ellucian

Luminis Platform
Installation Guide

Release 5.3
July 2016

Notices

Ellucian
4375 Fair Lakes Court
Fairfax, VA 22033
United States of America

Contents

Install Luminis Platform

Suggested prerequisites to verify before you install Luminis Platform and its components, and steps to set up the various installation files and run the Luminis installation scripts.

Before you begin the installation process, consider this information:

- Plan a virtual hostname before starting the installation. The common name on the Secure Sockets Layer (SSL) certificate should match the virtual hostname.
- Most of the issues encountered while installing Luminis Platform can occur when you install the different tiers on separate servers. For example, a typo in one of the `setup.properties` files can lead to errors and install failures. If in doubt, assume case sensitivity and pay strict attention to port numbers, admin names, and protocols (HTTP versus HTTPS, and so forth).
- You can mitigate installation and configuration errors by following the "Site Analysis Spreadsheet." When you complete a multi-server install, you should install the LDAP tier first. The Admin and CAS tiers dependent on the installed and running LDAP tier must be able to make successful bind requests to the LDAP tier.

If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

Related Links

Customize the installation values on page 13

Uninstall Luminis Platform software

Procedure to uninstall the current Luminis Platform software before you install the new software.

About this task

Warning! If you are reusing a server, you must uninstall all applications and reinstall the OS.

To uninstall Luminis Platform, run this command:

```
uninstall
```

Alternatively, to drop all database tables and uninstall Luminis Platform, run this command:

```
uninstall -d
```

Warning! This method is not recommended if you are installing Luminis Platform on multiple tiers. It is recommended if you are installing all components (LDAP, CAS, Portal and Admin) on a single machine.

Before you install Luminis Platform

Review the ports and other settings detailed in "Customize the installation values" and fill out the "Site Analysis Worksheet."

Each server included in your Luminis Platform deployment requires a separate `setup.properties` file containing customized configuration settings including port numbers, passwords, database settings, installation directory, and a list of components to install. The default values are used for the installation unless you override these values with the `setup.properties` files before the installation.

You must first set up an Oracle 11g, Oracle 12c, or MySQL 5.0 and above database to use as a single data source for Luminis Platform components. CAS, Admin and Portal tier servers all require a database for storage.

Before installation, the supporting database must be prepared as outlined in the following sections.

Java requirements

The Luminis Platform installer will install Java for its own use.

You do not need to pre-install Java for the installer to function properly, but if Java is already installed on the system, it must be version 6.1 or higher.

We recommend that any existing Java be removed from the system before installation to avoid potential conflict.

Operating systems requirements

Luminis Platform 5.x may be installed on either Redhat Linux version 5 or 6 (x86 32 or 64 bit) or Solaris SPARC 10 (32 or 64 bit).

The `gtar` command must be available on the execution path. On Linux, `gtar` is usually available by default. On Solaris, it may commonly be found in the `/usr/sfw/bin` directory. The path may be modified by executing the following command before you run the installation:

```
set PATH=$PATH:/usr/sfw/bin
```

This command is only required for installation and patching Luminis Platform versions 5.1.0 and later. It is not required to run Luminis Platform after the installation is complete.

Oracle requirements

At minimum, these requirements should be observed when installing and setting up an Oracle database to support Luminis Platform.

About this task

- Row-level locking should be configured
- A Luminis Platform database instance should exist with a database user that has a default tablespace in that database instance
- The database user should also set a TEMP tablespace
- The database user must be able to connect to the database and create new tables
- The database user should obtain the recommended grants

Your Oracle or MySQL database must be configured to use the UTF-8 character set. It is also important to note that Oracle 11g and Oracle 12c are supported; however, the Oracle 10g dialect must be used when installing Luminis Platform. This is done by setting the `luminis.database.dialect` property to org.hibernate.dialect.Oracle10gDialect in the `setup.properties` files before installation.

To create an Oracle tablespace and Oracle user setup:

Note: This is just an example of how to create a tablespace and user in Oracle. Ellucian recommends that a certified Oracle database administrator perform this initial setup.

Procedure

1. Create an Oracle 10g instance called LUMIN5TEST or LUMIN5PROD.
2. In the LUMIN5TEST instance, create a 200 MB tablespace called LUMIN5TEST.

   A sample SQL may appear as follows:

   ```
   CREATE SMALLFILE TABLESPACE "LUMIN5TEST" DATAFILE 'D:\APP\ORACLE
   \ORADATA\LUMSVCS\lumin5test.dbf'

   SIZE 200M AUTOEXTEND ON NEXT 10M MAXSIZE UNLIMITED LOGGING EXTENT
   MANAGEMENT LOCAL SEGMENT

   SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
   ```
3. Create an Oracle user named LUMIN5USER with these attributes:

   Password: `********`

   Default tablespace: `LUMIN5TEST`

   Default TEMP tablespace: `TEMP`

   Role(s): `CONNECT`

   System Privileges: `CREATE ANY INDEX, SYNONYM, SEQUENCE, TRIGGER, TABLE, and VIEW`

   `DROP ANY INDEX, SYNONYM, SEQUENCE, TRIGGER, TABLE, and VIEW`

A sample SQL may appear as follows:

```
CREATE USER "LUMIN5USER" PROFILE "DEFAULT" IDENTIFIED BY "*******"
DEFAULT TABLESPACE "LUMIN5TEST" TEMPORARY TABLESPACE "TEMP" ACCOUNT
UNLOCK;
GRANT "CONNECT" TO "LUM5USER";
GRANT RESOURCE TO "LUM5USER";
GRANT CREATE INDEX TO "LUM5USER";
GRANT CREATE VIEW TO "LUM5USER";
GRANT UNLIMITED TABLESPACE TO "LUM5USER";
```

Related Links

1-Tier development installation with Oracle 10g/11g on page 26

MySQL requirements

Requirements for installation and set up for a MySQL database to support Luminis Platform.

About this task

- Install the Server and Client Tools
- A Luminis Platform database instance should exist with a user possessing a default database in that MySQL instance
- The database user should only have rights to the database instance set up to support Luminis Platform and should have all permissions for this instance

To create a MySQL tablespace and SQL user setup:

Note: This is just an example of how to create a tablespace and user in MySQL. Ellucian recommends that a MySQL database administrator perform this initial setup.

Procedure

1. Log into the MySQL client as root user.
2. Create a database called LUMIN5TEST or LUMIN5PROD.

   A sample SQL may appear as follows:

   Create database LUMIN5TEST DEFAULT CHARACTER SET utf8;
3. Create a MySQL user named lumindev5 with the following attributes:

   Password: `********`

   Default tablespace: `LUMIN5TEST`

   CREATE USER 'lumindev5'@'localhost' IDENTIFIED BY '********';

   GRANT ALL PRIVILEGES ON LUMIN5TEST.* TO 'lumindev5'@'localhost';

```
CREATE USER 'lumindev5'@'%' IDENTIFIED BY '********';
GRANT ALL PRIVILEGES ON LUMIN5TEST.* TO 'lumindev5'@'%';
```

Liferay license key requirements

Beginning with Luminis 5.2, a license key is required to run the portal. Liferay EE GA2 requires this key for activation.

About this task

Note: If you requested and applied the Liferay license key in a previous release, you do not need to do so again.

Procedure

1. To request the license key from Ellucian®, open a case in Ellucian Client Support, and enter this text for the Subject:

   `Luminis 5.3 License Key Request`
2. Before installing Luminis 5.3, complete these steps to create a `setup.properties` file:
   a) Edit the `setup.properties` file.
   b) Add the `liferay.license.location` property.
   `liferay.license.location=/home/lpadmin/liferaylicensefile.xml`
   c) Save the changes.

   If for any reason the Liferay license file needs to be updated after Luminis Platform 5.2.0 has been installed, you can install the License key by placing the license key file into the `Liferay_deploy` directory.

   The Liferay deploy directories can be found here:

   - Administration Portal

     `$CP_ROOT/products/liferay/liferay-admin/deploy`
   - Student Portal

     `$CP_ROOT/products/liferay/liferay-portal/deploy`
3. To create a `setup.properties` file, see “Customize the installation values.” You must include a property to include the path to the license key within your Luminis deployment.
4. Add install property, `liferay.license.location`, to the `setup.properties` file before installing Luminis 5.3. This property tells the install process where to locate the license key file.

Related Links

Apply the 5.3 Patch Upgrade on page 66
Customize the installation values on page 13

Install the Luminis Platform system

The servers you install the Luminis Platform software on must not have previously had any Luminis Platform 4 or iPlanet/Sun products installed on them.

Old components, even if they have been uninstalled, can cause the installation to fail or the subsequent operation of the server and its components to be compromised.

Installing the Luminis Platform software may require installing software on several servers.

Minimum requirements for installation

Read and observe these requirements before you begin your install.

- All servers hosting Luminis Platform components should have a newly installed OS or VMware image with a suggested minimum 25 GB of storage and 4 GB of memory on the supported operating system:
  - Red Hat 5.x or 6.x
  - Solaris SPARC 10.x

    If Luminis Platform has been previously installed, follow the steps outlined in "Install the Luminis Platform system", or return to a pre-install snapshot of the OS and or VM.

    Note: If you run a single-box server deployment, ensure you have at least 2 GB of system memory for each tier instance or limit heap space with the `setup.properties` file. The minimum size of the Admin and portal servers should be 500 MB.
- The Luminis Platform LDAP server must be running. The Luminis Platform LDAP tier server must be installed and started before other tiers are installed. This is not necessary in a single-box installation, since the installer will install and start the LDAP server during the installation process.
- Installing Portal tiers. Portal tier Web servers must be installed one at a time. Do not attempt to simultaneously install Portal tiers on multiple servers.

If you plan on running one or more of the tiers using privileged ports (referring to ports lower than 1024) you must install Luminis Platform as root. If you want to install using non-privileged ports, see "Install Luminis Platform as Non-root User using Linux and Solaris."

Installation tasks

To complete the installation:

About this task

Note: In these instructions, the download directory is referred to as `$HOME`

Procedure

1. Download the Luminis Platform executable specific to your OS platform.

   There are 32-bit and 64-bit versions for Linux and Solaris. The 32-bit version can be installed on a 32-bit server or a 64-bit server, but an error occurs if you attempt to install a 64-bit version on a 32-bit server.

   - Linux: LP-5.3.0.0.XXXX-linux-32bit (64bit)
   - Solaris: LP-5.3.0.0.XXXX-solaris-32bit (64bit)
2. Create one or more `setup.properties` files containing configuration values specific to your installation.
3. If you are installing on Solaris or Linux and want the option to startup the Luminis Platform servers as a non-root user:

   a) Add a new user and group using the Unix `useradd` and `groupadd` commands.

   A user name might be lp5user and a group user might be lp5group.

   b) Add the `system.user` and `system.group` properties to the `setup.properties` files.

   The properties might display as`system.user=lp5user` or `system.group=lp5group`
4. Run the installer executable with the path to your `setup.properties` file.

   - On Linux, run the following command from `$HOME`:

     ```
     ./LP-5.3.0.0.XXXX-linux-32bit /<path>/setup.properties
     ```
   - On Solaris, run the following command from `$HOME`:

     ```
     ./LP-5.3.0.0.XXXX-solaris-64bit /<path>/setup.properties
     ```

     Note: Solaris requires a compatibility pack. The SunOS package `SUNWxcu4` should install `/usr/xpg4/bin/df` binary, which supports deprecated `df` command syntax used within Luminis Platform 5 installer on Solaris.

     Note: If SELinux is not properly configured, it can cause issues during the Luminis 5.3 installation or patch. SELinux can be disabled by setting SELINUX=disabled in the `/etc/selinux/config` file.
5. Source the `.cprc` file found in `$CP_ROOT`

   `$CP_ROOT` is the root directory specified by the `cp.root` property in `setup.properties`.

   ```
   cd $CP_ROOT . ./.cprc
   ```

Results

Luminis Platform is now installed. You can start up the servers with the `lpstart` command.

Related Links

Create custom configuration values on page 26
Customize the installation values on page 13
Luminis Platform System Basics on page 56

Customize the installation values

You can customize the values for your installation by editing your `setup.properties` file.

Installation values are delivered in the `default.properties` file. For more information about editing the `setup.properties` file, see “Luminis Platform installation configuration with setup.properties”.

Do not change `default.properties` directly.

Table 1: Default Installation Values

Property	Default Value	Description
cp.root	<no value>	The root directory where Luminis Platform is installed. This is a required value.
luminis.database.type	<no value>	The database type used for data storage in Luminis Platform. For example, supported values in Luminis Platform are oracle and mysql (values must be entered in lowercase). This is a required value.
luminis.database.host	<no value>	The fully qualified hostname where the database is installed. This is a required value.
luminis.database.name	<no value>	The name of the database used for Luminis Platform. This is a required value.
luminis.database.port	<no value>	The port number of the database used for Luminis Platform. This is a required value.
luminis.database.user.id	<no value>	The database user's identification. This is a required value.
luminis.database.user.password	<no value>	The database user's password. This is a required value.
luminis.database.driver	<no value>	The driver used to access the database information. For example, the driver may appear as follows: oracle.jdbc.OracleDriver This is a required value.

Property	Default Value	Description
luminis.database.dialect	<no value>	The dialect used to access the database information. Luminis Platform requires org.hibernate.dialect.MySQL5Dialect for MySql databases, and org.hibernate.dialect.Oracle10gDialect for Oracle databases.
luminis.database.url	<no value>	The Java database connectivity URL to a SQL database used by Liferay portals, and the defaults to the embedded Hypersonic database. For example: `jdbc:oracle:thin:@%{luminis.database.host}:%{luminis.database.port}:%{luminis.database.name}` This is a required value.
school.name	<no value>	The name of the college or university, such as Wasatch University. This is a required value. This value cannot be changed after installation.
school.abbrev	<no value>	An abbreviation for the name of the college or university, such as WU. This is a required value. This value cannot be changed after installation.
school.city	<no value>	The city where the college or university is located. This is a required value. This value cannot be changed after installation.
school.state	<no value>	The state where the college or university is located, such as Utah. This is a required value. This value cannot be changed after installation.
school.country	<no value>	The country where the college or university is located, such as US. This is a required value.

Property	Default Value	Description
		This value cannot be changed after installation.
school.web.id	<no value>	The Web identifier and domain name for the school. This value is used as a suffix for e-mail addresses, such as wasatch. edu. This is a required value. This value cannot be changed after installation.
tiers	ldap,cas,admin,portal	A comma-separated list of tiers to be installed. The default value is all the tiers, which essentially installs everything on a single server.
default.language.code	<all possible values>	The language code for the default supported language. For example, enter en for English.
default.country.code	<all possible values>	The country code for the default supported language. For example, enter US for the United States.
supported.locales	%{default.language.code}_ %{default.country.code}	A comma-delimited list of all the supported locales, such as supported. locales=en_US,fr_FR,es_MX,ar_SA,pt_BR.
user.timezone	<no value>	Default timezone to be used by the Luminis/Liferay system. Values must follow the Java standard timezone ID, for example, America/ Los_Angeles.
local.host	Auto-detected	The fully qualified hostname, automatically-detected by the installer is determined through the hostname command. This property can be overridden for virtual servers.
ldap.host	%{local.host}	The LDAP hostname.
ldap.port	389	The LDAP port.
ldap.secure.port	636	The LDAP secure port.
ldap.base.dn	o=cp	Luminis Platform's base DN.

Property	Default Value	Description
ldap.directory.manager.dn	cn=Directory Manager	The directory manager's DN.
ldap.directory.manager.password	cp.admin	The directory manager's password.
ldap.directory.person.id.attribute	employeeNumber	The directory attribute for the immutable person identifier. Luminis Platform assumes the directory attribute will contain one value and is of the Directory String syntax. If you use a Multi Value directory attribute, Luminis Platform only uses the first value that is returned.
ldap.directory.udc.id.attribute	departmentNumber	Directory attribute for the Banner UDC identifier. Luminis Platform assumes the directory attribute will contain one value and is of the Directory String syntax. If you use a Multi Value directory attribute, Luminis Platform only uses the first value that is returned.
ldap.directory.type	opendj	The type of directory service Luminis Platform will use. Accepted values are opendj and activedirectory.
ldap.directory.user.dn	ou=people	Directory attribute where users are placed in the directory tree. Luminis Platform will concatenate this configuration value to the `ldap.base.dn` configuration value to build a Distinguished Name used to search for users in the directory.
admin.id	admin	The username for the default administrative user. This user profile is created automatically during initial system installation.

Property	Default Value	Description
admin.password	admin	The password for the default administrative user.
jmx.admin.id	%{admin.id}	The Java Management Extension (JMX) administrative user identification. This value can be overridden.
jmx.admin.password	%{admin.password}	The JMX administrative password. This value can be overridden.
admin.jmx.port	9002	The port number to use for the JMX server on the Admin tier.
admin.jmx.rmi.server.host	localhost	Hostname to use for JMX/RMI server connection for the admin server. See the section on connecting to a JMX server through a firewall in the *Luminis Administrator's Guide*.
portal.jmx.port	9001	The port number to use for the JMX server on the Portal tier.
portal.jmx.rmi.server.host	localhost	Hostname to use for the JMX/RMI server connection for the portal server. See the section on connecting to a JMX server through a firewall in the *Luminis Administrator's Guide*.
security.authentication.provider	cas	Determines whether Luminis uses CAS or SAML as the authentication protocol for logging in users.
cas.host	%{local.host}	The fully qualified hostname where the CAS server is running. This value can be overridden.
cas.http.port	8090	The CAS server port. This value can be overridden.
cas.https.port	8447	The CAS server secure web port.
cas.webapp.path	cas-web	The webapp path for the CAS server. This is the part of the URL after the host:port name, as in `https://<cashost>/`

Property	Default Value	Description
		[bold]cas-web[/bold]/index.jsp.
cas.auth.ldap.host	%{ldap.host}	The LDAP server CAS uses to authenticate user login requests. This can be an external LDAP server. Clustered hosts are a custom configuration requiring a Services engagement.
cas.auth.ldap.port	%{ldap.port}	The LDAP port for CAS authentication.
cas.auth.ldap.base.dn	%{ldap.base.dn}	The base DN for the LDAP used for CAS authentication.
cas.auth.ldap.directory.manager.dn	%{ldap.directory.manager.dn}	The Directory Manager DN for the LDAP used for CAS authentication.
cas.auth.ldap.directory.manager.password	%{ldap.directory.manager.password}	The Directory Manager password for the LDAP used for CAS authentication.
cas.auth.ldap.search.attribute	Uid	The LDAP attribute to search for when using CAS to authenticate.
ldap.connect.timeout	100	The CAS value used to recycle stale LDAP connections. This value is used with the `ldap.read.timeout` property.
ldap.read.timeout	100	The CAS value used to timeout the attempt to read from `cas.auth.ldap.host`.
saml.idp.host.protocol	https	Web protocol to communicate with the SAML server: HTTP or HTTPS.
saml.idp.host.name	[none]	Name of the SAML authentication server.
saml.idp.host.port	[none]	Port to use when connecting to the SAML server.
saml.idp.entityid	[none]	The IDP entity ID configured in the SAML server to be used by Luminis.

Property	Default Value	Description
saml.idp.login.service.path	/samlsso	URL path to be appended to the server name and port for sending authentication assertions to the SAML server for login.
saml.idp.logout.service.path	/samlsso	URL path to be appended to the server name and port for sending authentication assertions to the SAML server for logout.
saml.sp.admin.entityid	[none]	Service provider Entity ID configured on the SAML server to be used by the Luminis admin server.
saml.sp.portal.entityid	[none]	Service provider Entity ID configured on the SAML server to be used by Luminis portal servers.
saml.keystore.filename	%{cp.root}/.keystore	Keystore file to be used for storing certificates used by SAML.
saml.keystore.password	%{java.keystore.password]	Password for the keystore specified above.
saml.encryption.certificate.alias	tomcat	Alias of certificate to be used for encrypting SAML assertions between Luminis and the SAML server.
saml.encryption.certificate.password	%{java.keystore.password}	Password for the certificate alias specified in `lams.encryption.certificate.alias`.
saml.signing.certificate.alias	tomcat	Alias of the (optional) certificate to be used for signing SAML assertions and responses.
admin.host	%{local.host}	The fully qualified hostname where the Admin server is installed.
admin.http.port	8080	The Admin server port.
admin.https.port	8443	The Admin server secure Web port.
portal.virtual.host	%{local.host}	The fully qualified Web name used to access the Luminis

Property	Default Value	Description
		Platform portal system. This is the name used from the browser to enter the Luminis Platform portal, whether directly to the installed portal or through a load balancer.
portal.virtual.http.port	80	The HTTP port to access the Luminis portal via a non-secure browser request. This port is the load balancer non-SSL port in a load-balanced system.
portal.virtual.https.port	443	The HTTPS port to access the Luminis portal via a secure browser request. This port is the load balancer SSL port in a load-balanced system.
portal.http.port	80	The HTTP non-secure port of the portal server.
portal.https.port	443	The HTTP secure port of the portal server.
http.proxy.host	<no value>	The HTTP proxy host used to navigate HTTP traffic away from an internal network.
http.proxy.port	<no value>	The HTTP proxy port used to navigate HTTP traffic away from an internal network.
https.proxy.host	<no value>	The Secure proxy host.
https.proxy.port	<no value>	The Secure proxy port.
import.test.users	yes	The value determines whether a default administrative user will be imported into LDAP during the installation. Leave the value as default Yes if you are using the Luminis internal directory. If you use an external directory and do not want Luminis to create an admin user for Luminis Platform, set the value to No.
java.keystore.file	%{cp.root}/.keystore	Luminis Platform's keystore file. For more information about the Java keystore, see “Install and Configure external CAS.”

Property	Default Value	Description
java.keystore.password	changeit	Luminis Platform's keystore file password. The default is the Sun changeit password. For more information about the Java keystore, see “Install and Configure external CAS.”
admin.min.heap	1024m	The Admin server minimum memory heap size.
admin.max.heap	2048m	The Admin server maximum memory heap size.
portal.min.heap	1024m	The Portal server minimum memory heap size.
portal.max.heap	2048m	The Portal server maximum memory heap size.
system.group	root	The system group name. The group with permission to start and stop Luminis Platform servers.
system.user	root	The system user name that administers Luminis Platform. Specifying the system user and system group sets all file ownership to this user and group.
root.access.required	yes	An indicator to specify whether or not the installed system requires root or administrator access to run. If set to no then all ports must be set to non-privileged ports, such as >1024.
http.nonproxy.hosts	%{local.host}	Non-proxy hosts are those that should have a direct HTTP connection, instead of connecting through the proxy server.
https.nonproxy.hosts	%{local.host}	This host is similar to `http.nonproxy.hosts`.
tmp.dir	%{cp.root}/tmp	The name of a temporary directory used by some Java processes. This name is used to set the `java.io.tmpdir` command-line parameter for Java execution.

Property	Default Value	Description
enable.google.saml.sso	no	A flag to indicate whether Google SSO should be configured in the CAS server. If you set the value as yes, the Google SSO is configured in the CAS server. Any value other than yes is interpreted as no.
path.to.private.key	<none>	The full path to the private key file on the Luminis Platform file system. The installation process copies this file to the `WEB-INF/classes` directory for the CAS server during installation. For more information about setting up an external CAS server, see “Install and Configure external CAS.”
private.key.filename	dsaprivkey.der	The name of the private key file.
key.algorithm	DSA	The encryption algorithm used when the key files are generated. Possible values are RSA or DSA.
path.to.public.key	<none>	The full path to the public key file on the Luminis Platform file system. The installation process copies this file to the `WEB-INF/classes` directory for the CAS server during installation.
public.key.filename	dsapubkey.der	The name of the public key file.
jms.type	sun	The provider configuration type for the Java Message Service (JMS) connection to Luminis Message Broker (LMB).
jms.enabled	false	This determines whether to enable JMS. The two available values are true and false. If you set the value to true, JMS is enabled.
jms.host	<no value>	The JMS host name.

Property	Default Value	Description
jms.connection.url	http://%{jms.host}/imq/tunnel	The JMS connection URL.
jms.addressList	mq://%{jms.host}:7676/ssljms	The JMS address list.
jms.userid	admin	The JMS access user identification.
jms.password	cp.admin	The JMS password.
banner.sso.gateway	<no value>	The URL to the Banner® SSO Web proxy, which is part of the Banner Identity Gateway deployment.
cas.server.gateway	false	A property that determines whether the login screen should be displayed to the user. The two available values are true and false. If you set the value to true, the login screen is displayed to the user.
cas.server.renew	false	A property that determines whether users must log in to each application. The two available values are true and false. If you set the value to true, users must log into each application.
cas.server.url	https://%{cas.host}:%{cas.https.port}/%{cas.webapp.path}/	The URL to the CAS server. Generally, this is a secured container, meaning the URL should include HTTPS.
cas.server.proxyCallbackUrl	https://%{cas.host}:%{cas.https.port}/manager/proxy/Receptor	The server proxy callback URL, which should point to the CAS server and port.
cas.client.admin.serverName	%{admin.host}:%{admin.https.port}	The client server address, which consists of the fully qualified server name and port where the application is running. If you are running a secured application, the client server address should be the SSL port.
cas.client.portal.serverName	%{portal.virtual.host}:%{portal.https.port}	The client server address, which consists of the fully qualified server name and

Property	Default Value	Description
		port where the application is running. If you are running a secured application, this URL should be the SSL port.
cas.client.admin.proxyCallbackUrl	https://%{admin.host}:%{admin.https.port}/web/luminis-admin-group	The client proxy callback URL, which should point to the server and port where the client application is running. If you are running a secured application, this URL should be the SSL port.
cas.client.portal.proxyCallbackUrl	https://%{portal.virtual.host}:%{portal.https.port}/web/luminis	The client proxy callback URL, which should point to the server and port where the client application is running. If you are running a secured application, this URL should be the SSL port.
system.user	root	The user that has ownership of the Luminis Platform files after installation and is expected to be the one to start, stop, and generally manage the Luminis installation from the Operating System level.
system.group	root	The group that assigned to all Luminis Platform files after installation. This is the group that the Luminis system user is expected to belong to for managing Luminis.
jgroups.keystore.password	%{java.keystore.password}	The password for the keystore used by jgroups, which is the protocol layer used to transmit and receive multicast cache invalidation commands. A certificate is generated and stored for encrypting the cache invalidation traffic.
cache.distribution.mcast_address	230.0.0.1	Multicast address used for communication between the Luminis servers for cache

Property	Default Value	Description
		invalidation. Must be a valid multicast address (see IPv4 address space specifications) and must not conflict with any other multicast address user on the sub-net.
cache.distribution.mcast_port	4446	The port used by multicast cache invalidation. Must not conflict with other multicast address/ports used on the sub-net.
cache.distribution.auth_string	%{jgroups.keystore.password}	The authorization string used to access the keystore for the certificate used to encrypt multicast cache invalidation traffic.
cache.distribution. use_multicast	true	If the value is true, the multi-node cache update protocol uses IP multicasting to broadcast cache update information. If false, a point-to-point TCP protocol is used. Multicasting is more efficient, but may not be available on all networks, particularly in cloud-based infrastructure.
cache.distribution.tcp_port	4455	If the `cache.distribution. use_multicast` property value is false, this port is used for the TCP connection between nodes.
cacerts.password	changeit	Password for the Certificate Authority Certificate file that is used from the Java installation of Luminis 5.3.
deactivate.liferay.default.site	true	Disables the Liferay Guest site. The Liferay Guest site is enabled by default; the Luminis Platform 5.3 guest site is disabled. To make the guest site available, set this property to false and activate the Guest site from the **Site Details** option in the Liferay **Control**

Property	Default Value	Description
		Panel, then restart Luminis Platform.

Related Links

Pre-installation step on page 73
Site Analysis Worksheet on page 91
Install Luminis Platform on page 6
Site Analysis Worksheet on page 91
Installation tasks on page 11
Create custom configuration values on page 26
Luminis with SAML installation properties on page 36
Customize the default site names on page 41
Setup the admin tier index on page 53
Install and Configure external CAS on page 73
Liferay license key requirements on page 10
Set up Amazon S3 support in Luminis Platform on page 50
Prerequisites for Amazon S3 setup on page 51

Create custom configuration values

Specify the local, unique values for your Luminis Platform installation.

To create custom configuration values, you must create a `setup.properties` file to override the default values. When you install Luminis Platform, you are required to pass in the location of your `setup.properties` file. If you choose to implement a multi-server installation, you should create a different `setup.properties` file for each server.

These examples display different methods of how you can configure the `setup.properties` file for different types of installations. You can use these examples as a template and modify as needed.

Note: Ellucian recommends that you install CAS separately from other tiers if you want to access CAS from outside the private network, or if you manage SSO with external services, such as Google SSO or Office 365, and require CAS to reside in the DMZ.

Related Links

Installation tasks on page 11
Customize the installation values on page 13

1-Tier development installation with Oracle 10g/11g

Example of a `setup.properties` file you might use if you install all of your components on the same tier using the Oracle 10g or 11g database.

```
# This file contains values that override the default values (LP5).
# %%templatized-file%%
cp.root=/opt/lp5
tiers=ldap,cas,portal,admin
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
cas.host=cashost.domain.edu
admin.host=adminhost.domain.edu
portal.virtual.host=virtualhost.domain.edu
# Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

Related Links

Oracle requirements on page 8

1-Tier development installation with MySQL

Example of a `setup.properties` file you might use if you install all of your components on the same tier using the MySQL database.

```
# %%templatized-file%%
cp.root=/opt/lp5
tiers=ldap,cas,portal,admin
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
cas.host=cashost.domain.edu
admin.host=adminhost.domain.edu
portal.virtual.host=virtualhost.domain.edu
# Possible values are oracle/mysql
luminis.database.type=mysql
# Use the following for the MySQL DB
luminis.database.host=localhost
luminis.database.name=test
```

```
luminis.database.port=3306
luminis.database.user.id=root
luminis.database.user.password=root
luminis.database.driver=com.mysql.jdbc.Driver
luminis.database.dialect=org.hibernate.dialect.MySQL5Dialect
luminis.database.url=jdbc:mysql://%{luminis.database.host}:
%{luminis.database.port}/%{luminis.database.name}?
useUnicode=true&amp;characterEncoding=UTF-8
```

Separate portal 2-tier install

Example `setup.properties` file if you install the LDAP, CAS, and Admin servers on one tier, and the Portal server on a second tier.

About this task

In this instance, install the Admin tier first and run the `startup` command to start it. Then install the Portal tier.

The following example is the `setup.properties` file for the Admin tier, which includes LDAP, CAS and Admin servers.

To complete a two-tier Luminis Platform installation, with Portal on a separate tier:

Procedure

1. Install the Admin tier.

   In this example setup.properties file, the Admin tier includes the LDAP and CAS servers:

```
# This file contains values that override the default values (LP5).
# %%templatized-file%%
cp.root=/opt/lp5

tiers=ldap,cas,admin

school.name=Wasatch University

school.state=Utah

school.city=Salt Lake City

school.country=US

school.abbrev=SU

school.web.id=wasatch.edu

ldap.host=ldaphost.domain.edu

cas.host=cashost.domain.edu

admin.host=adminhost.domain.edu
```

```
portal.virtual.host=virtualhost.domain.edu

# Possible values are oracle/mysql

luminis.database.type=oracle

# Use the following for the ORACLE DB

luminis.database.host=luminishost.sct.com

luminis.database.name=DEV

luminis.database.port=1521

luminis.database.user.id=dbuser

luminis.database.user.password=dbuser

luminis.database.driver=oracle.jdbc.OracleDriver

luminis.database.dialect=org.hibernate.dialect. Oracle10gDialect

luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

2. Source the `.cprc` file and start the server using the `lpstart` command.
3. Install the Portal tier.

   A sample setup.properties file:

```
# This file contains values that override the default values (LP5).

# %%templatized-file%%

cp.root=/opt/lp5

tiers=portal

school.name= Wasatch University

school.state=Utah

school.city=Salt Lake City

school.country=US

school.abbrev=SU

school.web.id=wasatch.edu

ldap.host=ldaphost.domain.edu

cas.host=cashost.domain.edu
```

```
admin.host=adminhost.domain.edu

portal.virtual.host=virtualhost.domain.edu

# Possible values are oracle/mysql

luminis.database.type=oracle

# Use the following for the ORACLE DB

luminis.database.host=luminishost.sct.com

luminis.database.name=DEV

luminis.database.port=1521

luminis.database.user.id=dbuser

luminis.database.user.password=dbuser

luminis.database.driver=oracle.jdbc.OracleDriver

luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect

luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%(luminis.database.name}
```

4. Source the `.cprc` file and start the server using the `lpstart` command.

Separate CAS 2-tier install

In this example, the Admin tier contains the OpenDJ/LDAP, Admin, and Portal servers. The CAS tier contains only the CAS server.

About this task

To complete a two-tier Luminis Platform installation with CAS on a separate tier:

Procedure

1. Install the Admin tier.

   In this example `setup.properties` file, the Admin tier is luminishost12h.sct.com, and the CAS tier is luminishost12f.sct.com.

```
 # This file contains values that override the default values (LP5).
# %%templatized-file%%
cp.root=/opt/lp5
tiers=ldap,portal,admin
 school.name=Wasatch University
 school.state=Utah
 school.city=Salt Lake City
 school.country=US
 school.abbrev=SU
```

```
school.web.id=wasatch.edu
cas.host=cashost.domain.edu
# Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

2. Source the `.cprc` file and start OpenDJ.

```
. <cp.root>/.cprc
10-ldap start
```

3. Install the CAS tier.

   In this example `setup.properties` file, the Admin server is luminishost12h.sct.com, and the CAS tier is luminishost12f.sct.com.

```
# This file contains values that override the default values (LP5).
# %%templatized-file%%
cp.root=/opt/lp5
tiers=cas
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

4. On the CAS tier, source the `.cprc` file and start the CAS server using the `lpstart` command.

```
. <cp.root>/.cprc
```

5. Start the remaining servers on the Admin tier using the `lpstart` command.

Separate Portal and separate Admin 3-tier install

Example three-tier installation with the Admin server and the Portal server each on its own tier, and a Resource tier containing the LDAP and CAS servers.

Procedure

1. Install the Resource tier which includes LDAP and CAS servers.

   In this example `setup.properties` file, the Admin tier is luminishost12h.sct.com, the Portal tier is luminishost12f.sct.com, and the Resource tier is luminishost12m.sct.com.

   Ellucian recommends that you install CAS separately from other tiers if you want to access CAS from outside the private network, or if you manage SSO with external services, such as Google SSO or Office 365, and require CAS to reside in the DMZ.

   ```
   # This file contains values that override the default values (LP5).
   # %%templatized-file%%
   cp.root=/opt/lp5
   tiers=ldap,cas
   school.name=Wasatch University
   school.state=Utah
   school.city=Salt Lake City
   school.country=US
   school.abbrev=SU
   school.web.id=wasatch.edu
   ldap.host=ldaphost.domain.edu
   cas.host=cashost.domain.edu
   admin.host=adminhost.domain.edu
   portal.virtual.host=virtualhost.domain.edu
   luminis.database.name=DEV
   luminis.database.port=1521
   luminis.database.user.id=dbuser
   luminis.database.user.password=dbuser
   luminis.database.driver=oracle.jdbc.OracleDriver
   luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
   luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
   %{luminis.database.port}:%{luminis.database.name}
   ```

2. On the Resource tier, source the `.cprc` file and start all servers using the `lpstart` command.

   ```
   . <cp.root>/.cprc
   ```

3. Install the Admin tier, which contains only the Admin server.

   In this example `setup.properties` file, the Admin tier is luminishost12h.sct.com, the Portal tier is luminishost12f.sct.com, and the Resource tier is luminishost12m.sct.com.

   ```
   # This file contains values that override the default values (LP5).
   # %%templatized-file%%
   cp.root=/opt/lp5
   tiers=admin
   school.name=Wasatch University
   school.state=Utah
   ```

```
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
cas.host=cashost.domain.edu
portal.virtual.host=virtualhost.domain.edu
 # Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

4. Source the `.cprc` file and start the server using the `lpstart` command.
5. Install the Portal tier, which contains only the Portal server.

   In this example setup.properties file, the Admin tier is luminishost12h.sct.com, the Portal tier is luminishost12f.sct.com, and the Resource tier is luminishost12m.sct.com.

```
# This file contains values that override the default values (LP5)
# %%templatized-file%%
cp.root=/opt/lp5
tiers=portal
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
cas.host=cashost.domain.edu
admin.host=adminhost.domain.edu
# Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

6. On the Admin tier, source the `.cprc` file and start the Admin server using the `lpstart` command.

```
. <cp.root>/.cprc
```

7. On the Portal tier, source the `.cprc` file and start the Portal server using the `lpstart` command.

```
. <cp.root>/.cprc
```

2-Tier install with external LDAP for CAS authentication

Example to demonstrate the setup up for a Luminis Platform install where the CAS server uses a pre-installed external LDAP server for authentication.

The Admin tier contains all of the other components. For instructions on installing an external LDAP server, see "External Directory Server (LDAP) SunONE DS Setup."

A sample `setup.properties` file is shown below. In this example, the Luminis Platform tier hostname is everything.wasatch.edu and the tier with the external LDAP installed and running is extldap.wasatch.edu.

```
# This file contains values that override the default values (LP5).
# %%templatized-file%%
cp.root=/opt/lp5
tiers=ldap,cas,portal,admin
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
ldap.host=ldaphost.domain.edu
cas.host=cashost.domain.edu
admin.host=adminhost.domain.edu
portal.virtual.host=virtualhost.domain.edu
cas.auth.ldap.host=extldap.wasatch.edu
#port number for external ldap
cas.auth.ldap.port=389
#base DN for external ldap
cas.auth.ldap.base.dn=o=Wasatch
#external  ldap attribute to search.  (The user's login id)
cas.auth.ldap.search.attribute=loginId
#external ldap manager DN
cas.auth.ldap.directory.manager.dn=cn=Directory Manager
#external ldap manager password
cas.auth.ldap.directory.manager.password=*****
# Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
```

```
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
```

2-Tier Install with SAML Authentication and SSO

You can configure Luminis Platform to use SAML instead of the CAS server as the authentication protocol.

This section describes how to install Luminis Platform with the Ellucian Identity Server (EIS) as the SAML authentication provider. Other SAML-compatible servers require similar configuration operations, but may vary in detail. You are expected to understand your particular SAML-enabled security product and configure it as appropriate to function with Luminis.

Ellucian Identity Server (EIS) configuration

Luminis requires that you configure two service providers on the EIS server, one for the admin server and one for the portal. If there is more than one portal behind a load balancer, configure the portal service provider to service the portal virtual host name at the load balancer.

About this task

To create a new service provider in EIS:

Procedure

1. Log in as an administrator.
2. Navigate to **Main** > **Identity column** > **Service Providers**.
3. Click **Add**.
4. Enter a name and description.
5. Under Basic Information, select **Inbound Authentication Configuration** and the **SAML2 Web SSO Configuration**.
6. Click **Configure**.
7. On the **Register New Service Provider** page:
   a) Enter an **Issuer** name. This name is used in Luminis to configure the service provider EntityID.
   b) Use this form for the **Assertion Consumer URL**:

      `https://<your.luminis.server>:<port>/c/portal/login`

   c) Check the **Enable Single Logout** check box. Use this form for the **Custom Logout URL**:

      `https://<your.luminis.server>:<port>/c/portal/singlelogout`.

Make sure the *<your.host.name>* matches (including sub-domain) the value you use in the `setup.properties` file for `portal.virtual.host` for the portal, or `admin.host` for the admin server SAML service provider.

d) **Optional:** Check the **Enable Response Signing**, **Enable Assertion Signing**, and **Enable Assertion Encryption** check boxes to enable additional security.

8. Click **Update** to save your configuration.
9. To configure the Identity Provider in EIS to accept SAML sign-on requests:
   a) Navigate to **Identity Providers** > **Identity** > **List** > **Resident Entity Provider**.
   b) Change the **Home Realm Identifier** to `https://<your_saml_host.yourdomain>:8447/samlsso`
   c) Select the **Inbound Authentication Configuration** bar and then open **SAML2 Web SSO Configuration**.
   d) Enter a name for your SAML entity ID and save it to be entered in the Luminis setup properties for the `saml.idp.entityid` value.
   e) Save the changes.

Luminis with SAML installation properties

Install Luminis Platform with the installation executable and the `setup.properties` file as listed in "Customize the installation values."

In this example `setup.properties` file, the Luminis Platform tier hostname is everything.wasatch.edu and the SAML server is saml.wasatch.edu:

```
cp.root=/opt/lp5
tiers=ldap,admin,portal
school.name=Wasatch University
school.state=Utah
school.city=Salt Lake City
school.country=US
school.abbrev=SU
school.web.id=wasatch.edu
# Possible values are oracle/mysql
luminis.database.type=oracle
# Use the following for the ORACLE DB
luminis.database.host=luminishost3.sct.com
luminis.database.name=DEV
luminis.database.port=1521
luminis.database.user.id=dbuser
luminis.database.user.password=dbuser
luminis.database.driver=oracle.jdbc.OracleDriver
luminis.database.dialect=org.hibernate.dialect.Oracle10gDialect
luminis.database.url=jdbc:oracle:thin:@%{luminis.database.host}:
%{luminis.database.port}:%{luminis.database.name}
security.authentication.provider=saml
saml.idp.host.name=saml.wasatch.edu
saml.idp.host.port=443
saml.idp.entityid=WASATCH_U_IDP
```

```
saml.idp.admin.entityid=<your_saml_service_provider_admin_name>
saml.sp.portal.entitid=<your_saml_service_provider_portal_name>
```

Related Links

Customize the installation values on page 13

Configure Luminis Platform tiers using a load balancer

How to configure the Luminis Platform tiers using a load balancer with node-to-node encryption or SSL termination, and how to configure the load balancer to monitor each group of servers to fail over to available portal(s).

The load balancer acts as the externally visible host and must be configured to pass through Luminis-related requests to the appropriate servers. Multiple Luminis Portal servers may be installed and the load balancer configured to serve requests to all servers based on rules for balancing and fail-over. Please see the documentation for your load balancer to determine how to implement the appropriate balancing and fail-over policies for your installation.

Related Links

Portal Server on page 55

Configure the Luminis Platform installation properties file

You can configure the CAS server to operate behind the load balancer, or be visible externally separate from the load balancer. You can also configure the Admin server to run behind the load balancer.

About this task

Luminis Platform must be configured to recognize the externally visible host name for the installation. Clustering the CAS or Admin tiers behind the load balancer is not supported, although they may be accessed singly through the load balancer. Additionally, when installing on multiple nodes the Admin tier must be installed before all Portal tiers.

To configure Luminis Platform to operate behind a load balancer, complete these steps before running the installation:

Procedure

1. Navigate to the `setup.properties` file.
2. Set the value of the `portal.virtual.host` property to the load balancer's Web name.

   Set the `portal.virtual.http.port` to the port that users will access the server via non-SSL connections (usually port `80`). Set the `portal.virtual.https.port` to the secured port that will be accessed through the load balancer (usually `443`).
3. If you want the load balancer host name to be the host name of the Admin server, set the value of the `admin.host` property to the load balancer host name.

   Be sure to select a different port than the Portal server for the Admin server HTTP and HTTPS ports.

4. To allow direct access to the CAS server, set `cas.host` to the name of the CAS server.

   Note: If you want the load balancer to act as the front end to your CAS server, set the `cas.host` property to the host name of your load balancer. You should also setup the load balancer to forward all traffic coming in on the CAS ports to the CAS server, as described in the following steps.

5. To specify the load balancer as the front end to the CAS server, set `cas.host` to the load balancer name, and set `cas.local.host` to the name of the physical CAS server. Be sure to select unique port numbers for the HTTP and HTTPS CAS ports.
6. Set the LDAP host name, `ldap.host`, to the LDAP server. This host can sit behind your firewall because end users cannot see it.

Related Links

Configure the load balancer for node-to-node encryption on page 39

Configure the load balancer for node-to-node encryption

The load balancer should be configured to send unencrypted server HTTP requests to the HTTP port on the Portal servers, and to serve encrypted SSL requests to the HTTPS port on the Portal servers in your cluster.

About this task

These requests should navigate to the respective ports on the back-end servers. When Luminis Platform, as configured out-of-the-box, receives an unencrypted HTTP request, the server automatically redirects the browser to re-send the request via HTTPS. In effect, the load balancer receives an HTTPS request, decrypts the request to determine which back-end server to send it to after the session has been “stickied” or assigned to a specific back-end Portal server, and re-encrypts the request to send to the back-end server. All Luminis Platform traffic on the back end navigates through SSL in this configuration. This secure configuration allows safety within and outside of the firewall.

Procedure

1. Set up the load balancer to flow through all of the Portal servers you have in your portal cluster. The flow works for the HTTP requests and the HTTPS requests.
2. Set up the load balancer to stick the HTTPS session using cookies.

   Note: It is not necessary to sticky the HTTP requests. HTTP requests are automatically forwarded to HTTPS.

3. In environments where the CAS server resides behind the load balancer and CAS validation occurs on a public interface such as the load balancer external interface, change the `cas.local.host` property within the ticketValidator bean configuration to `cas.host`.

   ```
   <bean id="ticketValidator"
   class="org.jasig.cas.client.validation.Cas20ServiceTicketValidator">
   ```

   ```
   <constructor-arg value="https://${cas.host}:${cas.https.port}/
   ${cas.webapp.path}"/>
   ```

```
<property name="proxyCallbackUrl" value="https://${admin.host}:
${admin.https.port}/proxy/receptor"/>
```

The `cas.local.host` property is located in the `tomcat-<node>/webapps/ROOT/META-INF/luminis-liferay-securityContext.xml` folders for the Admin and Portal tiers.

Note: The default implementation of CAS server included with Luminis installer assumes the hostname used within `cas.local.host` is accessible to CAS client implementations such as Portal node or Admin node.

4. If you want to run CAS behind the load balancer, ensure that the load balancer forwards all traffic from the CAS ports to the CAS server.
5. If you want to run the Admin server behind the load balancer, ensure that the load balancer forwards all traffic to both the HTTPS and HTTP Admin ports to those respective ports on the Admin servers.

   How requests should be handled by a load balancer at this location:

   `luminis.edu`

   You have two portal servers running behind the load balancer, `lum1.edu` and `lum2.edu`. CAS is running through the load balancer as well, but CAS is referenced as `cas.luminis.edu`. The `lum1.edu:80` and `lum2.edu:443` ports are configurable:

   `browser > HTTP://luminis.edu > lum1.edu:80`

   Luminis Platform re-directs the browser to the following browser:

   `https://luminis.edu > lum2.edu:443`

Results

The load balancer should sticky the session so that all other requests from this browser navigate to the `lum2` server. The user remains on `lum2.edu` for the remainder of the session.

Luminis Platform re-directs the browser to CAS for authentication.

`browser > https://cas.luminis.edu`

You can configure CAS to sit behind the firewall and your load balancer to forward all requests that go to the load balancer on the CAS ports to the CAS server.

When the user successfully authenticates with CAS, the user returns to the browser, as follows:

`https://luminis.edu -> lum2.edu:443`

Configure the load balancer for node-to-node encryption

Configure Luminis Platform to allow SSL termination at the load balancer so that all requests between the load balancer and the Portal servers are HTTP rather than HTTPS.

Additionally, it is highly recommended that you configure the load balancer to redirect all incoming HTTP traffic to the HTTPS ports on the load balancer to guarantee that all transactions are protected by SSL encryption. In this configuration, Luminis Platform does not check whether requests that should normally be done via secure channel are in fact secured. Many load balancers have sophisticated programming mechanisms to allow partitioning the URL space for required

HTTPS access. However, the most simple mechanism is to redirect all HTTP requests to the HTTPS port on the load balancer.

For SSL termination at the load balancer, configure the Luminis platform installation `setup.properties` file. There will be a post-installation step required to allow Luminis Platform to accept requests that are typically secure on the non-secure HTTP port.

The Load balancer should be configured to pass all requests to the HTTP or HTTPS ports to the HTTP port on the Luminis Portal servers, and the configured HTTP and HTTPS ports to the HTTP port on the Luminis Admin server. Requests for CAS may also be configured to be redirected to the non-secure port if CAS is also being served through the load balancer. CAS will respond to either type of request.

Related Links

Configure the Luminis Platform installation properties file on page 37

Post-installation configuration

After you install Luminis Platform, change two configuration values to allow access to most URLs through the non-SSL port of the Portal servers.

For instructions on how to set configuration values on the running system, see the "Configuration Management with JMX" section in the *Luminis Platform Administration Guide*.

- Set `security.ssl.loadbalancer.termination.portal` to true for all Portal servers.
- Set `security.ssl.loadbalancer.termination.admin` to true on the Admin server.

Configure the fail over option

If the lum2 server is unavailable for any reason, the load balancer should be configured to fail the user over to the lum1 server.

The most common setup is to monitor each group of servers. This monitor usually attempts to grab a known resource from the server. You should set up a monitor for both the HTTP group and the HTTPS group.

For the resource to grab, you can use the `/luminis/images/misc/arrow.gif` file.

This is a small graphic file that indicates that the operation was successful. Resources other than a PNG or other small graphic redirect you to the HTTPS version of that resource.

Note: If the server the user is stickied to goes down, the load balancer redirects the user to a different server. When that user makes a RESTful (representational state transfer) service call to that new server, the browser asks the user to re-authenticate via CAS. If the RESTful request fails, a non-restful request to the new server automatically re-authenticates the user using CAS and the request proceeds transparently as if the user were still on the original server. This is called a transparent fail over. After the user has experienced transparent fail over, the RESTful services resume their proper function. For more information about REST, see the *RESTful Web Service Design and Development Guide*.

Customize the default site names

When Luminis Platform is installed, you can customize the default site names.

Default site and URLs:

- Luminis Administrators Community

  Site URL = /luminis-admin-group

- Home Community

  Site URL = /home-community

These properties are grabbed by Liferay in the `portal-ext-admin/portal-ext-portal.properties` file via:

```
community.default.admin=%{luminis.admin.community}
community.default.home=%{luminis.home.community}
```

To change the default settings before installation, set the properties below to the desired values in the `setup.properties` file.

For example:

```
luminis.admin.community=Wasatch Admin Site
luminis.admin.community.url=/wasatchadminsite
luminis.home.community=Wasatch Home Site
luminis.home.community.url=/wasatchhomesite
```

For information on how to change the default settings post-installation, see "Change the default site name" in the *Luminis Platform Administration Guide*.

Related Links

Customize the installation values on page 13

Test the installation

After you finish installing Luminis Platform servers, review the installation logs that are generated by the install program and stored in the system.

You should also run a variety of test cases that verify the success of the installation and health of the various components you have installed.

The following sections provide information about the installation logs that you should review and recommendations for setting up and executing installation test cases. If you notice serious anomalies in the logs or experience failures during testing, you should contact Ellucian Client Support.

Start up tiers

Ensure that all tiers start up successfully.

Before you start up the Luminis Platform tiers, source the `.cprc` file found in `$CP_ROOT`. `$CP_ROOT` is the root directory specified by the `cp.root` property in `setup.properties`.

```
cd $CP_ROOT
. ./.cprc
```

Start up all Luminis Platform components on a server in proper order with the `lpstart` command and shutdown all Luminis Platform components on a server with the `lpstop` command.

These scripts are within the `$CP_ROOT/bin` directory:

Script	Description
10-ldap	Starts and stops the LDAP tier
20-cas-webserver	Starts and stops the CAS tier
25-admin-webserver	Starts and stops the administrative tier
30-portal-webserver	Starts and stops the Portal tier

You can stop a component by running the following:

```
<script>stop
```

You can start a component by running:

```
<script>start
```

For example to stop and start the LDAP tier, you would run:

```
10-ldap stop
10-ldap start
```

After you ensure that all tiers start up successfully, log into Luminis Platform using the administrator defined by the `admin.id` property you specified in the `setup.properties` used to install the Admin tier.

Review the logs

Check the various log files for errors.

During installation, a file is written to the system to record the overall status of the installation and of each tier that has been set for inclusion. After your installation is complete, you can review this log to verify installation options and to scan for any exceptions that need to be dealt with.

The Luminis install log is located under the following directory path:

```
$CP_ROOT/logs
```

The most important log files associated with each tier are:

Tier	Log file

LDAP (OpenDJ)	`$CP_ROOT/products/opends/logs/access`
	`$CP_ROOT/products/opends/logs/server.out`
	`$CP_ROOT/products/opends/logs/errors`

CAS	`$CP_ROOT/products/tomcat/cas-server/logs/catalina.out`
Admin	`$CP_ROOT/products/tomcat/tomcat-admin/logs/catalina.out` `$CP_ROOT/products/tomcat/tomcat-admin/logs/luminis.log`
Portal	`$CP_ROOT/products/tomcat/tomcat-portal/logs/catalina.out` `$CP_ROOT/products/tomcat/tomcat-portal/logs/luminis.log`

Administrative and Portal Server login information

After Luminis Platform and all related components are installed and started, you can log into the administrative server and Portal server.

Log in to the administrative server

Log into the administrative server.

Procedure

1. Open a Web browser and navigate to the administrative server.

   The server Web address would display in the following format:

   `https://wasatch.edu:8443`

   For example, the Web address may display as follows:

   `https://wasatch.edu:443`
2. Login using the default admin username and password.

   The default username and password are admin and admin. You can change the default username and password by adjusting the values of the `admin.id` and `admin.password` properties in the `setup.properties` file before installation.

Log in to the Portal server

Log in to the Portal server.

Procedure

1. Login to the administrative server.
2. Create a non-administrative user in the **User Management** portlet.
3. Open a Web browser and navigate to the Portal server.

   The server Web address would display in the following format:

   `http://<portal.virtual.host>:<portal.http.port>`

   For example, the portal Web address may display as http://wasatch.edu:FQDN

Post-installation tasks

The tasks in this section must be completed after you have run the installer.

Install a server certificate

To run the system securely, portions of it must be rendered through Secure Socket Layer (SSL) encryption. To render Web resources through SSL, a server certificate must be installed on your Web server.

During installation, a temporary certificate was generated and installed. Before you go into production, you should purchase and install a commercial server certificate. Until that time, all system users who log into the system receive a pop-up window stating that the server is not a trusted source.

The CAS, Admin, and Portal tiers all require a separate server certificate if they are installed on separate servers. Each certificate is tied to the hostname of the server on which it is used. If multiple Portal tiers are installed and configured behind a load balancer, only one certificate is needed.

Note: Make sure all Luminis servers are shut down before you complete the steps below. During the creation of the Certificate Signing Request (CSR), the administrator will have the option of generating a new certificate. If a certificate is changed while the system is running, it will cause problems if users attempt to log into the system.

This section outlines the procedures necessary to install a server certificate.

Note: Depending on the company that you elect to use, applying for and obtaining a server certificate can take anywhere from a few days to a couple of weeks. Please take this time into account as you plan for the purchase and installation of a certificate.

Create a certificate signing request and generate a new certificate

Create a certificate signing request (CSR) and generate a new certificate if needed.

About this task

To create the CSR:

Procedure

1. Source the `.cprc` file found in `$CP_ROOT`.
2. Navigate to the `$CP_ROOT/install/scripts directory`.
3. Run the `create_csr.sh` script and include the name of the directory where you want the CSR file (`cert.req`) to be created.

   For example, to create the `cert.req` file located in `/opt: run` the following command:

   ```
   ./create_csr.sh /opt/
   ```

Check to make sure the certificate information is accurate. If not, type G to generate a new certificate.

You see this information:

```
Check this certificate to make sure it contains the correct
information
-------------------------------------------------------------------
Alias name: tomcat
Creation date: Nov 19, 2010
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=luminishost8a.sct.com, OU=Wasatch University, O=WU, L=Salt
Lake City, ST=Utah, C=US
Issuer: CN=luminishost8a.sct.com, OU=Wasatch University, O=WU, L=Salt
Lake City, ST=Utah, C=US
Serial number: 4ce6ec89
Valid from: Fri Nov 19 14:30:49 MST 2010 until: Mon Nov 14 14:30:49
MST 2011
Certificate fingerprints:
MD5: 04:46:72:74:27:0B:2A:7E:51:DB:0B:65:8F:00:1F:F1
SHA1: 74:00:CE:E4:49:E5:32:8A:15:56:96:59:D2:6E:3A:58:15:7A:90:B9
Signature algorithm name: SHA1withRSA
Version: 3
-------------------------------------------------------------------
```

Warning! Make sure the host name is correct (CN=<host name>).

```
Also make sure the state (ST=<state>) is spelled out.
If any of the certificate information is incorrect then
you will need to generate a new certificate.
What do you want to do? (G)enerate a new certificate, (C)ontinue, or
E(X)it:
```

4. If you do not need to generate a new certificate then proceed to step 6. If you need to generate a new certificate and you typed G, then you will be prompted to enter all the certificate information manually. If the value inside the brackets '[ ]' is correct then type S to skip ahead to the next:

```
Host [luminishost8a.sct.com] (Type 's' to skip): s
School Name [Wasatch University] (Type 's' to skip): s
```

```
School Abbreviation [WU] (Type 's' to skip): s
City [Salt Lake City] (Type 's' to skip): s
State [UT] (Type 's' to skip): Utah
Country [US] (ex. US) (Type 's' to skip): s
Your new certificate will look like this:
-------------------------------------------------------------------------
Owner: CN=luminishost8a.sct.com, OU=Wasatch University, O=WU, L=Salt
Lake City, ST=Utah, C=US
Issuer: CN=luminishost8a.sct.com, OU=Wasatch University, O=WU, L=Salt
Lake City, ST=Utah, C=US
-------------------------------------------------------------------------
What do you want to do? [(C)ontinue or E(X)it]:
```

5. Type `C` to continue and your new certificate will be generated. If you mistyped some of the information then type `X` to exit. Return to step 3.
6. Type `C` to create the CSR.
   The CSR message is displayed as shown below:

```
Certification request stored in file </opt/cert.req>
Submit this to your CA
```

Submit the CSR to a Certificate Authority

You must submit your CSR to a commercial Certificate Authority (CA).

There are a number of companies that provide this service. The decision about which company to use is impacted by a number of factors, such as cost of the certificate, the amount of time it takes to process the CSR and return a certificate, and ancillary service offerings provided by the CA. After you select a CA vendor such as Thawte or Verisign, access the CA's Web site and follow the instructions provided for submitting the CSR.

Obtain the certificate and install it on the Web server

When your CSR is verified, the CA typically sends the CSR as files attached to an e-mail message.

The message includes a CA root certificate and your signed certificate. The actual certificate looks similar to the following:

```
-----BEGIN CERTIFICATE-----
MIICrzCCAhigAwIBAgIDE5jtMA0GCSqGSIb3DQEBBAUAMIGHMQs
wCQYDVQQGEwJaQTEiMCAGA1UECBMZRk9SIFRFU1RJTkcgUFVSUE
9TRVMgT05MWTEdMBsGA1UEChMUVGhhd3RlIENlcnRpZmljYXRpb
24xFzAVBgNVBAsTDlRFU1QgVEVTVCBURVNUMRwwGgYDVQQDExNU
aGF3dGUgVGVzdCBDQSBSb290MB4XDTAxMDMwNzIzMzYzM1oXDTA
```

```
xMDQwNzIzMzYzM1owgYkxCzAJBgNVBAYTAlVTMQ0wCwYDVQQIEw
RVdGFoMRcwFQYDVQQHEw5TYWx0IExha2UgQ2l0eTEeMBwGA1UEC
hMVQ2FtcHVzIFBpcGVsaW5lLCBJ
bmMuMRQwEgYDVQQLEwtFbmdpbmVlcmluZzEcMBoGA1UEAxMTc2N
pcGlvLmluLnRlYW1wLmNvbTCBnzANBgkqhkiG9w0BAQEFAAOBjQ
AwgYkCgYEAqQAJD2CiRxhcO9uj4H3PaTESs+PSqNnbHEInKnESx
UHVpBgfEI8yM4RUUZW2MnVuAYRO7qLGn16UXyVXIm0PkUT9fBOW
ao/7vtjhD/YGGk
1OObDbkW+CGz4u0OtAD46JwvSIhnDMP872N5Mq29Mj+VbA3ypNQ
J+TE2+ai9W/
zMCAwEAAaMlMCMwEwYDVR0lBAwwCgYIKwYBBQUHAwEw
DAYDVR0TAQH/
-----END CERTIFICATE-----
```

Install the root and intermediate certificate

Steps to install the root certificate or an intermediate certificate.

Procedure

1. Source the `.cprc` file found in `$CP_ROOT`.
2. Navigate to the `$CP_ROOT/install/scripts` directory.
3. Run the `import_root_cert.sh` script and include the parameters to specify the location of the root certificate file and root certificate alias.

   If you received a root certificate and an intermediate certificate file from the CA, do this:

   ```
   ./import_root_cert.sh -f <path to intermediate certificate> -a
   CAIntermediate
   ```

   ```
   ./import_root_cert.sh -f <path to root certificate> -a CARoot
   ```

Install the signed certificate

Steps to install your new signed certificate.

Procedure

1. Source the `.cprc` file found in `$CP_ROOT.`
2. Navigate to the `$CP_ROOT/install/scripts` directory.
3. Run the `import_signed_cert.sh` script and include the parameter to specify the location of the signed certificate file.

Results

For example, if you received your certificate from Verdigris, the signed certificate file might be `verisigncert.pem`. Enter this command:

```
./import_signed_cert.sh -f /opt/verisignsignedcert.pem
```

Set the default language in the Control Panel

Steps to set the default language to one other than English.

Procedure

1. Log into Luminis Platform.
2. In the **Control Panel**, expand the **Portal** menu, then click **Portal Settings**.
3. Under the Miscellaneous menu, click **Display Settings**.
4. In the **Default Language** drop-down list, select the desired default language.
5. Click **Save**.

Configure the Documents and Media Repository

You have two options to configure the Document Library.

The default option is a Jackrabbit repository configured to store documents in the Luminis database. You can also choose to set up the Amazon Web Service Simple Storage Service (Amazon S3) interface to store and retrieve data from the Web.

Configure the Jackrabbit repository

Luminis shares the Content Management System (CMS) repository with Liferay for the **Documents and Media**, **Targeted Content**, **Targeted Announcements**, and other portlets.

By default, the Luminis Platform installer for 5.3 will configure Liferay's Document Library to point to a Jackrabbit repository configured to store documents in the Luminis database. Apache Jackrabbit is a fully conforming implementation of the Content Repository for Java Technology API (JCR).

Jackrabbit is used as a CMS or Web Content Management (WCM) for Luminis Platform sites and users. It is intended to be as invisible as possible, but still provide ownership and content control for sites and courses.

JCR file management is used by the **Documents and Media**, **Web Content**, **Targeted Content**, and other portlets. When a file is uploaded the system calls the jackrabbit to track and manage the file. Files uploaded with Luminis Platform version 5.3 will share the Liferay Jackrabbit repository.

You can find the Jackrabbit configuration files at the following location for the administration and portal servers respectively:

```
$CP_ROOT/products/liferay/liferay-admin/data/jackrabbit/repository.xml
$CP.ROOT/products/liferay/liferay-portal/data/jackrabbit/repository.xml
```

To change the Jackrabbit repository configuration to store documents to a shared file system, refer to the Jackrabbit documentation located at http://jackrabbit.apache.org/.

Liferay 6.1 offers several options for the Document Library repository for clustered deployments, including an advanced file system store, CMIS store, or Amazon or Documentum stores. To

implement one of the other repository options please refer to the “Liferay Clustering” section in the *Liferay Portal 6.1 – User Guide* located at http://www.liferay.com/documentation.

Configure Amazon Web Service S3 storage

Amazon Simple Storage Service (Amazon S3), provides developers and IT teams with secure, durable, highly-scalable object storage.

You can configure Luminis to use Amazon S3 for the document library repository instead of JCR in one of these scenarios:

- Migrate Luminis 5.2.2 documents from JCR to S3 and configure S3 during the patch process
- Configure S3 in a clean install

For additional information and an introduction to Amazon S3, see the Amazon Web Service page.

Set up Amazon S3 support in Luminis Platform

Instructions to set up Amazon S3 in Lu minis Platform.

Procedure

1. Create an Amazon Web Service (AWS) account at http://aws.amazon.com/.
2. Log in to AWS and create a new Access key from the Security Credentials account detail page. Store the resulting download file in a safe location. It contains the secrets key needed to use the access key.
3. Navigate to the storage management home page, click **Create Bucket**, and enter a name for the AWS bucket.
4. In `setup.properties`, add entries for these items. Replace the <> bracketed entries with the values from the AWS setup above:

   ```
   jcr.initialize.on.startup=false
   aws.s3.access.key=<AWS S3 Access Key>
   aws.s3.secret.key=<AWS S3 Secret Key>
   aws.s3.bucket.name=<AWS S3 Bucket name>
   dl.store.config=S3Store
   ```

5. Run a clean installation.
6. After installation, the properties you set in the `setup.properties` are copied to the `portal-ext.properties` on each node.
7. To verify the S3 file repository, log into Luminis Platform, navigate to the **Documents and Media** portlet, and add a file. Navigate to the AWS S3 management page, double-click to open the bucket you created in “Prerequisites for Amazon S3 setup”, and follow the links into the repository structure to see the file you added.

   Note: If you are going to use the Amazon S3 system as a content migration source, do not change the value of `dl.store.config` or `dl.store.impl` until after the migration.

Related Links

Customize the installation values on page 13

Migrate content to Amazon S3 when you patch to 5.3.0

When you patch to the latest version of Luminis Platform, you must migrate the content from the previous version to Amazon S3.

Prerequisites for Amazon S3 setup

To migrate the content, begin with a Luminis platform 5.2.2 system configured to run the JCRStore model either by Jackrabbit or another JCR provider (this should be the default).

Procedure

1. To determine the type of the JCR file store subsystem, open `$CP_ROOT/products/tomcat/tomcat-admin/webapps/ROOT/WEB-INF/classes/portal-ext.properties`
2. Validate this property:
   `dl.store.impl=com.liferay.portlet.documentlibrary.store.JCRStore`
3. Log in to Luminis Platform, add the **Documents and Media** portlet to the page, and add one or more files.
4. Edit the `portal-ext.properties` file on each node.
   a) Add these properties to the `portal-ext.properties` file:

   Note: Enter appropriate values collected from the security credentials page of an Amazon services account.

   ```
   dl.store.s3.access.key=<AWS S3 Access Key>
   dl.store.s3.secret.key=<AWS S3 Secret Key>
   dl.store.s3.bucket.name=<AWS S3 Bucket name>
   ```

   b) Set this property to `false`.

   ```
   jcr.initialize.on.startup=false
   ```

5. Restart Luminis Platform.

Related Links

Customize the installation values on page 13

Migrate content to Amazon S3

Instructions to set up Luminis Platform to migrate content to Amazon S3.

Procedure

1. In the Luminis Platform admin server, navigate to the **Control Panel Server** menu. Choose **Server administration** and go to **Data Migration**.
2. In the drop-down menu in the section **Migrate documents from one repository to another**, change to store type to this line:

```
com.liferay.portlet.documentlibrary.store.S3Store
```

3. Click **Execute** to run the migration.

   When the migration begins, the Portal temporarily switches into maintenance mode. The count of files in the store display and then move to the Amazon S3 bucket you created in “Set up Amazon S3 support in Luminis Platform.”

   When the process is complete, you are redirected to the main page of the portal, files are moved into the bucket, and Amazon S3 is set as the file storage location. Document previews and thumbnails are not moved but are generated in place when the Document library detects they are missing.

4. To verify the files have been migrated, navigate to the AWS S3 management page, double-click to open the bucket, and follow the links into the repository structure to see the files.

Set Amazon S3 as the primary storage location
Set Amazon S3 as the preferred storage option and prevent the system from reverting to JCRStore.

Luminis Platform will store content in Amazon S3 until the next restart. To set Amazon S3 as the preferred storage option and prevent the system from reverting to JCRStore, set this property in `portal-ext.properties`:

```
dl.store.impl=com.liferay.portlet.documentlibrary.store.S3Store
```

Patch to 5.3.0
Add properties to the list of `resolved.properties`, and patch the system to 5.3.0.

Properties to add to `resolved.properties`:

```
jcr.initialize.on.startup=false
aws.s3.access.key=<AWS S3 Access Key>
aws.s3.secret.key=<AWS S3 Secret Key>
aws.s3.bucket.name=<AWS S3 Bucket name>
dl.store.config=S3Store
```

After you complete the patch, verify these in the `portal-ext.properties` file:

```
jcr.initialize.on.startup=false
dl.store.s3.access.key=<AWS S3 Access Key>
dl.store.s3.secret.key=<AWS S3 Secret Key>
dl.store.s3.bucket.name=<AWS S3 Bucket name>
dl.store.impl=com.liferay.portlet.documentlibrary.store.S3Store
```

Also verify these settings in the `portal-ext.properties` files found at `tomcat-admin/webapps/luminis/WEB-INF/classes/portal-ext.properties` and `tomcat-portal/webapps/luminis/WEB-INF/classes/portal-ext.properties`:

```
dl.store.config=S3Store
jcr.initialize.on.startup=false
dl.store.s3.bucket.name=<AWS S3 Bucket name>
```

Lucene setup in a Liferay cluster

Luminis Platform 5.3 will configure the Lucene search index using a shared index on the file system for all nodes. This is not the configuration recommended by Liferay for production systems unless you have a file locking-aware SAN.

Liferay supports other options for the Lucene search index:

- Enable the clustering in Liferay and enable the replication on the Lucene index
- Change the shared index to store in the database rather than file system

For more information on implementing one of these options, refer to the “Liferay Clustering” section in the *Liferay Portal 6.1 – User Guide* located at http://www.liferay.com/documentation.

Note: If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

If you choose to continue using the shared Lucene index on the file system, set up the admin and portal tier indexes after Luminis Platform is installed. These steps will share the search index and allow the system to use a single index for all of the nodes in the cluster.

Setup the admin tier index

The location of the Lucene index files is defined by the `lucene.dir` property in the `portal-ext.properties` file.

By default, this location is as follows:

```
$CP_ROOT/products/liferay/lucene
```

The location of the index files can be changed before installation by adding the `lucene.dir` property to the `setup.properties` file. Be aware that the value of this property must have a trailing "/".

For example: `lucene.dir=%{cp.root}/products/liferay/lucene/`

To change the location of the Lucene index files after installation, overwrite the `lucene.dir` property in the `$CP_ROOT/products/tomcat/tomcat-admin/webapps/ROOT/WEB-INF/classes/portal-ext.properties` file. Share the parent data folder so that all folders related to Lucene that are created on the server startup can access the folder.

Related Links

Customize the installation values on page 13

Setup the portal tier index

Instructions to set up the portal tier index.

Procedure

1. Overwrite the value of `lucene.dir` in `$CP_ROOT/products/tomcat/tomcat-portal/webapps/ROOT/WEB_INF/classes/portal-ext.properties` to point to the shared folder on the admin tier.
2. Restart the portal server.
3. Repeat the above steps for each of the portal nodes in the cluster environment.

Verify Liferay patch installation

To verify whether the Liferay patches are installed, use the Liferay Patching Tool.

All installed Liferay patches are listed.

Tier	Location
Admin	$CP_ROOT ./products/tomcat/tomcat-admin/patching-tool/patching-tool.sh info
Portal	$CP_ROOT ./products/tomcat/tomcat-portal/patching-tool/patching-tool.sh info

Luminis Platform installation tiers

The functionality of Luminis Platform software is packaged in four tiers that consist of various proprietary and third party software components. Each of these tiers can be installed separately, combined on one tier, or various combinations of tiers on separate servers.

Before installation, you are required to create a single or series of `setup.properties` files. The tiers property denotes which tiers to install. See “Luminis Platform installation configuration with setup.properties” for examples of the various methods of installing the different servers.

LDAP tier

The LDAP tier uses OpenDJ as the Directory server and stores user information such as user names and encrypted passwords.

This tier must be installed and started before the CAS, Admin and Portal tiers because the other tiers are all dependent upon the LDAP tier.

Related Links

Setup external directory server SunONE DS on page 70

CAS tier

The CAS tier is a Tomcat server running JA-SIG CAS and is used for authentication.

This tier must be installed and started before the Admin and Portal tiers because the other tiers are dependent upon the CAS tier. Users are redirected to the CAS interface when they try to log into Luminis Platform external CAS server.

Related Links

Install and Configure external CAS on page 73

Admin Server

The Admin server is a portal for administrators to configure the system, create custom content, and add, edit, or delete users.

This is a Tomcat Web server bundled with the Luminis Platform Web portal. This The Admin tier should be created before you create any Portal servers.

Portal Server

The Portal server is for non-administrative users, such as student, faculty, and alumni users.

This portal is a Tomcat Web server bundled with the Luminis Platform Web portal.

Related Links

Configure Luminis Platform tiers using a load balancer on page 37

Luminis Platform System Basics

Information about the Luminis® Platform, default installation values, and some basic information for supporting and working with the system, such as procedures for uninstalling, starting up and shutting down the system and its components.

Before you install any components of the system, you should review this information thoroughly so that you have a better understanding of what you are installing and where you will install it. If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

Related Links

Installation tasks on page 11

Luminis Platform version information

To print out the Luminis Platform version information, use the `lpver` command.

After you run the command, the build displays in the following manner:

```
Luminis LP-5.3.0.0 build <####>
```

System startup and shutdown

Servers installed on separate computers require the Directory Server on the Luminis Platform server to be running before they can successfully start up.

Because the Luminis Platform system contains several computers, and a number of software components and configurations, you must know how to start up and shut down the system properly. The following sections provide procedures for starting the Luminis Platform system. Review these procedures as you will use them frequently after you install the system, administer the system, or reset system configurations as outlined in the rest of this guide.

Start the Luminis Platform system

Instructions to start up the entire Luminis Platform system.

Procedure

1. Power on all the computers installed with Luminis Platform components, starting with the Luminis Platform server.
2. Log in to the Luminis Platform server as the administrative user.
3. From the command line type: `. /<path to CP_ROOT/.cprc>`

Startup and shutdown of Luminis Platform servers

In some instances, you may need to shut down and restart the Web server.

For example, if you make changes to the configuration directory, you must shut down and restart (bounce) the Web server for changes to take effect. Before starting up the Luminis Platform servers, source the `.cprc` file found in `$CP_ROOT`:

```
. /<path to $CP_ROOT/.cprc>
```

Start up all Luminis Platform components on a server in proper order with the `lpstart` command and shutdown all Luminis Platform components on a server with the `lpstop` command.

Within the `$CP_ROOT/bin` directory are the following scripts:

Script	Description
10-ldap	Starts and stops the LDAP server
20-cas-webserver	Starts and stops the CAS server
25-admin-webserver	Starts and stops the administrative server
30-portal-webserver	Starts and stops the Portal server

You can stop a component by running the following script:

```
<script> stop
```

You can start a component by running the following script:

```
<script> start
```

For example to stop and start the LDAP server, you would run the following script:

```
10-ldap stop
10-ldap start
```

Startup timeout option

If the server requires more than 15 minutes to start up, use `-t` option to configure the default startup timeout option.

The Admin and Portal servers can take a long time to start up. By default, the startup script times out at 15 minutes if the server has not started. The standard timeout command is as follows:

```
lpstart -t 15
```

For example, if you want to assign a 20 minute limit for all servers to start up, run this command:

```
lpstart -t 20
```

If you want to assign a 20 minute server-start limit to the Admin server only, run this command:

```
25-admin-server -t 20 start
```

Install Patches to Luminis Platform

Installing or uninstalling Luminis® Platform patches, and also patching the external directory LDAP server.

Periodically, Ellucian® releases upgrades, patches, or hotfixes for Luminis Platform.

Warning! A license key is required to run the portal. If you have not installed this key, see “If SELinux is not properly configured, it can cause issues during the Luminis 5.3 installation or patch. SELinux can be disabled by setting SELINUX=disabled in the following file: /etc/selinux/ con fig”. After the key is installed, it will apply to all future patches.

If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

Install Luminis Platform patches or hotfixes

After a maintenance patch or hotfix is released, you must remove any previously installed hotfixes before you can install the new patch or new hotfix. The secure LDAP customization must also be removed before applying the latest patch.

About this task

The installation process for hotfixes and patches are the same.

Note: If a customer has changed 5.1.1 to use LDAP ports other than ports used during initial install (an action that is not officially supported), the customer must revert the changes before applying 5.3.

You can patch Luminis Platform to a different version by running a patch or hotfix executable on each server. Running the patch shuts down all Luminis Platform components installed on the server, such as the LDAP, CAS, Admin and Portal components. Any files updated during the patch are backed up and can be restored later if necessary.

In a multi-node installation where Portal, Admin, or CAS servers are installed on separate nodes from the LDAP server, you should shut down the Portal, Admin, and CAS servers before you patch the LDAP node.

The Luminis Platform patch or hotfix is platform-independent, so one patch or hotfix executable is used for all Linux, Solaris and Windows platforms.

To upgrade Luminis Platform to a newer version:

Procedure

1. If using Linus and Polaris, login as a root user unless Luminis Platform was installed as a non-root user.
2. Copy the Luminis Platform patch executable to any directory.
3. Source the `.cprc` file found in `CP_ROOT`.

Note: CP_ROOT is the root directory specified in the setup.properties file as cp.root when Luminis Platform was first installed. Sourcing the .cprc sets some required environment variables.

4. Run the patch executable.

   Suppose Luminis Platform was installed on a Linux server with CP_ROOT set to /opt/lp5. Use these steps to install a 5.0.1.0 patch:

   a) Copy LP-5.0.1.0.XXXX-patch to /home.

      ```
      . <cp.root>/.cprc
      ```

      ```
      cd /home
      ```

      ```
      ./LP-5.0.1.0.XXXX-<patch>
      ```

      The following prompt is displayed:

      Warning! All Luminis Platform Servers installed on this tier will be shutdown before patching.

      Do you want to continue? (Y or N):

   b) Type Y to continue.

Uninstall Luminis Platform patches or hotfixes

After a patch or hotfix is installed, it can be removed to return to a previous install level.

About this task

For example, when you complete the first installation the version displays as 5.0.0.0. If you apply subsequent patch 5.0.1.0, the Luminis Platform system is updated to the 5.0.1.0 version.

To remove the 5.0.1.0 patch and return to the 5.0.0.0 version:

Procedure

1. Source the .cprc file in the CP_ROOT directory.
2. Cd to $CP_ROOT/install/patch
3. Run the unpatch script passing in the patch version to uninstall.

   For example, the script might read as follows:

   ```
   ./unpatch 5.0.1.0
   ```

   Before you can install the patch, any installed hotfix should be removed using the unpatch script as shown above. To find out the version of the installed hotfix, run the lpver script as described in this example:

   a) Source the .cprc file as follows:

      ```
      . <cp.root>/.cprc
      ```

b) Run the `lpver` script.

Note: In cases where the upgrade patch fails and system needs to be reverted to previous state, in order to restore both file system and database to the same version's data, it is recommended that you capture database (DB) instance and VM (if applicable) snapshots, taken immediately before the upgrade attempt. Luminis Platform 5 versioning discrepancies between DB and file system may lead to additional issues during future upgrade attempts.

Deactivate and clear JMS/LMG synchronization subscriber

If you previously used synchronization events from Banner, you must clear and remove the previous JMS durable subscriber on the JMS/LMG gateway.

About this task

Luminis Platform 5.2.2 adds functionality to support Smart/Notify events from Banner. When you deactivate and clear the JMX/LMG synchronization subscriber, Luminis Platform creates two new subscriptions, one for synchronization and one for Smart/Notify events, with the client ID changed to allow recognition of which subscription goes with which type of operation.

Procedure

1. Shut down Luminis before you proceed with the subscription removal. If your LMG gateway is configured to use the Luminis directory (LDAP) for message storage, then re-start the directory using this command:

   ```
   $CP_ROOT/bin/10-ldap start
   ```

2. To find the previous durable subscriber on Banner eLearning LMG, execute this IMQ server command:

   ```
   <glassfish_home>/mq/bin/imqcmd list dur -b <hostname> -u <username>
   ```

3. Find the subscriber for the Luminis synchronization events, and note its subscriber name. The subscriber is automatically assigned your Luminis admin host name as the client ID, and the destination name is `com_sct_ldi_sis_EntityEvents`. To purge and remove the durable subscriber, execute these commands:

   ```
   <glassfish_home>/mq/bin/imqcmd purge dur -b <hostname> -u <username> -
   n <durable_subscriber_name> -c <client_id>
   ```

   ```
   <glassfish_home/mq/bin/imqcmd destroy dur -b <hostname> -u <username>
   -n <durable_subscriber_name> -c <client_id>
   ```

Patch the external directory (LDAP) server

Occasionally, the external Directory server (DS) should be updated with Luminis Platform specific schemas or configuration changes. These updates are included within the patch executable and must be completed manually.

To update the external DS, extract the patch installer files to a temporary directory using the `--target` option as follows:

```
./LP.5.3.0.0.XXXX-patch --target /opt/_patch
```

In this temporary directory you will find a subfolder named `ldif`. Open each of the files contained within and follow the instructions in the file.

Remove custom directory schema changes

New configuration properties were added to Luminis 5.3. These properties control where in the directory service you place directory attribute values for the Luminis Person and enable you to remove directory schema changes to the directory service.

Prerequisite

Before running the Luminis Platform 5.3 patch, add these properties to the `$CP_ROOT/install/resolved.properties` file:

- `ldap.directory.person.id.attribute=cn`
- `ldap.directory.udc.id.attribute=udcid`

Change Luminis Platform person directory attributes

The configuration property, recommended directory attributes, and steps to move the current directory attribute values to the new directory attributes.

About this task

These recommended directory attributes come from the standard `inetOrgPerson` object class. Luminis Platform assumes the directory attribute will contain one value and is of the 'Directory String' syntax. If you use a 'Multi Value' directory attribute, Luminis Platform only uses the first value that is returned.

Table 2: Default Installation Values

Property	Default Value	Description
ldap.directory.person.id.attribute	employeeNumber	The directory attribute for the immutable person identifier.
ldap.directory.udc.id.attribute	departmentNumber	Directory attribute for the Banner UDC identifier.

For information on how to change the configuration properties with jConsole, see "Configuration management with JMX" in the *Luminis Platform Administration Guide*.

After you change the configuration properties, users cannot log in until the existing directory attribute values are moved to the new directory attributes. Use `lptool` to move the directory attribute values. The process may be lengthy if you have a large number of Luminis Platform users.

After you change the configuration properties, complete these steps to move the current directory attribute values to the new directory attributes:

Procedure

1. On the Luminis Administration server, run this command:

   ```
   lptool updtdirattrs start
   ```

2. All activity, including errors, is logged at `$CP_ROOT/products/tomcat/tomcat-admin/logs/luminis.log`.

   If errors do occur, review the log and make the necessary changes so the directory attribute values move successfully. You can run the `lptool` command multiple times. Once all directory attribute values are successfully moved, the time of the last successful lptool command run is displayed in the command window.

   Note: If you run the Luminis Platform 5.3 step before you run the Prerequisite on page 62, to enable users to login in you must set the original configuration properties for the Luminis Person directory attributes to the original values. For information on how to change the configuration properties with jConsole, see "Configuration management with JMX" in the *Luminis Platform Administration Guide*.

   Table 3: Original Installation Values

Configuration Property	Original Directory Attribute
ldap.directory.person.id.attribute	cn
ldap.directory.udc.id.attribute	udcid

Single sign on to Banner with Luminis Platform

Single sign on to Banner with Luminis Platform 5.3.0 and BEIS ssomanager require that CAS provides values from the Luminis Platform 5 directory for UDC_IDENTIFIER and 'cn' to identify a person. Luminis Platform 5 defaults to standard directory attributes for these values.

This process differs from earlier versions of Luminis Platform 5, where LDAP attributes used for these values were ‘udcid’ and ‘cn’.

If you would like to use the same attributes for these values as with earlier versions of Luminis Platform 5, you will need to add `ldap.directory.person.id.attribute` and `ldap.directory.udc.id.attribute` properties to your `setup.properties` file BEFORE you install Luminis Platform 5.3.

When adding these properties to your `setup.properties` file, the values should be set to these directory attribute values:

```
ldap.directory.person.id.attribute=cn
```

```
ldap.directory.udc.id.attribute=udcid
```

Refer to the "Customize the installation values" table, for more information about the Luminis directory properties.

Changing the default directory attribute values after Luminis 5.3.0 install

If you installed Luminis Platform 5.3 with the new defaults or configure other directory attributes for these values, the installation results in a CAS `deployerConfigContext.xml` file which does not take into consideration the new Luminis Platform 5.3 directory property attribute values.

The CAS `deployerConfigContext.xml` assumes attributes will be those used in earlier Luminis Platform 5 versions, with these entries within bean 'attributeRepository':

```
<entry key="udcid" value="UDC_IDENTIFIER" />
```

```
<entry key="cn" value="cn" />
```

Changes are needed for `deployerConfigContext.xml` to send the correct attributes for BEIS ssomanager.

Update `deployerConfigContext.xml` bean 'attributeRepository' to specify LDAP attribute for key values for 'UDC_IDENTIFIER' and 'cn'.

Following is the full path to this file:

```
$CP_ROOT/products/tomcat/cas-server/webapps/cas-web/WEB-INF/
deployerConfigContext.xml
```

If the default properties and values are used:

```
ldap.directory.person.id.attribute=employeeNumber
```

```
ldap.directory.udc.id.attribute=departmentNumber
```

Then these are the keys and values in `deployerConfigContext.xml`:

```
<entry key="departmentNumber" value="UDC_IDENTIFIER" />
```

```
<entry key="employeeNumber" value="cn" />
```

This is what `deployerConfigContext.xml` will look like after correcting it:

```
----------
<bean id="attributeRepository"
 class="org.jasig.services.persondir.support.ldap.LdapPersonAttributeDao">
.
.
.
  <property name="resultAttributeMapping">
    <map>
      <!-- Mapping between LDAP entry's attributes (key) and
 Principal"s (value) -->
      <entry key="uid" value="uid" />
      <entry key="departmentNumber" value="UDC_IDENTIFIER" />
      <entry key="employeeNumber" value="cn" />
      <entry key="givenname" value="Formatted Name" />
      <entry key="mail" value="EmailAddress" />
    </map>
  </property>
</bean>
----------
```

A restart of CAS is required for these changes to take effect.

Apply the 5.3 Patch Upgrade

These sections describe how to apply the 5.3 upgrade in an unlatched Luminis® 5.2.2 environment.

Before you apply the Luminis Platform 5.3 upgrade, backup your files, especially any customizations you may have applied since installing and patching Luminis Platform. Installation of the upgrade will overwrite certain files and nullify your customizations.

The following key configuration and properties files have been updated for 5.3 Any customizations made to these files will need to be re-implemented following the upgrade.

Note: Do not attempt to simply restore the file from backup, because other critical settings may have changed in addition to your customizations.

- Admin and Portal tiers

  `$CP_ROOT/products/tomcat/tomcat-<tier>/shared/classes/bootstrap.properties`

  `$CP_ROOT/products/tomcat/tomcat-<tier>/webapps/ROOT/WEB-INF/classes/portal-ext.properties`

  `$CP_ROOT/products/tomcat/tomcat-<tier>/webapps/banner-cas-client/WEB-INF/classes/cas-properties/cas-client.properties`

- CAS Tier

  `$CP_ROOT/products/tomcat/cas-server/webapps/cas-web/WEB-INF/deployerConfigContext.xml`

All overridden files will be archived in the directory `$CP_ROOT/install`, in a file named `luminis-backup-<version>.zip`. Customized files may be extracted for this backup file for reference. For example, when the 5.3.0.0 patch is applied, the file `luminis-backup-5.3.0.0.zip` will be created containing all the 5.3.0.0 files that were overwritten. Files can be extracted using normal zip tools if an administrator wishes to compare and re-apply any customizations.

Note: The patch should be executed as the "install user." For example, if the initial install was performed as 'root, the upgrade should be performed as root.' To verify <install user>, check the `$CP_ROOT/logs/install.log` for LP-5.3.0.0.3814-xxxx-xxx setup process execution.

If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

Warning! Beginning with Luminis 5.2, a license key is required to run the portal. Liferay EE 6.1.20 requires this key for activation.

Note: If SELinux is not properly configured, it can cause issues during the Luminis 5.3 installation or patch. SELinux can be disabled by setting SELINUX=disabled in the file: `/etc/selinux/config`

Related Links

Liferay license key requirements on page 10

Install a patch on a single-tier environment

A patch executed in a single-tier environment will automatically shutdown servers during upgrade.

About this task

Note: If the Luminis 5.1.1 system was changed to use LDAP ports that were not used during the initial install (an action that is not officially supported), it is recommended you contact Ellucian customer support for help in reverting the changes before applying 5.3.The 5.3.0 upgrade release contained CAS client upgrades. You must install this upgrade on all tiers in the Luminis Platform 5.2.2 environment, including the CAS node, if it resides separate from the other nodes.

To patch a single-tier install:

Procedure

1. Login to the tier as <*install user*>.
2. Source `.cprc` file as follows:

   ```
   . <cp.root>/.cprc
   ```
3. Run the `LP-5.3.0.0.XXX-patch` patch upgrade, and follow the instructions on the screen.
4. Once you successfully complete the patch, it is recommended that you start the Luminis components individually to allow Liferay to complete its patching process. First start `10-ldap`, then `20-cas-webserver`, then `25-admin-webserver`. After you start the Luminis admin server, review the logs for successful startup and completion of the automatic Liferay patch process. Log in as an admin user to verify portal operation:
   - `$CP_ROOT/bin/10-ldap start`
   - `$CP_ROOT/bin/20-cas-webserver start`
   - `$CP_ROOT/bin/25-admin-webaserver start`

     The admin Web server is likely to require the most time to start up, and the Luminis startup command may time out and print an error message. Monitor the `luminis.log` file in `$CP_ROOT/products/tomcat/tomcat-admin/logs` to determine when the Liferay update has completed successfully. For more information about this upgrade issue, see the LP5 Master Wiki on the Ellucian eCommunities Web site.

   The log file will contain entries similar to the following:

   ```
   3% - INFO [localhost-startStop-1]
   com.liferay.portal.kernel.upgrade.UpgradeProcess:175 Upgrading
   com.liferay.portal.upgrade.UpgradeProcess_6_0_0

   7% - INFO [localhost-startStop-1]
   com.liferay.portal.kernel.upgrade.UpgradeProcess:175 Upgrading

   com.liferay.portal.upgrade.v6_0_1.UpgradeSchema

   10% - INFO [localhost-startStop-1]
   com.liferay.portal.kernel.upgrade.UpgradeProcess:175 Upgrading
   ```

```
com.liferay.portal.upgrade.v6_1_0.UpgradeCamelCasePortletPreferences

13% - INFO [localhost-startStop-1]
com.liferay.portal.dao.db.BaseDB:627 Dropping stale indexes

30% - INFO [localhost-startStop-1] com.liferay.portal.dao.db.BaseDB:96
create index IX_4BFABB9A on DLFileVersion (uuid_);

40% - INFO [localhost-startStop-1] com.liferay.portal.dao.db.BaseDB:96
create index IX_7F26B2A6 on MDRRuleGroup (uuid_);

51% - INFO [localhost-startStop-1]
com.sghe.luminis.cache.impl.CachingServiceImpl:236 WARNING: using
default config for unconfigured cache: com.liferay.portal.kernel.dao.
orm.FinderCache.com.liferay.portal.model.impl.PermissionImpl.List2

65% - INFO [com.liferay.portal.kernel.deploy.auto.AutoDeployScanner]
com.liferay.portal.kernel.deploy.auto.AutoDeployDir:177 Processing

LP5-ellucian-theme-6.0.6.1.war

67% - INFO [localhost-startStop-1]
com.liferay.portal.plugin.PluginPackageUtil:1033 Reading plugin
package for luminis

89% - INFO [localhost-startStop-1]
com.sghe.luminis.liferay.integration.LARImporter:84 Checking LAR
imports...

96% - INFO [localhost-startStop-1]
com.liferay.portal.deploy.hot.PortletHotDeployListener:294 Registering
portlets for luminis-banner

98% - INFO [localhost-startStop-1]
com.liferay.portal.deploy.hot.HotDeployImpl:178 Deploying LP5-
ellucian-theme from queue

99% - INFO [main] org.apache.catalina.startup.Catalina:691 Server
startup in XXXXXXXX ms
```

Install a patch on a multiple-tier environment

In Luminis Platform 5 environments running on two or more servers -- separate Directory Server (DS), CAS, Admin, and Portal servers -- the patch should be applied to the tier hosting the DS server first, as it will need to be restarted during upgrade.

About this task

After the DS server is restarted the patch can be applied to the CAS, Admin, and Portal tiers, in that order.

Note: If the Luminis 5.1.1 system has been changed use LDAP ports other than ports used during the initial install (an action that is not officially supported), it is recommended you contact Ellucian customer support for help in reverting the changes before applying 5.3. The 5.2.2 upgrade release contained CAS client upgrades. You must install this upgrade on all tiers in the Luminis Platform 5.2 environment, including the CAS node, if it resides separate from the other nodes.

To apply the patch in a multiple-tiered environment:

Procedure

1. Shutdown all tiers in this order:
    1. Portal
    2. Admin
    3. CAS
    4. DS
2. Patch the tier hosting the DS server:
    a) Login to tier hosting DS server as <*install user*>.
    b) Source the `.cprc` file:

    ```
    . <cp.root>/.cprc
    ```

    Unmount `-f -l $CP_ROOT/products/luminis-repository/repository/datastore/`.
    c) Run the `LP-5.3.0.0.XXX-patch` patch upgrade, and follow the instructions on the screen.
    d) Once the patch completes successfully, use the `lpstart` command to start the patched Luminis Platform 5.3 environment as `<system.user>`, specified during initial setup. The `<system.user>` variable can be verified via `$CP_ROOT/logs/install.log`.
3. While the other tiers are stopped, patch the CAS, Admin and Portal tiers.
    a) Login to each tier as <*install user*>.
    b) Source the `.cprc` file:

    ```
    . <cp.root>/.cprc
    ```

    c) Run the `LP-5.3.0.0.XXX-patch` patch upgrade, and follow the instructions on the screen.
4. Mount `<NFS-device> $CP_ROOT/products/luminis-repository/repository/datastore/`.
5. Start the patched Luminis Platform 5.3 environment as the <*system.user*>, in this order:
    1. DS tier. This tier should already be running.
    2. CAS
    3. Admin
    4. Portal tier(s)

Setup external directory server SunONE DS

How to setup a SunONE directory server as an external Lightweight Directory Access Protocol (LDAP) and install Luminis® Platform.

You can use an external directory server to store user information.

Related Links

LDAP tier on page 54

Create new DN for Luminis Platform

Before installing a new directory server (DS), you must create a new branch off of the current base DN.

About this task

To create a new branch, complete the steps as displayed in this example:

Procedure

1. Create a new branch from the base DN. You can name the DN according to your preference as long as you ensure the name is the same as the `ldap.base.dn` property in the `install.properties` file.

   For example, if you are using a SunONE DS like the one installed with Luminis Platform 4, the base DN name might be `o=luminis`.

2. To create a DN named `o=platform5,` copy these lines and save them into an LDIF file named `lp5.ldif`:

   ```
   dn: o=platform5, o=luminis
   changetype: add
   objectclass: top
   objectclass: organization
   ```

3. Take the new `lp5.ldif` file and import it using the `ldif2ldap` script found in the DS instance directory.

   For example, the new branch may appear as follows:

   ```
   ./ldif2ldap -D "cn=Directory Manager" -w <password> -f /opt/lp5.ldif
   ```

Set appropriate LDAP properties in setup.properties files

Once the new `o=platform5`, `o=luminis` directory server (DS) branch is created, use this value in your `setup.properties` file or files as the value for the `ldap.base.dn` property.

Make sure to set all the other LDAP properties specific to the external DS.

An example LDAP property within `setup.properties` file is displayed:

```
ldap.base.dn=o=platform5,o=luminis
ldap.host=ldaphost.wasatch.edu
ldap.port=389
ldap.secure.port=636
ldap.directory.manager.dn=cn=Directory Manager
ldap.directory.manager.password=****
```

Extract the LDAP schema LDIFs from the installer

Before installation, you must copy the Luminis Platform schema LDIFs into the config/schema directory of the external DS.

Procedure

1. Extract the installer files to a temporary directory using the `--target` option.

   The installer files appear as follows:

   ```
   ./LP.5.3.0.0.XXXX-solaris-32bit --target /opt/_temp
   ```

2. In this temporary directory is a subfolder named `ldif`. Copy the files from this directory into the config/schema directory of the external DS.

Install Luminis Platform on all servers

Install Luminis Platform on each of the servers.

Import default users

Import the `test-users.ldif` file into the external DS.

The file is located on the tier where the Admin server is installed, in `$CP_ROOT/install`.

Use the `ldif2ldap` script to import the file.

Start all Luminis Platform servers

Once the external DS is started, start all Luminis Platform servers.

Install and Configure external CAS

The steps to install Luminis® Platform against an externally installed and managed CAS server.

You do not use the Luminis Platform installer to install the CAS server. In order for the Admin server or any Portal server to connect via Secure Socket Layer (SSL) to the CAS server and allow users to log into the system, you must store the CAS SSL certificate in the servers. Some scripts are provided to aide this installation process.

Note: If you are using both directory service and an authentication source that are external to the default Luminis Platform OpenDJ and CAS deliverables, portal users must exist in both the external directory service and in the Luminis Platform directory.

Related Links

Customize the installation values on page 13
CAS tier on page 55

Pre-installation step

In order to install Luminis Platform and use an externally installed and managed CAS server, you must set these properties in your `setup.properties` files.

About this task

```
cas.host
cas.http.port
cas.https.port
cas.webapp.path
```

For example, the properties may be set as follows:

```
cas.host=luministest.wasatch.com
cas.http.port=8090
cas.https.port=8447
cas.webapp.path=cas-web
```

Related Links

Customize the installation values on page 13

Store CAS certificate in LDAP for automated keystore import

After you install at least one Luminis Platform Portal or Admin Server, you must export the CAS SSL certificate from the Java keystore on the CAS server and then store it into LDAP so that the Admin and Portal servers can access the CAS server.

About this task

To export the CAS SSL certificate from the Java keystore and store it into LDAP, complete these steps on the CAS server:

Procedure

1. Set the JAVA_HOME environment variable.

   The variable would display in the following format:

   ```
   JAVA_HOME=/opt/jdk1.6.0_05
   ```

2. Run the following commands:

   ```
   KEYTOOL="$JAVA_HOME"/bin/keytool
   ```

   ```
   "$KEYTOOL" -export -rfc -keystore <keystore file> -alias <certificate alias> \
   ```

   ```
   -keypass <key password> \
   ```

   ```
   -storepass <keystore password> -file <export file name>
   ```

3. Copy the exported SSL certificate to a server with at least one Luminis Platform-type server installed, such as an LDAP, Portal, or Admin server. Paste the certificate into the `$CP_ROOT/install` directory and rename it cas.cert.
4. Source the `$CP_ROOT/.cprc` file.
5. Run the script `$CP_ROOT/install/scripts/store_cas_cert.sh`

   After you complete these steps, start or restart the portal or admin server to add the CAS certificate to the server's keystore.

Add Luminis Platform as a service in the external CAS server

In order for CAS to perform authentications for Luminis Platform, CAS must be configured to support Luminis Platform.

About this task

Add the necessary Luminis Platform service configurations:

Procedure

1. Authenticate to the add services URL for the external CAS server:

`https://<cas-server:port>/cas-web/services/add.html`

2. For example if the CAS server hostname were dculumsvc05, the URL would be as follows:
   `https://dculums05:8888/cas-web/services/add.html`
3. Click the **Add New Service** link and add five services for the Luminis Platform server:
   a) For the first service, enter:
      Name: `CAS Services Management`
      Service URL: `https://dculums05.wasatch.edu:8888/cas-web/services/*`
      Description: `CASServices Management`
      Theme Name: `default`
      Status: `Enabled; Allowed to proxy; SSO Participant`
      Attributes: `uid; sn; cn; EmailAddress`
   b) For the second service, enter:
      Name: `Luminis`
      Service URL: `https://dculums05.wasatch.edu:443/c/portal/login`
      Description: `Luminis Portal`
      Theme Name: `default`
      Status: `Enabled; Allowed to proxy; SSO Participant`
      Attributes: `uid; sn; cn; EmailAddress`
   c) For the third service, enter:
      Name: `Luminis Admin Banner CAS Client`
      Service URL: `https://dculums05.wasatch.edu:8443/banner-cas-client/authorized/banner/**`
      Description: `Luminis Admin Protected Banner cas client`
      Theme Name: `banner`
      Status: `Enabled; SSO Participant`
      Attributes: `uid; sn; cn; EmailAddress`
   d) For the fourth service, enter:
      Name: `Luminis Portal Banner CAS Client`
      Service URL: `https://dculums05.wasatch.edu:443/banner-cas-client/authorized/banner/**`
      Description: `Luminis Portal Protected Banner cas client`
      Theme Name: `banner`
      Status: `Enabled; SSO Participant`
      Attributes: `uid; sn; cn; EmailAddress`

e) For the fifth service, enter:

Name: `Luminis Admin`

Service URL: `https://dculums05.wasatch.edu:8443/c/portal/login`

Description: `Luminis Admin Portal`

Theme Name: `default`

Status: `Enabled; Allowed to proxy; SSO Participant`

Attributes: `uid; sn; cn; EmailAddress`

Restart the Luminis Platform system

In order for the Luminis Platform server to recognize the CAS SSL certificate, use these commands to restart the Luminis Platform server.

About this task

```
lpstop
lpstart
```

Related Links

Startup and shutdown of Luminis Platform servers on page 57

Install Luminis Platform as Non-root User using Linux and Solaris

There are two alternatives for installing and running Luminis Platform.

By default, Luminis® Platform should be installed as the root user. This requirement rests on the fact that root access is required to run Luminis Platform servers on privileged ports, which are port numbers less than 1024.

Alternative options:

- Install as root, but start up Luminis Platform servers while logged in as non-root user
- Install as non-root user, and start up Luminis Platform servers while logged in as non-root user

Install as root user and startup as non-root user with root password required

Run Luminis Platform using privileged ports, those assigned port numbers less than 1024, and startup the servers while logged in as a non-root user.

Complete the steps described in “Installation tasks”, including the installing Solaris or Linux option.

After installation, login as the non-root user, source the `.cprc` file, and run `lpstart` to startup the Luminis Platform servers. You will receive a prompt for the root password after running `lpstart`.

Related Links

Installation tasks on page 11

Install as non-root user and startup as non-root user

If you want to run Luminis Platform using non-privileged ports, those assigned port numbers greater than 1024, it is possible to install and startup while logged in as a non-root user.

Procedure

1. In your `setup.properties` file set these properties:

   ```
   system.user=<non-root user>
   system.group=<non-root group>
   root.access.required=no
   <all ports for tiers installed must be > 1024>
   ```

   Some examples of partial `setup.properties` are:

   ```
   tiers=ldap,cas,admin,portal
   ```

```
system.user=lp5user
system.group=lp5group
root.access.required=no
admin.http.port=8080
admin.https.port=8443
portal.http.port=8085
portal.https.port=8445
ldap.port=10389
ldap.secure.port =10636
cas.http.port=8447
cas.https.port=8090
```

2. Login as the non-root user to install Luminis Platform.
3. Start up as non-root user using `lpstart`. You will not receive a prompt for the root password.

Warning! If the ports are not all set to non-privileged ports, the servers fail to start up.

Liferay 6.2 user interface changes for Luminis Platform 5.3

A list of changes made to the Liferay 6.2 user interface for Luminis® Platform 5.3.

These files override the default Liferay 6.2 distribution. If the updated files are removed, the Liferay 6.2 files are used, resulting in a reset to the original Liferay distribution. In most cases, the changed files reside on both Luminis Platform servers.

- Access the admin portal server at `$CP_ROOT/products/tomcat/tomcat-admin/<file location>`
- Access the student portal server at `$CP_ROOT/products/tomcat/tomcat-portal/<file location>`

Any updates to Liferay text for messages, alerts, buttons labels, headings, and so forth are located in the `Language-ext` properties files. The extensions to Liferay's language files are in the following locations:

- Access language files in the admin portal server at `$CP_ROOT/products/tomcat/tomcat-admin/webapps/ROOT/WEB-INF/classes/content/Language-ext_*.properties`
- Access language files in the student portal server at `$SCP_ROOT/products/tomcat/tomcat-portal/webapps/ROOT/WEB-INF/classes/content/Language-ext_*.properties`

There will be several language properties files that correspond to the languages supported by Luminis Platform. The file `Language-ext_en.properties` refers to the English version.

Note: The interface changes apply to new installations and patched upgrades.

If you have additional questions about Luminis Platform or associated third-party software, or you want submit a feature request or problem with the software, contact Ellucian Client Support.

Bookmarks description field

Updates	Change file location
Removed the HTML escaped characters from the description. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/bookmarks/folder_action.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/bookmarks/entry_columns.jspf

Control Panel home

Updates	Change file location
Removed the **Add Users** option from the main control panel page. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/control_panel_home/view_actions.jsp

Control Panel server administration

Updates	Change file location
Removed the **Data Migration** and **Shutdown** tabs from the main server administration page. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/admin/server.jspf /webapps/luminis/custom_liferay_jsps/html/portlet/bookmarks/entry_columns.jspf

Custom data fields

Updates	Change file location
Removed the **Delete** option from the **Actions** pop-up menu for just the `luminis_dynamic_group` custom field. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/expando/expando_action.jsp

Hidden user options

Updates	Change file location
Removed several options from the user form. The options are controlled by the following Liferay properties which are overridden in the `portal-ext.properties` file for admin and portal servers. • layout.form.update	/webapps/ROOT/WEB-INF/classes/portal-ext.properties

Updates	Change file location
• layout.set.form.update The changed file is located in the admin and student portals.	
Removed the **CSS** field from the **Look and Feel** option. The changed files are located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/display_settings.jsp

My Sites portlet

Updates	Change file location
Removed the **Available Sites** option from the toolbar.	/webapps/luminis/custom_liferay_jsps/html/portlet/my_sites/view.jsp
Removed the **Join** option from the **Action** drop-down menu. The changed files are located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/my_sites/site_action.jsp

Monitor option from Control Panel

Updates	Change file location
Disabled the Liferay**Monitor** portlet.	No files were changed. To activate, set the **Monitor** portlet to **Active**.

Password Policy option from Control Panel

Updates	Change file location
Disabled the Liferay **Password Policy** portlet.	No files were changed. To activate, set the **Password Policy** portlet to **Active**.

Portal settings

Updates	Change file location
Removed sharing options from portlet configuration.	/webapps/luminis/custom_liferay_jsps/html/portlet/portal_settings/edit_sharing.jsp

Updates	Change file location
The changed files are located in the admin and student portals.	/webapps/luminis_custom_liferay_jsps/html/portlet/ portal_settings/ tabs1.jsp

Roles

Updates	Change file location
Removed the **Regular** option from the **Add** drop-down menu.	/webapps/luminis/custom_liferay_jsps/html/portlet/roles_admin/ edit_role_tabs.jsp
Removed **Add Role** from Roles main view.	/webapps/luminis/custom_liferay_jsps/html/portlet/roles_admin/ view.jsp
Removed the **Permissions** and **Delete** options for regular roles from the **Actions** menu. Disabled editing of Role name when editing regular roles.	/webapps/luminis/custom_liferay_jsps/html/portlet/roles_admin/ role_action.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/roles_admin/ edit_role.jsp

Site membership

Updates	Change file location
Removed the **Add** option from the toolbar.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ edit_site_assignments_toolbar.jsp
Removed the **Assign Users** button.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ edit_site_assignments_users.jsp
Removed the **Assign User Group** button.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ edit_site_assignments_users.jsp
Removed the **Assign Organizations** button.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ edit_site_assignments_user_groups.jsp
Removed the **Remove Membership** option from the **Action** drop-down menu. These changes apply only to Course and Institution sites.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ edit_site_assignments_organizations.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ user_action.jsp

Sites

Updates	Change file location
Site details are Read Only for Luminis courses.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/ site/details.jsp

Updates	Change file location
Added help text for the Luminis **Dynamic Group** custom field. Removed **Add** sites option. Removed **Assign Members**, **Deactivate**, and **Delete** options from the **Action** menu for each site. The changed files are located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/site/custom_fields.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/toolbar_content.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/site_admin/site_action.jsp
Removed the Search Engine Optimizations from the right-hand panel. The options are controlled by the `sites.form.update.seo` Liferay property which is overridden in the `portal-ext.properties` file for admin and portal servers. The changed files are located in the admin and student portals.	/webapps/ROOT/WEB_INF/classes/portal-ext.properties

User administration

Updates	Change file location
Removed the **Add User** option from the **Add** drop-down menu. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/toolbar.jsp
Removed the **Impersonate User** and **Deactivate** options from the **Action** pop-up menu. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user_action.jsp
Removed the **Deactivate** button and check boxes from the user list view.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/view_flat_users.jsp

Updates	Change file location
The changed file is located in the admin and student portals.	

User detail form

Updates	Change file location
These fields were disabled in the user detail portlet: • Screen Name • Email Address • First Name • Middle Name • Last Name • Suffix The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/details_user_name.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/details.jsp

Users detail form - options in the right-hand panel

Updates	Change file location
Removed several options from the right-hand panel in the user form. The options are controlled by the following Liferay properties which are overridden in the `portal-ext.properties` file for admin and portal servers: • users.form.update.main • users.form.update.identification • users.form.my.account.main • users.form.my.account.identification The updated file is located in the admin and student portals.	/webapps/ROOT/WEB-INF/classes/portal-ext.properties
Changed the heading for announcements to display Liferay. Added a message to notify the user that additional	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/announcement.jsp /webapps/luminis/ROOT/WEB-INF/classes/content/Language-ext_*.properties

Updates	Change file location
updates will only affect the Liferay announcements. The changed file is located in the admin and student portals.	

Users detail form

Updates	Change file location
These fields were disabled in the user detail portlet: • Screen Name • Email Address • First Name • Middle Name • Last Name • Suffix The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/details_user_name.jsp /webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/details.jsp
Removed several options from the user form. The options are controlled by the following Liferay properties which are overridden in the `portal-ext.properties` file for admin and portal servers: • users.form.update.main • users.form.update.identification • users.form.my.account.main • users.form.my.account.identification The updated file is located in the admin and student portals.	
Changed the heading for announcements to display Liferay. Added a message to notify the user that additional updates will only affect the Liferay announcements. The changed file is located in the admin and student portals.	/webapps/luminis/custom_liferay_jsps/html/portlet/users_admin/user/announcement.jsp /webapps/ROOT/WEB-INF/classes/content/Language-ext_*.properties

Tips for integrating Microsoft Office 365

Information to ease the installation and integration of Microsoft Office 365 with Luminis® Platform.

Supported browsers for Microsoft Office 365

Supported browsers for Microsoft Office 365 are listed on Microsoft.com.

Note: The links on the Microsoft Web site are subject to change without notice.

Office 365 Prerequisites

The integration of Luminis to Office 365 requires that a CAS server and a Shibboleth server be configured together in order for federated authentication to work.

As Luminis Platform delivers a CAS solution, Luminis Platform 5.3 includes a Shibboleth server.

If you have a CAS server integrated with a Shibboleth server, then you must configure the Luminis Platform server with the existing CAS server.

If you do not have a CAS server in place, but you do have a Shibboleth server, you must complete one of the available options to install a CAS server to provide authentication to Luminis Platform.

Install a CAS server and then Integrate this server with your Shibboleth Server

If you choose to install a CAS server on your own, this CAS server requires integration with your Shibboleth server.

You can accomplish this integration by following the steps “Designate CAS the Authentication Provider for Shib IDP” in the Shibboleth-CAS Integration guide at the following link:

https://wiki.jasig.org/display/CASUM/Shibboleth-CAS+Integration

Note: The links on this Web site are subject to change without notice.

Once you complete the integration of the CAS server and Shibboleth, you must integrate Luminis Platform with the CAS server.

Install the CAS server using the Luminis 5.3 installer

If you use the Luminis 5.3 installer to install a CAS server, an optional, preconfigured Shibboleth server is installed to work with the CAS server.

You can configure the installed Luminis CAS server with your existing Shibboleth server by following the steps “Designate CAS the Authentication Provider for Shib IDP” in the Shibboleth-CAS Integration guide at the following link:

https://wiki.jasig.org/display/CASUM/Shibboleth-CAS+Integration

Note: The links on this Web site are subject to change without notice.

Configure Shibboleth before installation

If you integrate Luminis Platform with Office 365, you must configure the Shibboleth server.

During Luminis Platform 5.3 setup, an optional Shibboleth server is automatically installed to support Microsoft Office 365 integration on the Luminis Platform CAS tier. You can set the configuration values for the Shibboleth server within your `setup.properties` file before setup, or in the `$CP_ROOT/install/resolved.properties` file after installation. Set the configuration values as follows:

```
idp.host=<host name of server hosting the Shibboleth server>
 idp.http.port=<http port of Shibboleth server> idp.https.port=<https
 port of Shibboleth server> idp.soap.https.port=<soap port
 of Shibboleth server> idp.ajp.port=<AJP port of Shibboleth
 server> idp.scope=<domain covered by Shibboleth operation>
 idp.immutableid.name=<attribute name of user immutable ID>
idp.entity.id=<URL to access the Shibboleth webapp. For example:
 https:// [your server name]/idp/shibboleth. Default value is
 objectGUID>
idp.userid.name=<Default is userPrincipalName>
```

To configure Shibboleth to start automatically with Luminis Platform, copy the `15-shib-webserver` script file from `CP_ROOT/install/bin/15-shib-webserver` to `CP_ROOT/bin/15-shib-webserver`.

For more information about configuring Microsoft Office 365 with Luminis Platform, see “Integrate Microsoft Office 365 with Luminis Platform” in the *Luminis Platform Administration Guide*.

Luminis SAML encryption and signing options

Additional options for integrating SAML encryption with Luminis Platform and the Ellucian Identity Server (EIS).

Enable SAML assertion encryption

If you do not have a signed certificate from a certificate authority, you can create a self-signed certificate to use for encrypting assertions between the EIS server and Luminis Platform.

About this task

If you do have a signed certificate, skip the step on creating the self-signed certificate and import that certificate instead of the self-signed one.

Note: The default installation of Java contains security certificate limitations that will cause the SAML implementation used in Luminis to malfunction when assertion encryption or signing is enabled. In addition, if you use the EIS (WSO2) server, you must also install the Unlimited Security policy for encryption and signing to function properly.

Procedure

1. Create a self-signed public/private key pair on Luminis Platform. We recommend you create a SAML keystore separate from the default instance used by the Luminis Tomcat server. The keytool presents you with a dialog to fill in values necessary to create the key. The private key password you enter must match the password you enter for the EIS (WSO2) keystore password. WSO2 does not provide a way to specify these passwords separately.

   ```
   keytool -genkey -keyalg RSA -alias <your_encrypt_key_alias> -keystore
   <your_luminis_saml_keystore> -storepass <your_keystore_password> -
   validity <days_before_expiration> -keysize 2048
   ```

2. Export the public/private key just created to PKCS12 format keystore and copy it to the WSO2 server. The Java keytool `-importkeystore` imports or exports the keystore, and the source and target determine how the operation works. You will be prompted for passwords for the source store and the temporary PKCS12 format store you create.

   ```
   keytool -importkeystore -srckeystore ./<your_lumins_keystore> -
   destkeystore samlkey.p12 -deststoretype PKCS12
   ```

3. Import the public/private key into the existing java keystore on the EIS IDP server. You will be prompted again for passwords. Restart EIS for the key to be available for use.

   ```
   keytool -importkeystore -srckeystore samlkey.p12 -destkeystore
   <your_idp_keystore> -srcstoretype PKCS12
   ```

4. Use JConsole or another JMX console to set the Luminis configuration values for SAML encryption:

```
saml.keystore.filename=<your_luminis_saml_keystore>
saml.keystore.password=<your_keystore_password>
saml.encryption.certificate.alias=<your_certificate_alias>
saml.encryption.certificate.password=<your_certificate_password>
saml.signing.certificate.alias=<your_certificate_alias>
```

5. Restart Luminis Platform for the values to take effect.
6. Configure the EIS Service Provider to enable assertion encryption. Select the certficate alias from the key you created in step 1 (or signed certificate) from the drop-down as the certificate to use for encryption.

Related Links

Install Java Unlimited Strength security policy on page 90

Enable SAML assertion signing

SAML offers the option to sign responses and assertions in addition to encryption.

Procedure

1. Export the public key from EIS server using keytool:

   ```
   keytool -export -alias <eis_certificate_alias> -file eis.cert -
   keystore <keystore_file> -storepass <store_password> -rfc
   ```

   This produces an encoded text file containing the public key of the EIS server's tomcat certificate. The file looks similar to:

   ```
   -----BEGIN CERTIFICATE-----
   MIIDmzCCAoOgAwIBAgIELtZVYjANBgkqhkiG9w0BAQsFADB
   +MQswCQYDVQQGEwJVUzELMAkGA1UE
   CBMCVVQxFzAVBgNVBAcTDlNhbHQgTGFrZSBDaXR5MQswCQYDVQQKEwJXVTEbMBkGA1UECxMSV2Fz
   YXRjaCBVbml2ZXJzaXR5MR8wHQYDVQQDExZkZXYtMDA3OTMuZWxsdWNpYW4uY29tMB4XDTE1MDIy
   MDE2MzQ0OVoXDTE5MTIwNjE2MzQ0OVowfjELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAlVUMRcwFQYD
   VQQHEw5TYWx0IExha2UgQ2l0eTELMAkGA1UEChMCV1UxGzAZBgNVBAsTEldhc2F0Y2ggVW5pdmVy
   c2l0eTEfMB0GA1UEAxMWZGV2LTAwNzkzLmVsbHVjaWFuLmNvbTCCASIwDQYJKoZIhvcNAQEBBQAD
   ggEPADCCAQoCggEBAI
   +gxOSSBjiuEHR88IvHMR8I0ZPDzkbgEzv5teG2FTpyp2yPRgnCRq6rUcT4
   9VATynomljH5qAta/0l/8jzLKmekYNWR5juYcQt/Me3/epJLq98gqNPw1pbRSLgWHFa/
   GCBoUGpM
   gjBm2fIkItQjGnNe4m7NzyIb6j7B4Wfy0lA15GGI2fP3dp7KZc/
   cnqkEoFYkcO9EUOFkivO1Xa8w
   +Nc9mPAKUV4a0/r4rMXMAWdgYtc/
   CtyqGdwrho00pQtjccYqVij9r3ZvmCYLt3iECXAvIqFtR4rf
   sdjd7HXEyMvk9bndgj+9xN1/
   fFD8/8XGXsbJyXyHjvEzgm24MLcreOECAwEAAaMhMB8wHQYDVR0O
   BBYEFCK3THBYAGVCZUiJ8kDF/pK43+ndMA0GCSqGSIb3DQEBCwUAA4IBAQA0Xa9O7w
   +QvZIDXmgn
   PyLL8LPEqOFg/zOQL2ObOR+WRUSn2nKIK7R1B
   +Xku4d3lUKmEMOg3hoi4Oe6L27bbouFj+DmS7dd
   Y8etj95dNPV4yJAUDyAWiKywhQvIMt9YooWMT9HIVoJkItBQ05qg/
   qPciIW2UpxK70Wp0NstHaHG
   ```

```
Cj79NcOYtOZn2TVpYO/ib5ixklMh
+3VoYyXQ0bqPyzbM82rt09dBJ0K8EDckhvNI8iiNXjpCaMpG
n7BLB1u/
ohHAMEkREPZYRJvWu3iaJuoHxeT4xyv51HE6SM1yvoH56OwJG71Rs0p0xjBVNI9fNWAU
aGtJ650AnVHvXSs7YbNZ
-----END CERTIFICATE-----
```

2. Use JConsole or another JMX provider to set this certificate in the configuration value `saml.idp.certificate`. Do NOT include the lines "-----BEGIN CERTIFICATE-----" or "-----END CERTIFICATE-----", only the encoded data.
3. Restart Luminis Platform to load the certificate into the SAML configuration.
4. On the EIS server service provider definition, check the `Enable Response Signing` and `Enable Assertion Signing` check boxes.

Install Java Unlimited Strength security policy

To use the encryption and signing options available with SAML in both Luminis and EIS, download and install the "Unlimited Strength Jurisdiction Policy Files" for Java 7.

The Java JCE (Java Cryptographic Extensions) included in the JDK ships by default with a limited strength encryption and security policy to comply with cryptographic export restrictions. The policy files are available from this URL:

http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html

Note: It is the responsibility of the customer to verify that this unlimited strength policy is legal for the country in which you are deploying the product.

After you download the `UnlimitedJCEPolicyJDK7.zip` file from the URL above, unzip the file and copy the `local_policy.jar` and `US_export_policy.jar` files to the JDK directories for Luminis Platform and EIS.

- For Luminis, the target directory is `$CP_ROOT/products/java/jdk/jre/lib/security`
- For your installation of EIS, consult the installation and configuration guides for EIS, and copy the files to the `$JAVA_HOME/jre/lib/security` directory

Restart both Luminis and EIS for these changes to take effect.

Site Analysis Worksheet

This worksheet is provided as a step-by-step checklist to help you make important decisions about how you plan to install Luminis® Platform.

The answers are used to help create the `setup.properties` files.

Do you plan to use the default port numbers for each of these components? If not, add the port numbers you plan to use next to each of the components in the above list.

Question	To consider
What is the root directory where Luminis Platform will be installed?	For example, you can install the root directory in the following location: `/opt/<luminis> [Linux/Solaris]`
Will you use an Oracle or MySQL database?	
What is the fully qualified name of the server hosting the database?	For example, the server may appear as follows: `server1.wasatch.edu`
What is the port number for the database host?	The default port number for an Oracle database might be 1521. The default port number for a MySQL database might be 3306.
What is the name of the database?	
Determine the number of servers you plan to use for installing Luminis Platform system components and which components you plan to install on each server.	For example, if you install the components on three servers, you might make the following choices: • Tier 1: LDAP and CAS • Tier 2: Portal server • Tier3: Admin server

List all of the Luminis Platform components and the fully qualified name of the server each component will be installed on.	For example, the components and servers may be organized as follows:
	• `ldap - server1.wasatch.edu`
	`cas - server1.wasatch.edu`
	`portal - server2.wasatch.edu`
	`admin - server3.wasatch.edu`

Troubleshooting

Instructions for fixing known installation issues.

Note: If you have additional questions about Luminis® Platform or associated third-party software, or you want to report defects in the system, contact Ellucian Client Support.

Insufficient memory in the /tmp directory

This error occurs when the `/tmp` directory does not contain enough memory to decompress the Luminis Platform executable files.

Not enough space left in /tmp to decompress

This error occurs when there is not enough space.

For example:

```
Uncompressing LP-5.0.0.0 ……..
Not enough space left in /tmp (567904 KB) to decompress ./
LP-5.0.0.0.2642-solaris-64bit (675464 KB)
Consider setting TMPDIR to a directory with more free space.
```

In this example, to fix the error use the `--target` option to specify a decompression directory. Substitute the local directory with enough space for `/opt/tmp`. Run the executable as follows:

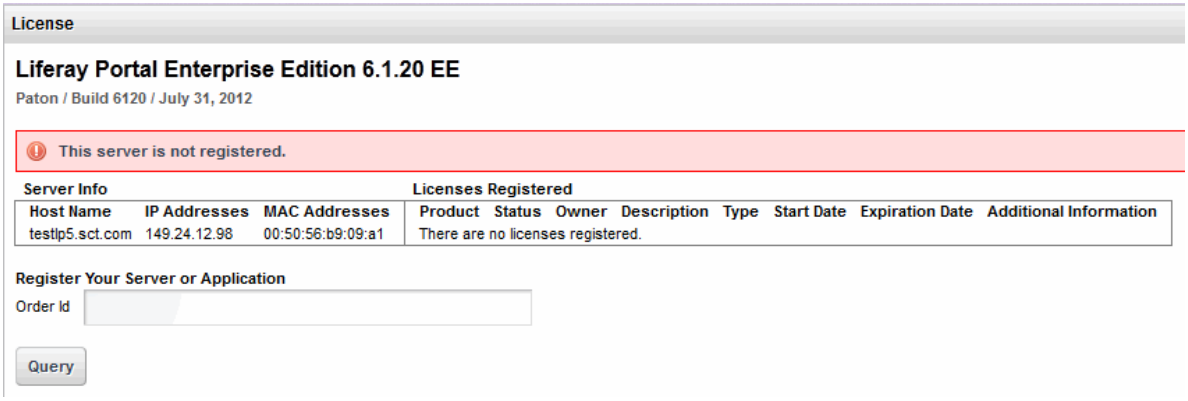
```
./LP-5.0.0.0.2642-solaris-64bit --target /opt/tmp /opt/setup.properties
```

Liferay License Key requirement error

If you do not type the path to the license key correctly, Luminis Platform will fail when you attempt to log in.

If the license key is not found, the user receives an error when they log into Luminis Platform for the first time after installation:

Figure 1: “This server is not registered” error message



If for any reason the Liferay license file needs to be updated after Luminis Platform 5.3.0 has been installed, you can install the License key by placing the license key file into the `Liferaydeploy` directory.

The Liferay deploy directories can be found in the following locations:

- Administration Portal

  `$CP_ROOT/products/liferay/liferay-admin/deploy`
- Student Portal

  `$CP_ROOT/products/liferay/liferay-portal/deploy`

“IO Error: Connection reset” errors received during 5.0.x install or during 5.1 upgrade.

Errors may occur during install or upgrade when the Linux entropy pool, used by `/dev/` random when establishing JDBC connections, runs out of random bytes.

To work around this behavior, install RNG (random number generator) tools and configure its daemon to populate entropy pool as described in For more information about this upgrade issue, see the LP5 Master Wiki on the Ellucian eCommunities Web site.

Examples of the error during 5.0.x install

Example installation error.

```
INFO LP5Installer:568 - ACTION 13754 + A,P,0 s - - %{cp.root}/install/
scripts/portal_cas_db_insert.sh

INFO ScriptExecutor:124 - java.sql.SQLRecoverableException: IO Error:
Connection reset

INFO ScriptExecutor:124 - at
oracle.jdbc.driver.T4CConnection.logon(T4CConnection.java:533)

INFO ScriptExecutor:124 - at
oracle.jdbc.driver.PhysicalConnection.<init>(PhysicalConnection.java:557)

INFO ScriptExecutor:124 - at
oracle.jdbc.driver.T4CConnection.<init>(T4CConnection.java:233)

INFO ScriptExecutor:124 - at
oracle.jdbc.driver.T4CDriverExtension.getConnection(T4CDriverExtension.java:29)

INFO ScriptExecutor:124 - at
oracle.jdbc.driver.OracleDriver.connect(OracleDriver.java:556)

INFO ScriptExecutor:124 - at
java.sql.DriverManager.getConnection(DriverManager.java:582)

INFO ScriptExecutor:124 - at
java.sql.DriverManager.getConnection(DriverManager.java:185)

INFO ScriptExecutor:124 - at
com.sghe.luminis.util.DatabaseTool.getDBConnection(DatabaseTool.java:425)

INFO ScriptExecutor:124 - at
com.sghe.luminis.util.DatabaseTool.listTables(DatabaseTool.java:586)
```

Example of the error during 5.1 upgrade

Example upgrade or patch error.

```
INFO LP5Installer:573 - ACTION 21572 + A,D,0 s - - %{cp.root}/install/
scripts/fix_liferay_portlet_roles.sh

Failed to create database connection.

java.sql.SQLRecoverableException: IO Error: Connection reset

at oracle.jdbc.driver.T4CConnection.logon(T4CConnection.java:533)

at
oracle.jdbc.driver.PhysicalConnection.<init>(PhysicalConnection.java:557)

at oracle.jdbc.driver.T4CConnection.<init>(T4CConnection.java:233)

at oracle.jdbc.driver.T4CDriverExtension.
getConnection(T4CDriverExtension.java:29)
```

```
at oracle.jdbc.driver.OracleDriver.connect(OracleDriver.java:556)
at java.sql.DriverManager.getConnection(DriverManager.java:579)
at java.sql.DriverManager.getConnection(DriverManager.java:221)
at com.sghe.luminis.install.FixLiferayPortletRoles.
main(FixLiferayPortletRoles.java:69)
Caused by: java.net.SocketException: Connection reset
at java.net.SocketOutputStream.socketWrite(SocketOutputStream.java:113)
at java.net.SocketOutputStream.write(SocketOutputStream.java:153)
at oracle.net.ns.DataPacket.send(DataPacket.java:248)
at oracle.net.ns.NetOutputStream.flush(NetOutputStream.java:227)
at oracle.net.ns.NetInputStream.getNextPacket(NetInputStream.java:309)
at oracle.net.ns.NetInputStream.read(NetInputStream.java:257)
at oracle.net.ns.NetInputStream.read(NetInputStream.java:182)
at oracle.net.ns.NetInputStream.read(NetInputStream.java:99)
at oracle.jdbc.driver.T4CSocketInputStreamWrapper.
readNextPacket(T4CSocketInputStreamWrapper.java:121)
```