- L-Read Permissions are not currently in use and are reserved for future use.
- Only one user or group can be added per row. Do not insert lists separated by commas, semicolons or other delimiters.

DOCUMENT LEVEL PERMISSIONS

Document Level permissions are also established in the Permissions tab of DAT. As with Document Type Level Permissions, the **Enforce Permissions** checkbox must be checked for the permissions in this tab to be enforced. You are able to enter as many rows as desired, by clicking **Insert document level permissions**. Clicking this will insert a new row where you will need to enter a name (any identifier of your choosing) and

Qdabra DBXL Training: Module 6

the xpath where the xml document stores the role information. Then, for each of these rows, you will be able to grant the following permissions: **Read, Write, L-Read, L-Write** and **Del**.

An example is shown below. In the form, a field will contain the user alias of the **Approver**, who will in turn be granted permission to read, write, lock and delete. Meanwhile, the **Auditor** will only be granted read access. Of course, this assumes that you have designed your form such that it contains these two pieces of information. This allows for fine grained control, as well as dynamic permissions, for instance, allowing the Approver to change the Auditor responsible for the document.

Note the following:

- Checking the Write checkbox will check the Read and L-Write checkboxes.
- L-Read Permissions are not currently in use and are reserved for future use.
- Only one user or group can be added per row. Do not insert lists separated by commas, semicolons or other delimiters.
- If you modify Document Level Permissions after creating documents, you must reshred the documents in order to apply the new permissions to the previously-existing documents.