



Universidad de Buenos Aires

ADMINISTRACIÓN  
DE USUARIOS Y SERVICIOS  
EN LDAP

Octubre 2010

**Resumen**

En este documento se describe brevemente la estructura de directorio LDAP sugerida por el CCC para la UBA y los pasos a seguir para la creación de usuarios y servicios.

## 1. ¿Cómo administrar usuarios y servicios?

Si bien se está trabajando para contar con un sistema para la administración de usuarios y servicios de la UBA, hoy la forma más amigable de hacer estas tareas es a través de la interfaz web de `ldaps.ccc.uba.ar`.

Cada dependencia cuenta con una cuenta de administrador, mediante la que puede acceder a la rama de su dependencia. Para ingresar al sistema con esta cuenta:

Login: `cn=admin,dc=<dependencia>,dc=uba,dc=ar`  
Clave: `123456`

La clave mencionada fue configurada con ese mismo valor para todos, pero **debe ser modificada** una vez que se ingresa al sistema. Esta cuenta tiene permisos totales sobre los usuarios cargados bajo la dependencia a la que pertenece, de ahí la importancia de modificar la clave, ya que de otra manera se estarían desprotegiendo los datos de los usuarios.

## 2. ¿Cuál es la estructura sugerida?

En la figura 1 se presenta la estructura general sugerida para el directorio LDAP. Las entradas cargadas en una dependencia están distribuidas, de acuerdo al tipo de información que sea y a la finalidad que tengan, en cuatro ramas: Aplicaciones, Usuarios, Skel y Configuraciones.

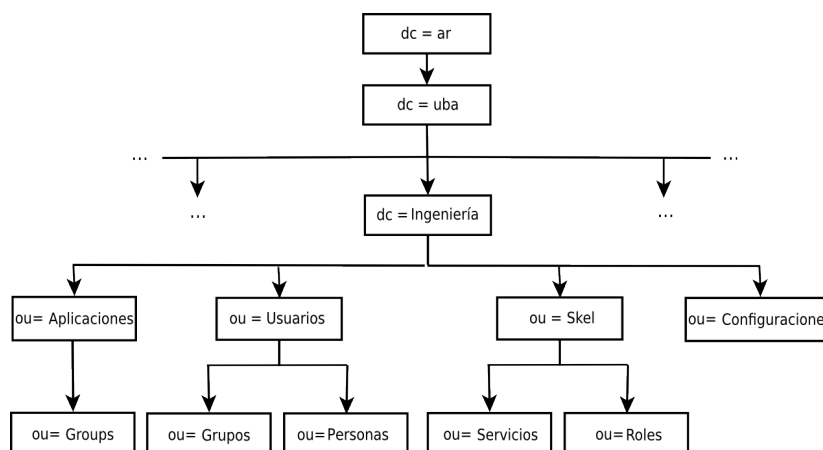


Figura 1: Estructura de directorio LDAP

En la figura 2 está expandida la rama donde están cargados los usuarios de una dependencia. Un usuario en particular se identifica unívocamente con un uid, y tiene asociados roles y servicios. Un rol o servicio nuevo que adquiriera el usuario es cargado como una nueva entrada, bajo la rama que corresponda.

## 3. ¿Cómo crear un usuario o servicio?

Para crear una entrada nueva la interfaz phpLDAPAdmin cuenta con la opción **Crear un objeto hijo** (figura 3). Al ejecutar esta operación se debe elegir una plantilla (figura 4). Existen varias predeterminadas, pero tanto para usuarios como para servicios se debe seleccionar **Default**.

Los usuarios (que correspondan a personas físicas) se crean dentro de la rama Personas (`ou=Personas,ou=Usuarios,dc=<dependencia>,dc=uba,dc=ar`), y utilizan el objectClass **persona** (figura 5).

Una vez elegido el objectClass aparecen todos los posibles atributos a completar, separados en obligatorios y optativos (figura 6). Es importante elegir un atributo RDN, que formará parte del DN que identificará unívocamente al objeto (en este caso, al usuario). Para los usuarios se sugiere la identificación mediante el atributo uid.

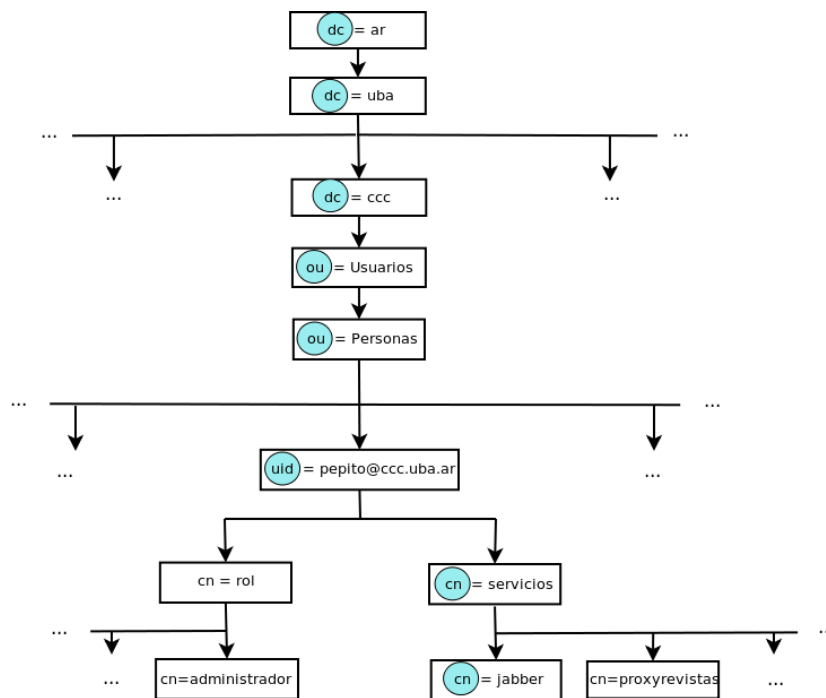


Figura 2: Estructura para los usuarios y servicios

Para poder asociarle servicios a las personas primero se debe crear la rama servicios. Los pasos son los mismos que para la creación de usuarios, salvo que el contenedor de la entrada ahora es el usuario ya creado en lugar de la rama personas (figura 5), es decir, `uid=<usuario>,ou=Personas,ou=Usuarios,dc=<dependencia>,dc=uba,dc=ar`, y que el objectClass ahora es **servicios**. Este objeto cuenta con un único atributo que debe ser completado también con el valor servicios. De esta manera queda creada para un usuario en particular la rama servicios

`(cn=servicios,uid=<usuario>,ou=Personas,`

`ou=Usuarios,dc=<dependencia>,dc=uba,dc=ar)`

donde se asociarán todos los servicios con que cuente el usuario.

Para la creación de servicios se siguen los mismos pasos que para creación de para la creación de usuarios, salvo que el contenedor es la rama servicios. El atributo elegido como RDN para todos los servicios es el `cn`, que lleva el valor que el mismo nombre del servicio, y para cada caso los atributos obligatorios y optativos variarán de acuerdo al servicio que sea. Por ejemplo, en la figura 7 se muestra la creación del objeto jabber, necesaria para brindarle el servicio de Jabber a un usuario. El uid (User name) de este objeto puede diferir del uid que identifica al usuario.

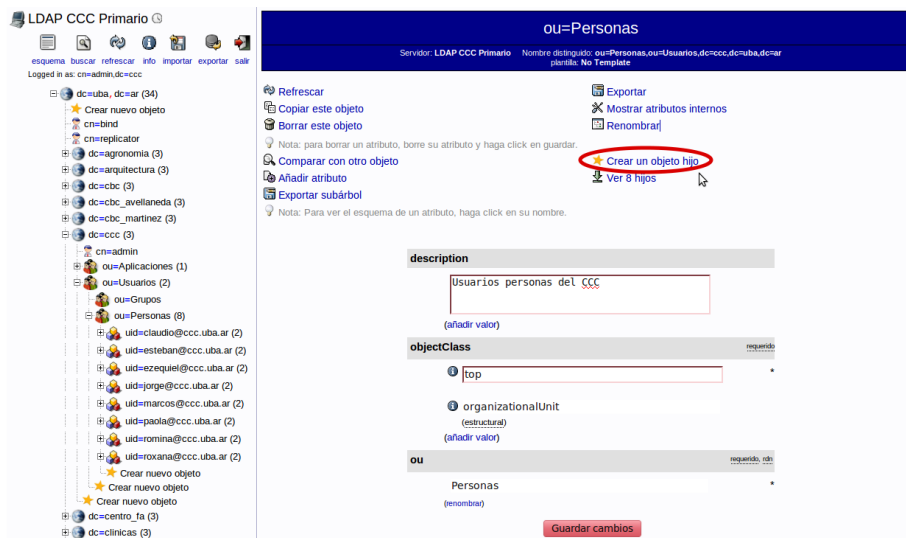


Figura 3: Creación de una nueva entrada



Figura 4: Plantilla a utilizar para usuarios y servicios

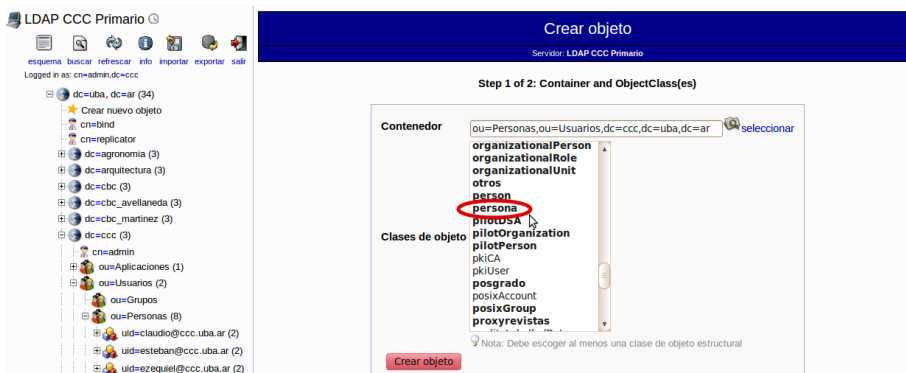


Figura 5: ObjectClass para los Usuarios

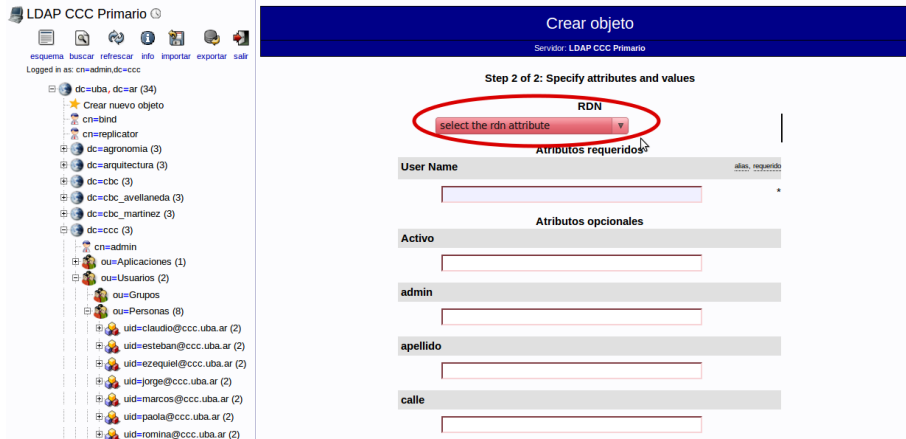


Figura 6: Formulario con atributos a completar

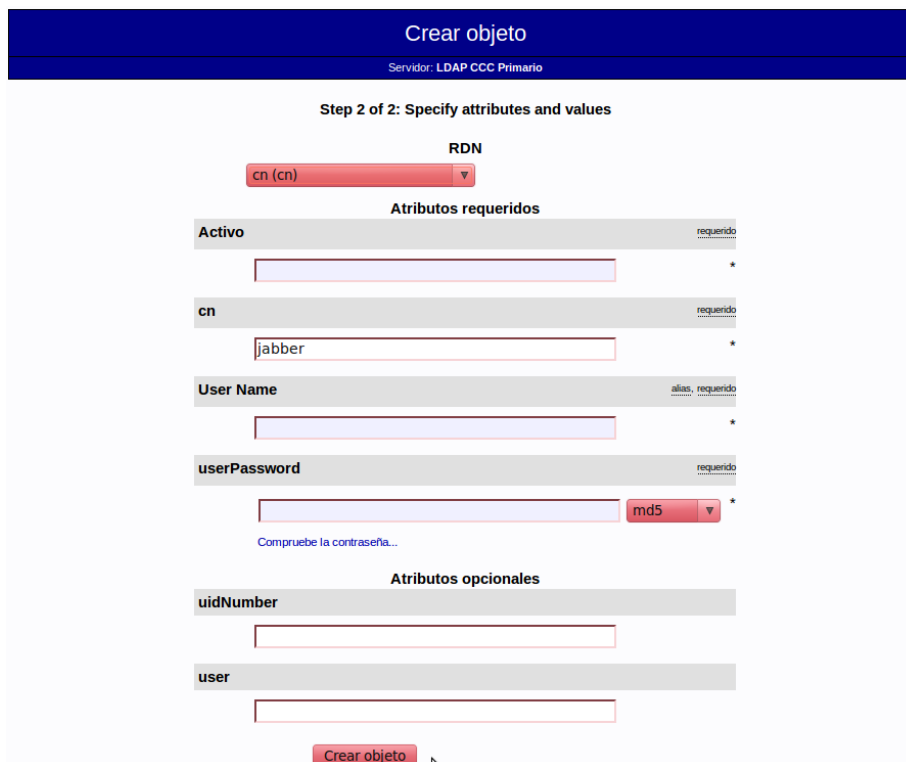


Figura 7: Object Class jabber