

SailPoint

IdentityIQ – Lo4

Date

—

Contents

Exercise 1.....	3
Enabling Lifecycle Manager	3
Overview.....	3
Installation of Lifecycle manager	3
Exercise 2	5
Turn on Group Provisioning and Create New Group in LDAP.....	5
Overview.....	5
Turn on Group Provisioning Feature of IdentityIQ.....	6
Verify the Existing LDAP Groups.....	6
Provision a new Group in LDAP called VPN.....	7
Exercise 3	9
Provision VPN Access Using Lifecycle Manager.....	9
Overview.....	9
Enable Business Process (Workflow) Tracing.....	11
Login as a Manager and Request VPN Access for employees.....	12
Exercise 4	16
Create and Manage Identities in IdentityIQ	16
Overview.....	16
Create an Identity using LCM.....	17
Define Provisioning Policies for Creating Identities.....	18
Exercise 5	28
Account Management with Lifecycle Manager	28
Overview.....	28
Configure Lifecycle Manager to support account requests	29
Request a New LDAP Account for our New User Fred.Smith	30
Request a New PRISM Account for Fred.Smith.....	31

Exercise 1

Enabling Lifecycle Manager

Use Case ID:	L04 – E01		
Use Case Name:	Enabling Life Cycle Manager		
Created By:		Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Admin, IIQ System		
Description:	In this exercise, we will enable Lifecycle Manager functionality.		
Preconditions:	IIQ System is Up and Running, LCM setup		
Post conditions:	LCM configuration		
Normal Flow:	1. Importing init-lcm.xml file 2. LCM configuration		
Exceptions:	NA		
Dependent Usecase:	NA		
Assumptions:	NA		
Notes and Issues:	NA		

Overview:

In this usecase, we are going to setup the following:

- LCM configuration

Lifecycle manager is installable as a separate component of IdentityIQ. In order to install and setup Lifecycle Manager, you must stop your application server, install Lifecycle Manager and restart your application server.

Installation of Lifecycle manager

1. Stop the Tomcat server using the **Stop Tomcat** shortcut
2. Launch the IIQ console using the **IIQ Console** shortcut
3. Install Lifecycle Manager by typing the following into the IIQ console:

```
> import init-lcm.xml
```

4. Start the Tomcat server using the **Start Tomcat** shortcut
5. Login to IdentityIQ as **spadmin/admin** and confirm that Lifecycle Manager is installed:
Look for the sections **Manage Access** and **Manage Identity**

Exercise 2

Turn on Group Provisioning and Create New Group in LDAP

Use Case ID:	L04 – E02		
Use Case Name:	Group based provisioning		
Created By:		Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Admin, IIQ System		
Description:	Turn on the IdentityIQ Group Provisioning feature and use IdentityIQ to provision groups to LDAP.		
Preconditions:	IIQ System is Up and Running, LCM setup		
Post conditions:	LCM configuration, Provisioning setup		
Normal Flow:	<ol style="list-style-type: none">1. Groups in IdentityIQ2. Provisioning configurations		
Exceptions:	NA		
Dependent Usecase:	NA		
Assumptions:	NA		
Notes and Issues:	NA		

Overview:

In this usecase, we are going to setup the following:

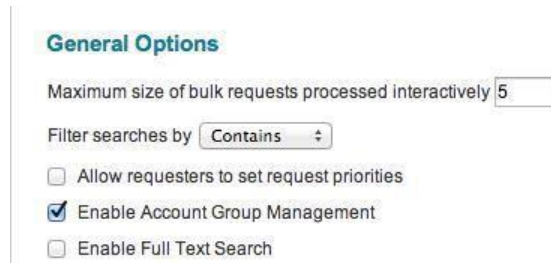
- Identify groups in IdentityIQ
- Provisioning using groups
- LDAP group access provisioning

Out of the box, IdentityIQ can support provisioning Groups to target applications that support it. In this exercise, we will use IdentityIQ to provision a group into LDAP. Once this group is created, we will be able to add additional users to it.

Note: You do not need to use group provisioning within your IdentityIQ implementation. It is also perfectly normal to create, edit and delete groups directly in the native target application.

Turn on Group Provisioning Feature of IdentityIQ

1. Navigate to **Global Setting** ➤ **Lifecycle Manager**
 - a. **Configuration**
2. On the **Additional Options** tab, confirm that **Enable Account Group Management** is selected:



General Options

Maximum size of bulk requests processed interactively

Filter searches by

☐ Allow requesters to set request priorities

☒ Enable Account Group Management

☐ Enable Full Text Search

3. Click **Save**

Verify the Existing LDAP Groups

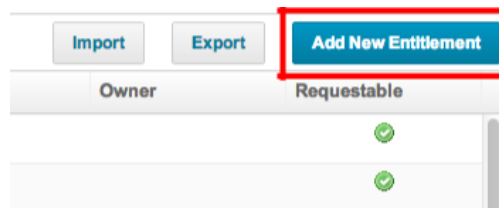
1. View the existing LDAP groups.
 - a. Use the desktop shortcut to launch the **OpenDS LDAP Control Panel**
 - i. Only click it once... it will take a few seconds to start.
 - ii. If necessary, **Start LDAP**
 - iii. Login with the OpenDS Admin Password: **password**
 - b. Select **Manage Entries**, and expand **groups**



d. List the existing groups:

Provision a new Group in LDAP called VPN

1. In IdentityIQ, navigate to the **Application** **Entitlement Catalog**
2. Click **Add New Entitlement** to create a new group.



3. On the **Standard Properties** tab, configure the new group as:
 - a. Application: **LDAP**
 - b. Display Value: **VPN**
 - c. Requestable: **checked**
 - d. Description: **This group controls access to the corporate VPN.**
 - e. Owner: **Randy.Knight**

*Indicates a required field.

Application * LDAP

Type * Group

Attribute * groups

Value * The value for this group will correspond to the dn attribute from the group schema.

Display Value VPN

Requestable ☒

B I U **English (United States)**

This group controls access to the corporate VPN.

4. On the **Group Properties** tab, configure the following:
5. DN: **cn=VPN,ou=groups,dc=training,dc=sailpoint,dc=com**
6. Description: **This group controls access to the corporate VPN.**
7. CN: **VPN**

Standard Properties | **Group Properties**

Group Attributes

DN*

Description

CN*

8. Click **Save**
9. If you configured everything successfully you should see the following:
10. A message that says a workflow was started to create the VPN group. This workflow comes out of the box, but could be customized if so desired. The workflow is called **Entitlement Update**.

✓ A workflow to update the Group named VPN was successfully launched.

11. Under **Define** **Entitlement Catalog** you should see the new entry for **VPN**. Note that the new LDAP group has a Description, Owner and is Requestable

Entitlement Catalog

✓ A workflow to update the Group named VPN was successfully launched.

LDAP

Application	Attribute	Display Name	Type	Description	Owner	Requestable
LDAP	groups	VPN	Group	This group controls access to the corporate VPN.	Randy Knight	✓

12. Check LDAP to see that the group was created.
13. On the manage entries page, drill down to the groups and confirm that your **VPN** group is created:



Exercise 3

Provision VPN Access Using Lifecycle Manager

Use Case ID:	L04 – E03		
Use Case Name:	VPN access provisioning		
Created By:		Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Admin, IIQ System		
Description:	The objective of this exercise is to allow managers to request VPN access for their users via Lifecycle Manager.		
Preconditions:	IIQ System is Up and Running, LCM setup		
Post conditions:	LCM configuration, Provisioning setup		
Normal Flow:	<ol style="list-style-type: none">1. VPN access setup2. Provisioning to VPN group using IdentityIQ		
Exceptions:	NA		
Dependent Usecase:	NA		
Assumptions:	NA		
Notes and Issues:	NA		

Overview:

In this usecase, we are going to setup the following:

- Identify VPN specific groups in IdentityIQ
- Provisioning to VPN groups

We just created a group in LDAP called VPN, and we made this account group request able, meaning that users can request it through LCM.

Next, we will login as a manager (Catherine.Simmons) and request VPN Access for all of the direct reports in her department.

This will kick off a workflow case for each user with appropriate approval steps and will eventually (assuming all approvals are affirmative) result in a provisioning of the entitlement in LDAP.

The default workflow for entitlement requests is called **LCM Provisioning**. Each Lifecycle Manager operation has a default workflow (Business Process) defined as seen here. Out of the box, the default workflows are:

Lifecycle Manager Configuration

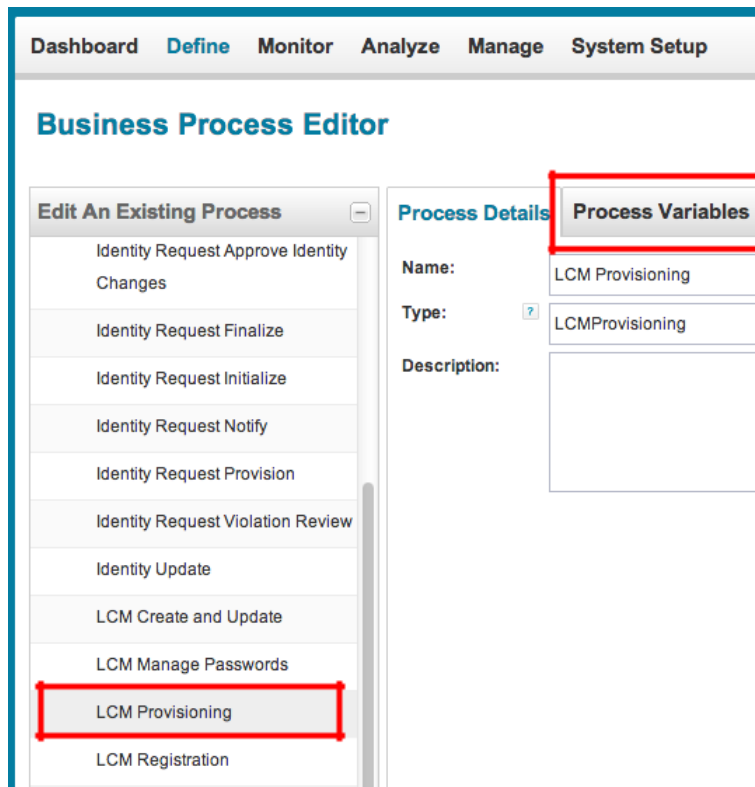
Lifecycle ActionsBusiness ProcessesAdditional Options

Action	Business Process
Request Access	LCM Provisioning
Manage Accounts	LCM Provisioning
Manage Passwords	LCM Manage Passwords
Edit Identity	LCM Create and Update
Create Identity	LCM Create and Update

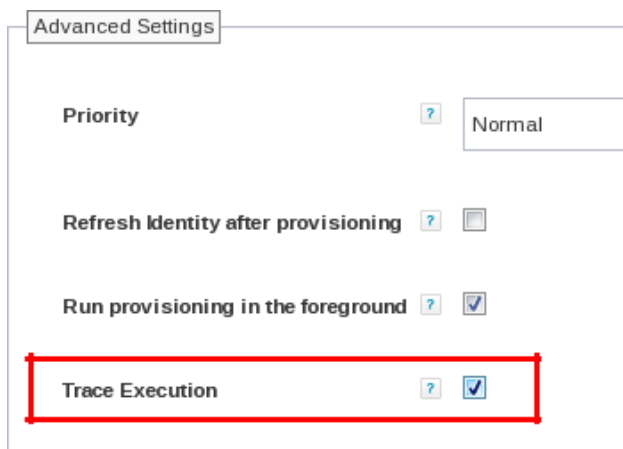
The **LCM Provisioning** workflow automatically checks for approval from the entitlement owner before provisioning the user's access. This out of the box behavior can be configured to support any desired functionality including policy checks, approvals, notifications, etc.

Enable Business Process (Workflow) Tracing

1. Navigate to **Setup** ➤ **Business Processes**
2. Select the **LCM Provisioning** Business Process, and on the right side of the screen, select the **Process Variables** tab.



3. Scroll down to the very bottom, and select **Trace Execution**. This will trace all workflow steps into the logs so that we can observe detailed workflow flow information.

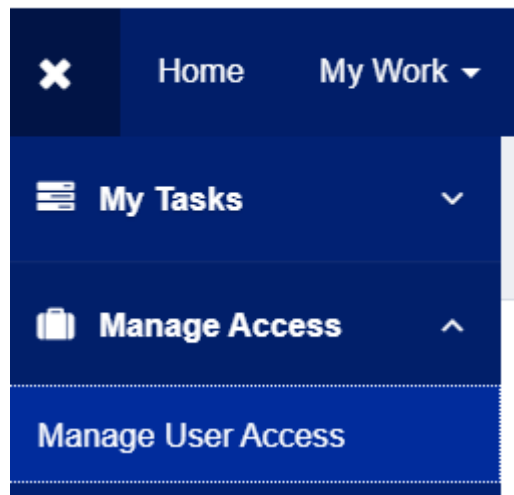


4. Click **Save**.

5. Start the desktop shortcuts **Tail Tomcat Standard Out** and **Tail Email Log**. During the request to add users to the VPN group in LDAP, we will view these logs to observe the workflow trace and emails being sent.

Login as a Manager and Request VPN Access for employees

1. Logout of IdentityIQ and login as **Catherine.Simmons/xyzzy**
2. Under **Manage Access**, select **Manage User Access**



a.

3. In the Available Identities list, you should only see direct reports for **Catherine.Simmons**. Select all of these direct reports and select **Submit**
4. On the **Request Access for 5 Identities** screen, search for **VPN**. Notice the search result count displayed on each tab.
5. Select the **Entitlements** tab and notice that all of our configured items are showing up on the VPN Entitlement such as Owner, Description.



6. Click **Next** to add the VPN entitlement to access request
7. On the confirmation page, if everything looks okay, click **Submit**.

Chosen Identities

Name	First Name	Last Name	Manager
Denise.Hunt	Denise	Hunt	Catherine.Simmons
Irene.Mills	Irene	Mills	Catherine.Simmons
Jeremy.Palmer	Jeremy	Palmer	Catherine.Simmons
Louis.Black	Louis	Black	Catherine.Simmons
Tammy.Daniels	Tammy	Daniels	Catherine.Simmons

Page 1 of 1 Show 5 items Displaying 1 - 5 of 5

Requested Access

Action	Name	Type	Comments	Risk S	Owner
	VPN • Attribute: groups • Application: LDAP	Entitlement		1	Randy.Knight

Page 1 of 1 Show 10 items Displaying 1 - 1 of 1

Submit Cancel Make Additional Changes

8. Click on **Track My Requests**.
9. There should be five requests in the queue, one for each subordinate employee that had the VPN entitlement requested for them. Click any request to see the current status of the request.

Access Requests

Filter by Identity

Advanced Search

Access Request ID	Priority	Type	Requester	Requestee	Request Date
5	Normal	Request Access	Catherine.Simm...	Louis.Black	1/3/14 4:32 PM

Request Items [View Complete Details](#)

Operation	Item	Value	Account	Application
Add	groups	cn=VPN,ou=groups,dc=training,dc=sailpoint,dc=com	cn=Louis.Black,ou=people,dc=training,dc=sailpoint,dc=com	LDAP

Pending Interactions

Description	Owner	Open Date	Details
Owner Approval - Account Changes for User: Louis.Black	Randy.Knight	1/3/14 4:32 PM	1 Approval

4	Normal	Request Access	Catherine.Simm...	Denise.Hunt	1/3/14 4:32 PM
3	Normal	Request Access	Catherine.Simm...	Tammy.Daniels	1/3/14 4:32 PM
2	Normal	Request Access	Catherine.Simm...	Irene.Mills	1/3/14 4:32 PM
1	Normal	Request Access	Catherine.Simm...	Jeremy.Palmer	1/3/14 4:32 PM

10. Observe the current status of the workflow in the log files.

- a. Check the output of the Email log file you should see that emails that were generated:

```
To: Randy.Knight@demoexample.com
Message-ID: <f6540250a97a44fba132515f41f64de3@example.com>
Subject: Changes requested to Tammy.Daniels need approval
MIME-Version: 1.0
Content-Type: multipart/mixed;
    boundary="--- =_Part_44_19584772.1396377798427"
X-Mailer: smptsend

-----=_Part_44_19584772.1396377798427
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Catherine.Simmons is requesting the following changes for 'Tammy.Daniels'

Application: LDAP
Account : cn=Tammy.Daniels,ou=people,dc=training,dc=sailpoint,dc=com
Operation: Add
Attribute: groups
Value(s): cn=VPN,ou=groups,dc=training,dc=sailpoint,dc=com
Priority: Normal
```

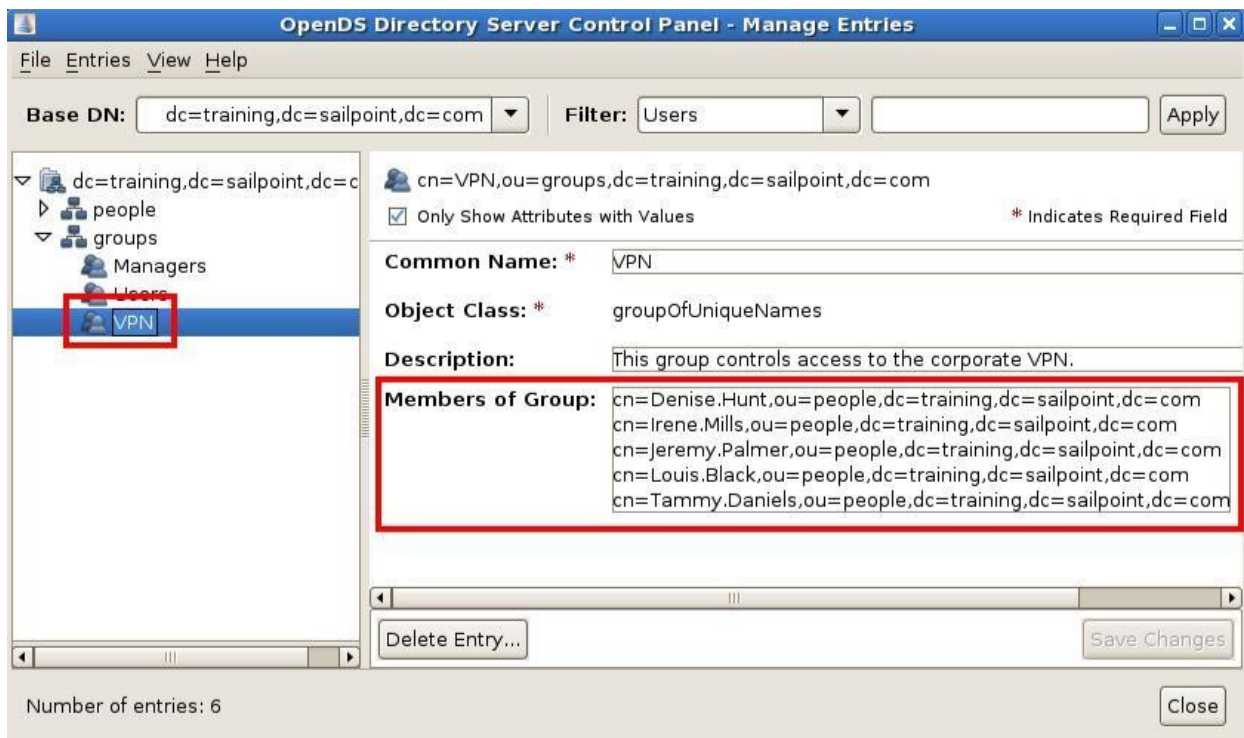
- b. Check the Standard Out log file and see that workflow tracing has occurred. The end of the trace shows that an approval has been requested:

```
Starting step Approval
Starting approval group in mode parallelPoll
Starting approval for Randy.Knight
Opening work item: Owner Approval - Account Changes for User:
Tammy.Daniels Starting step Approval
Starting approval group in mode parallelPoll
Starting approval for Randy.Knight
Opening work item: Owner Approval - Account Changes for User: Tammy.Daniels
```

- c. Logout and login as **Randy.Knight/xyzzzy**
- d. Click **Approvals** in the **Dashboard** and click the first approval.
- e. Handle the approval by selecting **OK** and then **Complete** the work.
- 1 Repeat the approval process for each user in the manager's department (there should be 5 total approvals to complete.)
- Notice the Standard Out log file after each item is approved by Randy:

```
Ending step Complete Identity Request
Skipping conditional step Update Ticket On Complete
Starting step end
Ending step end
Ending workflow Identity Request Finalize
Ending step Finalize
Ending workflow LCM Provisioning
```

2. Once the approvals are done, you can check in the OpenDS LDAP utility and confirm that 5 employees have correctly been added to the **VPN** group as shown here:



- b. Just to review what occurred:
- Once the manager requested that all 5 of her employees needed access to the VPN group, 5 workflows were started (each one was the **LCM Provisioning** workflow that is the default in IdentityIQ for Access Requests)
 - Each workflow determined that the owner of the **VPN** group was **Randy.Knight** from the settings in the Entitlement Catalog so the workflow routed the approval for each user to **Randy.Knight**
 - Randy.Knight** received an email notification and had 5 items in his inbox for his approval.
 - Once **Randy.Knight** approved each request, the workflows continued and provisioned access to the LDAP resource, which involved adding the users to the specific VPN group.
- c. Log out and back in as **Catherine.Simmons/xyzzy**

Exercise 4

Create and Manage Identities in IdentityIQ

Use Case ID:	L04 – E04		
Use Case Name:	Creating and managing identities		
Created By:		Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Admin, IIQ System		
Description:	Learn how to manage creating Identities and editing them using IdentityIQ with and without Identity Provisioning Policies		
Preconditions:	IIQ System is Up and Running, LCM setup		
Post conditions:	LCM configuration, Provisioning setup		
Normal Flow:	<ol style="list-style-type: none"> 1. Create identity form 2. Creating Identity from IdentityIQ 3. Provisioning required access 		
Exceptions:	NA		
Dependent Usecase:	NA		
Assumptions:	NA		
Notes and Issues:	NA		

Overview:

In this usecase, we are going to setup the following:

- Create identity form related to LCM
- Provisioning required access

Often, you will need to create Identities in IdentityIQ. When doing so, you can create them using LCM. LCM allows you to create and edit Identities and manage the creation and updating of the Identities using workflows to control the creation and editing processes. Also, you may define provisioning policies, which can help define the choices that are made when creating Identities in the system. In this exercise we will create identities two ways:

1. Using the out of the box configuration

2. Using pre-defined Provisioning Policies to help drive user's choices when creating a new identity.

Create an Identity using LCM

1. Logout and login as **Catherine.Simmons/xyzzz** and navigate to the dashboard and select
 - a. **Create Identity**
2. Create the Identity as shown here. Use **xyzzz** for the password.
 - a. Note that this is a default provisioning form that ships with IdentityIQ. As you enter the data, think about modifications that would make entering data less error prone and easier.

Create New Identity

If you would like to request that a new identity be created, please fill in the fields below. Fields marked with an asterisk are required.

Identity Name*	<input type="text" value="Fred.Smith"/>
Password*	<input type="password" value="*****"/>
Confirm Password*	<input type="password" value="*****"/>
First Name	<input type="text" value="Fred"/>
Last Name	<input type="text" value="Smith"/>
Email	<input type="text" value="fred.smith@example.com"/>
Manager	<input type="text" value="Catherine.Simmons"/> ▼
Display Name	<input type="text" value="Fred.Smith"/>
Inactive	<input type="text" value="False"/>
Location	<input type="text" value="Austin"/>
Employee ID	<input type="text" value="4141414141"/>
Region	<input type="text" value="Americas"/>
Status	<input type="text" value="Employee"/>

3. We will be presented with the confirmation screen, but since we are the manager as well, no approval is generated. Confirm the changes and click **Submit**.

The screenshot shows the 'Review & Submit' page in the SailPoint IdentityIQ interface. At the top, there are tabs for 'Dashboard' and 'Manage'. Below the tabs is a breadcrumb trail: 'Select Identity(s) > Create Identity > Review & Submit'. The main heading is 'Summary of Requests' with a subtext: 'Please verify the changes you have requested below.' Below this is a table with three columns: 'Action', 'Summary', and 'Identity'. The table contains one row for a 'Create' action. The 'Summary' column lists the details of the identity being created: name, firstname, lastname, email, manager, displayName, inactive status, location, empid, region, and status. The 'Identity' column shows the name 'Fred.Smith'. At the bottom of the table, there is a pagination bar showing 'Page 1 of 1' and a 'Show 10 items' dropdown. Below the table is a text area for 'Add Comments:' and three buttons: 'Submit', 'Cancel', and 'Make Additional Changes'.

Action	Summary	Identity
Create	Create Identity: name = 'Fred.Smith' firstname = 'Fred' lastname = 'Smith' email = 'fred.smith@example.com' manager = 'Catherine.Simmons' displayName = 'Fred.Smith' inactive = 'False' location = 'Austin' empid = '4141414141' region = 'Americas' status = 'Employee'	Fred.Smith

Page 1 of 1 Show 10 items

Add Comments:

Submit **Cancel** **Make Additional Changes**

- f. From the Dashboard, select **Track my Requests** and confirm that the Create Identity request operation was successful.
- g. Logout and Login in as **spadmin/admin**
- h. Navigate to the identity: **Fred.Smith** and confirm that the user was created correctly in IdentityIQ.
- i. As you probably could see, this was a tedious (and potentially error-prone) approach to entering an identity. In the next section, we will create a provisioning policy that will allow us to make creating an identity easier and provide nice features like allowed value dropdown selections, and data validation.

Define Provisioning Policies for Creating Identities

- h. Navigate to **Global Setting** **Import from File** and load the following files:
 /home/spadmin/ImplementerTraining/config/Rule-AllowedValues-Location.xml
 /home/spadmin/ImplementerTraining/config/Rule-AllowedValues-Region.xml
 /home/spadmin/ImplementerTraining/config/Rule-Validation-EmailAddress.xml

These rules will be used for our Provisioning Policies. The first two generate lists of allowed values we can use to populated drop-down lists. The last rule is used to validate that email addresses are correctly formatted.

2. Navigate to **Global Setting** ➤ **Identity Provisioning Policies** and next to **Create Identity**, select **Add Policy**

Configure Identity Provisioning Policy

Use the form below to build and modify the provisioning policies for creating and editing identities.

Below is a list of provisioning policies associated with this identity. You can add a new policy by clicking on the 'Add Provisioning Policy' button below or edit an existing one by clicking on it in the list.

Type	Name	Description	
Create Identity			 Add Policy
Update Identity			 Add Policy
Self-service Registration	Self-service Registration Form	This form is used to for self-service registration.	 Delete Policy

3. Configure the provisioning policy as shown:
 - a. Name: **Identity Create Policy**
 - b. Select **Add Field**
 - i. Attribute: **region**
 - ii. Display Name: **Region**
 - iii. Required: **checked**
 - iv. Scroll down to the **Value Properties** box.
 - v. Allowed Values: **Rule**
 1. Rule: **AllowedValues-Region**
 - vi. For the field, select **Save**
 - c. Select **Add Field**
 - i. Attribute: **location**
 - ii. Display Name: **Location**
 - iii. Required: **checked**
 - iv. Allowed Values: **Rule**
 1. Rule: **AllowedValues-Location**
 - v. For the field, select **Save**
 - d. Select **Add Field**
 - i. Attribute: **name**

- ii. Display Name: **Username**
 - iii. Help Text Value: **First.Last**
 - iv. Required: **checked**
 - v. For the field, select **Save**
 - e. Select **Add Field**
 - i. Attribute: **password**
 - ii. Display Name: **Password**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
 - f. Select **Add Field**
 - i. Attribute: **passwordConfirm**
 - ii. Display Name: **Password Confirmation**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
 - g. Perform an interim save of the **Identity Create Policy**
 - i. At the bottom right of the **Provisioning Policy Editor**, select **Save**
 - ii. Select **Save** to save the provisioning configuration
 - iii. Select **Identity Provisioning Policies**
 - iv. Select **Identity Create Policy**
- Note:** The Create Identity policy requires certain fields (i.e. name and password) to be defined before a save is allowed.
- h. Select **Add Field**
 - i. Attribute: **firstname**
 - ii. Display Name: **First Name**
 - iii. Required: **checked**
 - iv. For the field, select **Save**

- i. Select **Add Field**
 - i. Attribute: **lastname**
 - ii. Display Name: **Last Name**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
- j. Select **Add Field**
 - i. Attribute: **email**
 - ii. Display Name: **Email**
 - iii. Required: **checked**
 - iv. Scroll down to **Value Properties**
 - v. Value: **<enter email address here>**
 - vi. Validation: **Rule**
 - 1. Rule: **Validation – Email Field**
 - vii. For the field, select **Save**
- k. Select **Add Field**
 - i. Attribute: **manager**
 - ii. Display Name: **Manager**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
- l. Select **Add Field**
 - i. Attribute: **displayName**
 - ii. Display Name: **Display Name**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
- m. Select **Add Field**
 - i. Attribute: **inactive**

- ii. Display Name: **Inactive**
 - iii. Type: **Boolean**
 - iv. Required: **checked**
 - v. Value Properties, Value: **False**
 - vi. For the field, select **Save**
- n. Select **Add Field**
 - i. Attribute: **empId**
 - ii. Display Name: **Employee ID**
 - iii. Required: **checked**
 - iv. For the field, select **Save**
- o. Select **Add Field**
 - i. Attribute: **status**
 - ii. Display Name: **Status**
 - iii. Required: **checked**
 - iv. Scroll down to Value Properties
 - v. Value: **Contractor**
 - vi. Allowed Values: **Value**
 - 1. Add **Contractor** and **Employee** to the list

The screenshot shows the 'Allowed Values' configuration window. At the top, there are four radio buttons: 'None', 'Value' (which is selected), 'Rule', and 'Script'. Below the buttons is a list box containing two items: 'Contractor' and 'Employee'. Each item has a small 'x' icon to its left. To the right of the list box is a '+' button. The background of the window is light gray.

- vii. For the field, select **Save**
 - p. Confirm that the entire Provisioning Policy looks like this:

Provisioning Policy Editor

Name:	Identity Create Policy
Description:	
Owner:	<input checked="" type="radio"/> None

Add Field		Remove Field	Edit
<input type="checkbox"/>	Name	Type	
<input type="checkbox"/>	Region	string	
<input type="checkbox"/>	Location	string	
<input type="checkbox"/>	Username	string	
<input type="checkbox"/>	Password	secret	
<input type="checkbox"/>	Password Confirmation	secret	
<input type="checkbox"/>	First Name	string	
<input type="checkbox"/>	Last Name	string	
<input type="checkbox"/>	Email	string	
<input type="checkbox"/>	Manager	Identity	
<input type="checkbox"/>	Display Name	string	
<input type="checkbox"/>	Inactive	boolean	
<input type="checkbox"/>	Employee ID	string	
<input type="checkbox"/>	Status	string	

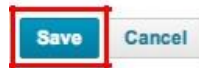
- q. Select **Save** to save the Identity Provisioning Policy
- 4. Select **Save** to save the provisioning configuration:

Configure Identity Provisioning Policy

Use the form below to build and modify the provisioning policies for c

Below is a list of provisioning policies associated with this identity. You ca
it in the list.
















Type	Name	Description
Create Identity	Identity Create Policy	
Update Identity		
Self-service Registration	Self-service Registration Form	This form is usec




- 5. Logout and login as **Catherine.Simmons/xyzzz**
- 6. From the Dashboard, click **Create identity** and observe the new Create New Identity page:

Region*	<input type="text"/>	▼
Location*	<input type="text"/>	▼
Username*	<input type="text"/>	
Password*	<input type="password"/>	
Password Confirmation*	<input type="password"/>	
First Name*	<input type="text"/>	
Last Name*	<input type="text"/>	
Email*	<input type="text" value=" <enter email address here>"/>	
Manager*	<input type="text"/>	▼
Display Name*	<input type="text"/>	
Inactive*	<input type="checkbox"/>	
Employee ID*	<input type="text"/>	
Status*	<input checked="" type="radio"/> Contractor <input type="radio"/> Employee	

7. Without entering any data at all, click **Submit** and observe that our email validation rule and required fields will warn the user about any data entry issues:

Region*	<input type="text"/>		 The field Region is required
Location*	<input type="text"/>		 The field Location is required
Username*	 <input type="text"/>		 The field Username is required
Password*	<input type="password"/>		 The field Password is required
Password Confirmation*	<input type="password"/>		 The field Password Confirmation is required
First Name*	<input type="text"/>		 The field First Name is required
Last Name*	<input type="text"/>		 The field Last Name is required
Email*	<input type="text" value="<enter email address here>"/>		 Need an @ sign in a valid email., Need an . in a valid email.
Manager*	<input type="text"/>		 The field Manager is required
Display Name*	<input type="text"/>		 The field Display Name is required
Inactive*	<input type="checkbox"/>		
Employee ID*	<input type="text"/>		 The field Employee ID is required
Status*	<input checked="" type="radio"/> Contractor <input type="radio"/> Employee		

8. Fill in the information as shown. Use *xyzzzy* for the password.

Region*	Americas	▼
Location*	Austin	▼
Username*	 Bob.Smith	
Password*	*****	
Password Confirmation*	*****	
First Name*	Bob	
Last Name*	Smith	
Email*	bob.smith@example.com	
Manager*	Catherine.Simmons	▼
Display Name*	Bob.Smith	
Inactive*	<input type="checkbox"/>	
Employee ID*	4e4e4e4e	
Status*	<input type="radio"/> Contractor <input checked="" type="radio"/> Employee	

9. Click **Submit** to submit the new Identity request

10. When you see the confirmation page, review and **Submit** the request.

11. Logout and login as **spadmin/admin** and confirm that **Bob.Smith** has an identity cube. Note that this Identity cube has no entitlements or accounts. Currently it is just a shell cube.

View Identity Bob.Smith

Attributes	Entitlements	Application Accounts	Policy
User Name	Bob.Smith		
First Name	Bob		
Last Name	Smith		
Email	bob.smith@example.com		
Manager	Catherine.Simmons		
Department			
Location	Austin		
Employee ID	4e4e4e4e		
Region	Americas		
Job Title			
Cost Center			
Status	Employee		

12. Note that you can further customize the creation of new Identities by the following techniques:
- Additional logic in your provisioning policies
 - Data validation - Detecting duplicate usernames or email addresses
 - Precalculation of an EmployeeID number
 - Customizing the out of the box workflow **LCM Create and Update** that is responsible for all create and edit operations that occur on Identities when using LCM

Exercise 5

Account Management with Lifecycle Manager

Use Case ID:	L04 – E05		
Use Case Name:	LCM account management		
Created By:		Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Admin, IIQ System		
Description:	The objective of this section is to manage account access using Lifecycle Manager.		
Preconditions:	IIQ System is Up and Running, LCM setup		
Post conditions:	Account management, Provisioning		
Normal Flow:	<ol style="list-style-type: none">1. Managing accounts using IdentityIQ2. Performing CRUD operation		
Exceptions:	NA		
Dependent Usecase:	NA		
Assumptions:	NA		
Notes and Issues:	NA		

Overview:

In this usecase, we are going to setup the following:

- Managing account using LCM
- Performing CRUD on accounts using IdentityIQ LCM

We will explore the following Account functions in this exercise:

- Creating a new account
- Requesting a role
- Requesting a role that will cause a new account request to occur
- Enabling and Disabling accounts

- Unlocking Accounts

Configure Lifecycle Manager to support account requests

1. Navigate to **Setting** and select **Lifecycle Manager**
2. Click the **Lifecycle Actions** tab and scroll down past to the **Manage Account Options** section and turn on **Allow requesting new accounts** as shown here:

Manage Accounts Options

☐ Show Enable/Unlock decision buttons regardless of whether the account is disabled or unlocked.

Choose which actions are enabled for each type of manage accounts request. Any options unselected will be unavailable to any user attempting to make that type of request.

Action	My Actions	Subordinate
Delete	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Disable	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Enable	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Unlock	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

3. Under **General Options** enable **Applications that support additional account requests** for all the applications or **LDAP** and **PRISM**
4. Scroll down to the **Manage Accounts Options** and in the drop-down selection box that says: **Applications that support account only requests** add **LDAP** and **PRISM** to the list:

Applications that support account only requests

☒ LDAP

☒ PRISM

5. Click **Save**

Request a New LDAP Account for our New User Fred.Smith

1. Logout and login as **Catherine.Simmons/xyzzz**
2. From the Dashboard, select **Manage Accounts** and **For Others**. You will be presented with a list of all the users who report to **Catherine.Simmons**

Note: This is because out of the box, managers can only request items for their direct reports. This is fully configurable through LCM

3. Request a new LDAP account for **Fred.Smith** and **Submit** the request
4. After you submit the request, look at Catherine's dashboard.
 - a. What warning is displayed? Why?

5. Check the status of the Access Request under **Track My Requests**, determine who is the approver and login as that user and approve the access request:

Access Requests

The screenshot shows the 'Access Requests' interface. At the top, there is a search bar with 'Filter by Identity' and a magnifying glass icon, and an 'Advanced Search' button. Below this is a table with columns: Access Request, Priority, Type, Requester, Requestee, Request Date, Current Step, and Com. A single request is listed with ID 8, Priority Normal, Type Manage Acco..., Requester Catherine.Sim..., Requestee Fred.Smith, Request Date 1/3/14 5:37 PM, and Current Step Initialize. Below the table, there is a section for 'Request Items' with a link to 'View Complete Details'. This section contains a table with columns: Operation, Account, Application, Instance, Comments, Approval Status, and Provisioning Status. The data row shows 'Create' for Operation, 'LDAP' for Application, 'Pending' for Approval Status, and 'Pending' for Provisioning Status. Below this is a section for 'Pending Interactions' with a table with columns: Description, Owner, Open Date, and Details. The data row shows 'Owner Approval - Account Changes for User: Fred.Smith' for Description, 'The Administrator' for Owner, '1/3/14 5:37 PM' for Open Date, and '1 Approval Item(s)[Click for Details]' for Details.

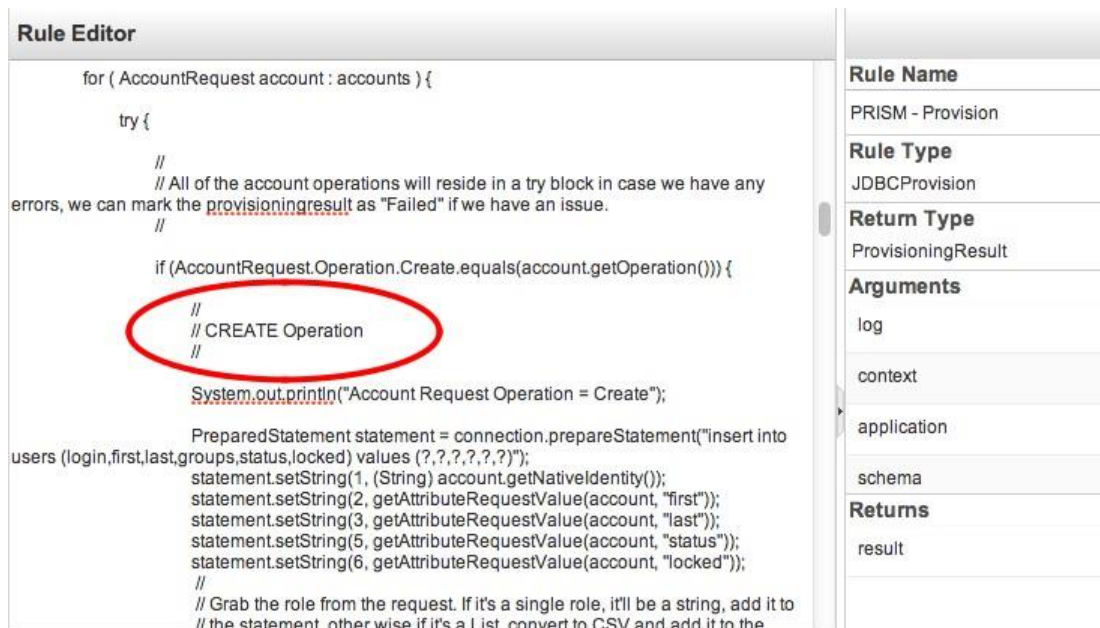
6. Check the LDAP repository and confirm that Fred.Smith has an account in the LDAP server. **Note:** You can search for users by filtering for users and using wildcard searching:

The screenshot shows the 'OpenDS Directory Server Control Panel - Manage Entries' window. The 'Base DN' is set to 'dc=training,dc=sailpoint,dc=com'. The 'Filter' is set to 'Users'. The search criteria is 'Fred*' and the 'Apply' button is highlighted with a red box. The search results show a tree structure with 'dc=training,dc=sailpoint,dc=com' expanded, showing 'people' expanded, and 'Fred.Smith' highlighted with a red box.

Request a New PRISM Account for Fred.Smith

Our new employee Fred.Smith also needs an account on the **PRISM** application. Next we will request a **PRISM** account for him. In this section, we will be relying on the **PRISM - Provision** rule to provision this access to the JDBC resource.

1. Login as **spadmin/admin** and confirm the PRISM application provisioning rule
 - a. Navigate to the PRISM application, bottom of the Attributes tab, and view the JDBC Provision Rule: **PRISM - Provision**



```
for ( AccountRequest account : accounts ) {  
    try {  
        //  
        // All of the account operations will reside in a try block in case we have any  
        // errors, we can mark the provisioningresult as "Failed" if we have an issue.  
        //  
        if (AccountRequest.Operation.Create.equals(account.getOperation())) {  
            //  
            // CREATE Operation  
            //  
            System.out.println("Account Request Operation = Create");  
            PreparedStatement statement = connection.prepareStatement("insert into  
users (login,first,last,groups,status,locked) values (?, ?, ?, ?, ?, ?)");  
statement.setString(1, (String) account.getNativeIdentity());  
statement.setString(2, getAttributeRequestValue(account, "first"));  
statement.setString(3, getAttributeRequestValue(account, "last"));  
statement.setString(5, getAttributeRequestValue(account, "status"));  
statement.setString(6, getAttributeRequestValue(account, "locked"));  
            //  
            // Grab the role from the request. If it's a single role, it'll be a string, add it to  
            // the statement. other wise if it's a List. convert to CSV and add it to the
```

Rule Name
PRISM - Provision

Rule Type
JDBCProvision

Return Type
ProvisioningResult

Arguments
log
context
application
schema

Returns
result

- b. Scroll through the rule and list three (of five) provisioning operations handled by this rule (the first provisioning operation is circled above):

2. Login as **Catherine.Simmons/xyzzz**
3. From the Dashboard, select **Manage Accounts** and **For Others**
4. Request a new PRISM account for **Fred.Smith** and **Submit** the request which will send it to the **PRISM Application Owners** workgroup (of which **Walter.Henderson** is a member).
5. From a terminal window, login into MySQL and confirm that there is no account in the application for **Fred.Smith**.

```
[spadmin@training ~]$ mysql -u root -p  
Enter password: root  
mysql> use prism  
  
mysql> select * from users where login =  
'Fred.Smith'; Empty set (0.00 sec)
```

6. Login as the approver, **Walter.Henderson/xyzzz** and approve the changes for the account request on **PRISM**. At this time, the provisioning request for a new **PRISM** account is sent to

the **PRISM - Provision** Rule. This Rule includes some print statements that inform the user of the request that is passed in and the final result. In this case, the following information is printed to the Standard Out log:

```
*****
Entering Provisioning Rule for PRISM
Current Time = Sat Nov 10 09:57:53 CST 2012
*****
***
The Provisioning Plan being passed in =
***
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE ProvisioningPlan PUBLIC "sailpoint.dtd" "sailpoint.dtd">
<ProvisioningPlan nativeIdentity="Fred.Smith" targetIntegration="PRISM">
  <AccountRequest application="PRISM" nativeIdentity="Fred.Smith"
    op="Create"> <AttributeRequest name="first" op="Add" value="Fred"/>
    <AttributeRequest name="last" op="Add" value="Smith"/> <AttributeRequest
    name="status" op="Add" value="A"/> <AttributeRequest name="locked"
    op="Add" value="N"/>
  </AccountRequest>
  <Attributes>
    <Map>
      <entry key="identityRequestId" value="0000000018"/>
      <entry key="requester" value="Catherine.Simmons"/>
      <entry key="source" value="LCM"/>
    </Map>
  </Attributes>
  <Requesters>
    <Reference class="sailpoint.object.Identity" id="ff8080813ade1e61013ae1068df1042d"
    name="Catherine.Simmons"/>
  </Requesters>
</ProvisioningPlan>

*****
Account Request Operation = Create
Preparing to execute: org.apache.commons.dbcp.DelegatingPreparedStatement@1a30be6
*****
Exiting Provisioning Rule for PRISM.
Result=
<ProvisioningResult status="committed"/>

*****
*****
```

7. Back in the terminal window, run the following to confirm that the account is there. (If you've logged out of MySQL since the last usage, remember to specify "use prism".)

```
mysql> select * from users where login = 'Fred.Smith';
```

login	description	first	last	groups	status	locked	lastLogin
Fred.Smith	NULL	Fred	Smith		A	N	NULL

```
1 row in set (0.00 sec)
```

8. The default values that are created in the **PRISM** application are determined by the Provisioning Policy attached to the **PRISM application**. Notice that the **groups** attribute is empty. If desired, our default provisioning policy could be changed to grant basic **User** access by provisioning the attribute **groups** to include **User** by default.
9. If you are interested in more on the Provisioning Policy and PRISM provisioning rule, Login as **spadmin/admin** and look at the **PRISM** application. Investigate both the **Provisioning Policy** and the **PRISM - Provision** rule. The provisioning policies provide the values to the plan, and the rule executes what is specified in the plan. Understanding this basic behavior of our provisioning capabilities is very important to understanding how the process works.