1. Introduction

The Rally Connector for LDAP is designed to integrate LDAP with Rally for enterprise user management by facilitating initial provisioning and the ability to enable/disable users based on LDAP filter parameters. This connector allows the enterprise administrator to initiate a Ruby script that runs on a set interval to add new employees and/or remove terminated employees from your Rally Enterprise subscription. Currently the connector is only supported on the Windows operating system.

The connector supports Active Directory on Windows and is configured via an XML file. A log file is created to track all changes made in Rally and LDAP by the connector. The connector supports the implementation of the LDAP client protocol, per RFC-2251.

1.2 Overview of Features

The connector only creates/modifies Rally user objects and never modifies LDAP entries. When a new user is created in Rally, the login name and email address are populated based on how you map these fields to LDAP attributes. Optional fields include First Name, Last Name and Display Name. The mapping of Rally fields to LDAP attributes are set up in the configuration file.

The connector provides two services:

1. Provision new users in Rally that meet LDAP filter criteria defined in the configuration file. Rally then automatically sends email to new users with information on setting up a password and logging into Rally.
2. Enable/disable existing users in Rally that meet both LDAP filter criteria defined in the configuration file.

The configuration file specifies which services to run, and in what order. For example, to only enable/disable existing users in Rally, you would specify "2" for <ServicePriority> in the config.xml file.

The connector runs the desired services at an interval specified in the configuration file. We recommend choosing an interval no shorter than 30 minutes.

2. Pre-Installation Requirements

2.1 Software/Hardware Requirements

The following are the hardware and software requirements to install and run the Rally Connector for LDAP.

- A browser (Internet Explorer or Firefox)

- Windows 2003 Server with Active Directory OR OpenLDAP 2.4.12 with Berkley DB 4.7.25
- Run connector on Windows XP or Windows 2003 Server
- A Rally subscription. Workspace administrator or subscription administrator privileges are required to run the connector.
- User with access to LDAP. You will need to know the host name and port for LDAP.
- A computer with access to Rally and LDAP server.
- Ruby programming language and Ruby gems (libraries) installed. Discussed in installation steps below.

2.2 Pre-Installation Checklist

- Determine which LDAP attribute/value combination indicates that someone in your LDAP store should be a Rally user. You can create a custom attribute in LDAP or use an existing attribute. If you wanted to assign all Rally users to a particular group, identify the Distinguished Name of the group and insert that in the LdapRallyValue tag in your configuration file. For example, this would only create users that belong to the Users group:

```
<!---LDAP filter to define who is a Rally user in the LDAP store--->
 <!---Used for service 1 (create) & service 2 (enable/disable) --->
 <LdapRallyAttr>memberOf</LdapRallyAttr>
 <LdapRallyValue>CN=Users,CN=Roles,DC=rally,DC=lab</LdapRallyValue>
```

- Determine which LDAP attribute/value combination indicates a disabled user. The enable/disable service uses the base LDAP filter (above) to determine Rally users and the disable filter to determine what constitutes a disabled or enabled user.

2.3 Implementation Plan

Most users follow an implementation plan similar to the following:

1. Determine which services to run.

2. Test the connector using the "Preview" mode by specifying a "p" after the service number in the XML configuration file (e.g. <ServicePriority>1p,2p</ServicePriority>). The preview mode will not create, enable or disable any users in Rally.

3. Review **logfile.log** (located in the same directory as the connector script) to see if any unexpected errors are encountered, and verify that the actions that would be taken are the correct ones. Report any errors to Rally Customer Support.

3. Installation

Basic Installation Steps

1. Install Ruby & the Rally Connector to LDAP code.
2. Setup the config.xml to reflect the Rally & LDAP environments.
3. Run configure_credentials script to encrypt appropriate passwords.
4. Run ldap_rally_service.rb to start the connector.
5. Setup the connector using Windows task scheduler every hour or desired interval.

3.1 Install Ruby & the Rally Connector for LDAP Source Files

1. Install the Ruby programming language for Windows. We have found the "one-click installer" at http://rubyinstaller.rubyforge.org/wiki/wiki.pl?RubyInstaller to be a good choice.
2. Install the **builder, rally_rest_api** and **net/ldap** gems by entering the following from the DOS command window:

```
gem install builder -y
gem install rally_rest_api -y
gem install ruby-net-ldap -y
gem install actionmailer -y
```

Answer **yes** to any questions about installing required dependencies.

3. Save the 'rally_ldap_connector.zip' file locally on your machine. Navigate to <install_dir> in Windows Explorer, right-click on the 'rally_ldap_connector.zip' zip file, and select **Extract All ...**

3.2 Setup the config.xml File

Create an initial config.xml by making a copy of sample.xml. Edit and save config.xml, entering the appropriate values between each begin and end tag. The XML is commented throughout to provide details on expected values for each tag.

Set <ServicePriority> as described in the **Implementation Plan** section above. We strongly recommend using the "preview" mode initially, to verify that the connector provision and enable/disable the users you expect. "Preview" mode is services 1p and 2p.

Set <RunInterval> to update frequency (in minutes) at which you wish transfers to occur. We recommend a <RunInterval> of no less than 30 minutes. To avoid long waits during initial testing, it's best to simply stop and restart ldap_rally_service.

Set <EmailEnabled> to true if you wish a certain user or group to receive an email when new users are created or disabled in Rally (based on existence or state in your LDAP repository). Only SMTP email servers are supported with authentication. For example, if you use GMail then your smtp server would be smtp.gmail.com and your port would be 587. You will also need to set the following values:

- <EmailServerHost> specify your Email Server
- <EmailServerPort> port # for Email Server
- <EmailAccountUserName> the email of the account you would like to use to SEND the email (who the email will be FROM)
- <EmailAccountEncryptedPassword> Use the configure_credentials.rb script (located within the /encryption directory) to set this value. Plain text passwords will not be accepted.
- <SendEmailTo> the email address you want the message sent TO

Example config.xml

```
<ConfigurationInfo>

 <!---Rally server URL or IP address--->
 <!---e.g. https://rally1.rallydev.com/slm for Production--->
 <RallyServerURL>https://sandbox.rallydev.com/slm</RallyServerURL>

 <!---Rally workspace admnistrator or subscription admin--->
 <RallyUserName>user@company1.com</RallyUserName>
 <RallyPassword>RallyPassword</RallyPassword>

 <!---LDAP connection information--->
 <LdapHost>host.company1.com</LdapHost>
 <LdapPort>389</LdapPort>

 <!---User with LDAP read permissions --->
 <LdapUserName>yourcompany\LDAPUserName</LdapUserName>
 <LdapPassword>LdapPassword</LdapPassword>

 <!---LDAP root distinguished name under which all subsequent queries are issued-->
 <!---If domain is www.company1.com, then it might be cn=users,dc=company1,dc=com-->
 <LdapTreeBase>cn=users,dc=company1,dc=com</LdapTreeBase>

 <!---LDAP filter to define who is a Rally user in the LDAP store
 Used for service 1 (create) & service 2 (enable/disable) -->
 Example of using group membership to determine if someone is a Rally user
   <LdapRallyAttr>memberOf</LdapRallyAttr>-->
   <LdapRallyValue>CN=Users,CN=Roles,DC=rally,DC=lab</LdapRallyValue>
  -->
 <LdapRallyAttr>extensionAttribute1</LdapRallyAttr>
 <LdapRallyValue>Rally</LdapRallyValue>

 <!---LDAP filter for what users should be disabled in Rally--->
 <LdapInactiveAttr>userAccountControl</LdapInactiveAttr>
 <LdapInactiveValue>66050</LdapInactiveValue>

 <!---Required mapping--->
 <!---Enter LDAP attribute name between each tag. --->
 <RallyLoginName>mail</RallyLoginName>
 <RallyEmailAddress>mail</RallyEmailAddress>

 <!---Optional mapping--->
 <!---Enter LDAP attribute name between each tag. Only used if provisioning new users in Rally --->
 <!--<RallyFirstName>givenname</RallyFirstName>
 <RallyLastName>sn</RallyLastName>
 <RallyDisplayName>displayname</RallyDisplayName>-->

 <!-- Email Settings -->
 <!-- Set EmailEnabled to TRUE and populate related fields to begin using -->
 <EmailEnabled>false</EmailEnabled>
 <EmailServerHost>smtp.mydomain.com</EmailServerHost>
 <EmailServerPort>587</EmailServerPort>
 <EmailAccountUserName>name@mydomain.com</EmailAccountUserName>
 <EmailAccountEncryptedPassword>211 210 226 55 178 45 250 211 122 67 185 226 222 90 158
69</EmailAccountEncryptedPassword>
 <SendEmailTo>recipient@domain.com</SendEmailTo>

 <!--- Specifies how often, in minutes, to run the services defined in <ServicePriority> --->
 <RunInterval>30</RunInterval>

 <!--- ServicePriority identifies the services to run and the order to run
 Value  -  Service Name
 LIVE MODE
 1      - Create a new user in Rally if matches LDAP Rally filter
 2      - Disable/enable existing Rally user if matches LDAP inactive filter
 PREVIEW MODE
 1p     - Service 1 preview mode.  New user login name is written to log file but user not created in Rally.
 2p     - Service 2 preview mode.  User login name written to log file if disabled/enabled.  No users
enabled or disabled.

 Example 1: <ServicePriority>1p,2p</ServicePriority>
 Example 2: <ServicePriority>2,1</ServicePriority>
 --->
 <ServicePriority>1p,2p</ServicePriority>

</ConfigurationInfo>
```

3.3 Run the configure_credentials.rb Script

If you have <EmailEnabled> set to true, you will need to encrypt your email password. To encrypt the email password, CD into the encryption directory from the command line and then run configure_credentials.rb. For example:

```
   CD <rally_ldap_install_dir>/encryption
   ruby configure_credentials.rb
```

Once you run this command, it will prompt you to enter 1 (Rally password), 2 (LDAP password) or 3 (email password) depending on the password you wish to encrypt.

Running this script will remove the plain text password and insert an encrypted password into the configuration file.

3.4 Running the Service

On Windows, if Ruby was installed using the One-Click Ruby Installer at http://rubyforge.org/projects/rubyinstaller/ , open a DOS shell and type:

```
ldap_rally_service <config_file_name>
```

where config_file_name if the name of your configuration file (e.g. ldap_rally_service config.xml).

Otherwise, if the ruby interpreter is in your Windows PATH environment variable, you should be able to type:

```
ruby ldap_rally_service.rb <config_file_name>
```

to start the service.

3.4 Setup Windows Task Scheduler

Follow these steps to setup the Windows Task Scheduler for the LDAP connector:

1. Create a batch file called "start_rally_ldap.bat" in your rally_ldap_connector installation directory that contains the following:

```
cd <install_dir>
ruby ldap_rally_service.rb config.xml
```

<intall_dir> will generally be something like C:\rubycode\ldap_rally_connector

There is only one log file so a user could also include clever logic in the batch script to create a new log file for each day.

2. Open rally_ldap_connector.rb and **comment out** (# is the comment character) the following 3 lines in the run method:

```
def run()
  rally_ok = connect_to_rally()
  ldap_ok = connect_to_ldap()

  if (ldap_ok && !rally_ok.nil?)
  services = @services_to_run.split(',')
    #loop do
      run_services()
      #sleep(60*@run_interval.to_f)
    #end
  end
end
```

3. Create a scheduled task by going to Control Panel >> Scheduled Tasks. Browse to the start_rally_ldap.bat to specify the program you want to run. Set the task to run daily on any set interval that is appropriate. If necessary, create a second Scheduled Task that runs when the computer starts up to force the script to run any time the machine is rebooted.

4.0 Known Issues

5.0 Frequently Asked Questions

1. How should an organization deal with multiple LDAP servers? For example, the North America site uses na.company.com and Europe uses emea.company.com

Setup a secure Rally group that both LDAP servers can access. Run a version of the connector for each LDAP server by modifying the config,xml and changing the root

base and LDAP connection details. e.g.

```
<LdapTreeBase>cn=users,dc=emea,dc=company,dc=com</LdapTreeBase>
```

2. I created a Rally group, but the users are not disabled from Rally when I remove a LDAP user from the group. I have setup the Rally filter as:

```
<LdapRallyAttr>memberOf</LdapRallyAttr>
<LdapRallyValue>CN=Users,CN=Roles,DC=rally,DC=lab</LdapRallyValue>
```

This is working as expected. The disable filter defines who is disabled from LDAP and is not related to the LdapRallyAttr and LdapRallyValue. e.g.

```
<LdapInactiveAttr>userAccountControl</LdapInactiveAttr>
<LdapInactiveValue>66050</LdapInactiveValue>
```

6.0 Contacting Support

Please submit a Rally Support case from Rally by navigating to **Help & Training** and click the **Contact Support** tab or visit http://www.rallydev.com/support.jsp.