UT5 - SERVICIOS DE RED IMPLICADOS EN EL DESPLIEGUE DE UNA APLICACIÓN WEB

Despliegue de Aplicaciones Web 2° DAW

Servicios de directorio: introducción

- Servicios de directorio
  - Sistema software que ofrece servicios de gestión y acceso a un conjunto de información (**directorio**).
  - Búsqueda de información basada en nombres.
- Pero… según la definición, ¿qué podría ser servicios de directorio?
  - ¿Un sistema de ficheros?
  - ¿Las bases de datos?
  - ¿El DNS?
- Se suele utilizar el término "servicio de directorio" para referirse a los servicios basados en los estándares **X.500**

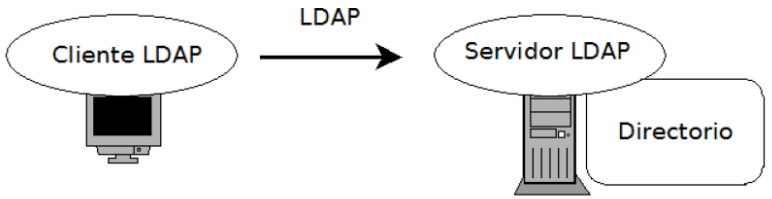
Servicios de directorio: X.500

- Conjunto de estándares sobre servicios de directorio definidos por la UIT (Unión Internacional de Telecomunicaciones) http://www.itu.int/es
- Define:
  - **Protocolos**
    - DAP (Directory Access Protocol)
    - DSP (Directory System Protocol)
    - DISP (Directory Information Shadowing Protocol)
    - DOP (Directory Operational Bindings Management Protocol)
  - **Modelos de datos**
- Características:
  - Arquitectura cliente/servidor.
  - Organización jerárquica de los datos.
  - Estructura flexible.
  - Muchas lecturas y pocas escrituras → Optimizados para lecturas.
  - Alto rendimiento (miles de accesos por segundo).
  - Distribuidos.

LDAP: introducción

- X.500
  - Protocolo DAP para acceder a los servicio de directorio a través de una red.
  - DAP se basaba en la pila de protocolos **OSI**.
- LDAP
  - Definido por la ITU con el objetivo de ofrecer la misma funcionalidad que DAP pero sobre la pila de protocolos **TCP/IP**.
  - Simplifica DAP.
- La terminología X500 y LDAP es similar.
- Versiones
  - LDAPv2 (obsoleto)
  - LDAPv3.
    - Remplaza a LDAP v2, es más rápido y tiene más opciones de autenticación
    - Soporta SSL/TLS y Certificados digitales X.509.

LDAP: introducción

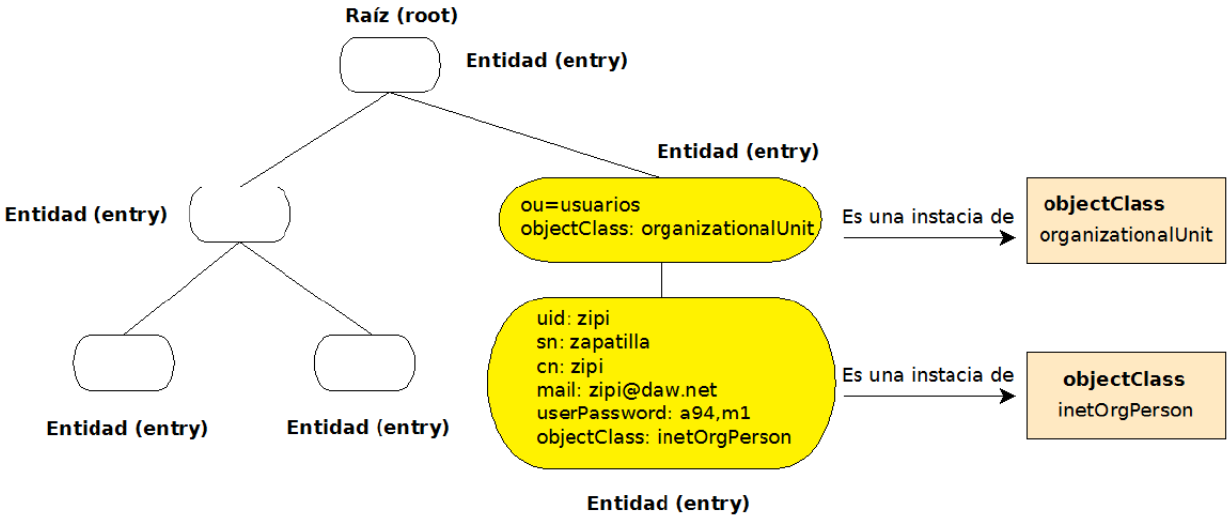


- Características:
  - LDAP es sólo un protocolo que define cómo acceder a un directorio de datos
  - Necesariamente, también define y describe:
    - Cómo los datos son **representados** en el directorio.
    - Cómo los datos son **cargados (importados) y exportados** en/del directorio (LDIF).
  - LDAP **NO define** cómo los datos son almacenados y manipulados.
  - Optimizado para consultas.
  - No transaccional (no hay roolback).

- **Modelo de información** (**modelo de datos**)
  - Define la *estructura* de la información almacenada en el directorio.
- **Modelo de nombrado**
  - Cómo *nombra* y se identifica a la información almacenada en el directorio.
- **Modelo funcional**
  - *Operaciones* sobre la información: búsquedas, lecturas, escrituras y modificaciones.
- **Modelo de seguridad**
  - Control de *acceso*.
  - *Quién y qué* puede hacer en el directorio.

LDAP: modelo de datos (DIT)

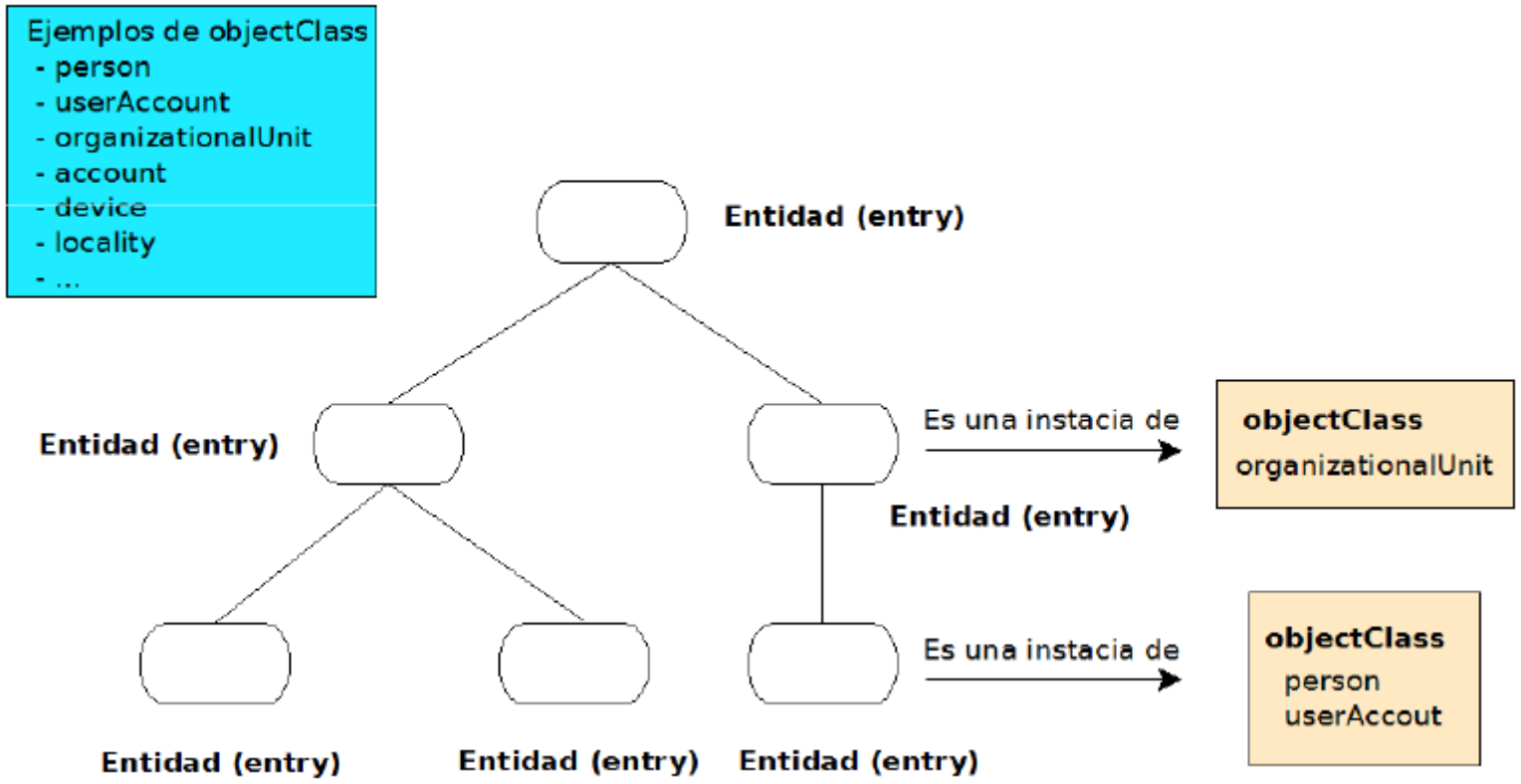
- La información de un directorio LDAP está formada por un conjunto de objetos - entradas (***entry***) organizadas jerárquicamente.
- La estructura resultante se denomina DIT (***Data Information Tree***).
- La entrada más alta del árbol se denomina normalmente raíz (root).



LDAP: modelo de datos (DIT)

objectClass

- Cada entidad (entry) es una instancia de una o varias clases (objectClass).



LDAP: modelo de datos (DIT)

- Cada ***objectClass*** tiene un **nombre** y define uno o varios **atributos** y sus tipos de datos.

Nombre: **account**
Atributos:
- **userid (obligatorio) (*)**
- **description**
- **localityName**
- **organizationName**
- …

Nombre: **person**
Atributos:
- **cn (common name) (*)**
- **sn (surname) (*)**
- **telephoneNumber**
- **organizationName**
- …

- Los objectClass son, por lo tanto, colecciones de **atributos obligatorios y opcionales.**
- Los objectClass puede formar parte de una jerarquía y heredar los atributos de sus padres.
- Se definen en esquemas.

LDAP: modelo de datos (DIT)

- Los objectClass pueden ser de tipo:
  - **STRUCTURAL**
    - Usados para crear entidades.
  - **AUXILIARY**
    - Añadidas en entidad existentes (que tienen al menos un objectClass STRUCTURAL)
  - **ABSTRACT**
    - Para definir jerarquías de objectClass

- Las entidades:
  - Deben pertenecer a un (uno y sólo uno) STRUCTURAL objectClass.
  - Pueden pertenecer a uno o varios AUXILIARY objectClasses.
  - Pueden pertenecer sólo a un ABSTRACT objectClass

LDAP: modelo de datos (DIT)

Atributos

- En función de los objectClass a los que pertenezcan las entidades tendrán valores para los atributos
- En las entidades se definen el atributo especial *objectClass* que contiene como valor el/los objectClass(es) a los que pertenece la entidad

- Los atributos son miembros de uno o más objectClass(es).
- Cada atributo define un tipo de datos que puede contener.
- Los atributos pueden ser opcionales (MAY) o obligatorios (MUST) dependiendo de la objectClass. Un atributo puede ser obligatorio en una objectClass y opcional en otra.
- Los atributos puede tener uno o varios valores.
- Los atributos tienen nombres y a veces abreviaturas.
  - Ejemplo: cn es una abreviatura de commonName.

LDAP: modelo de datos (DIT)

Esquemas

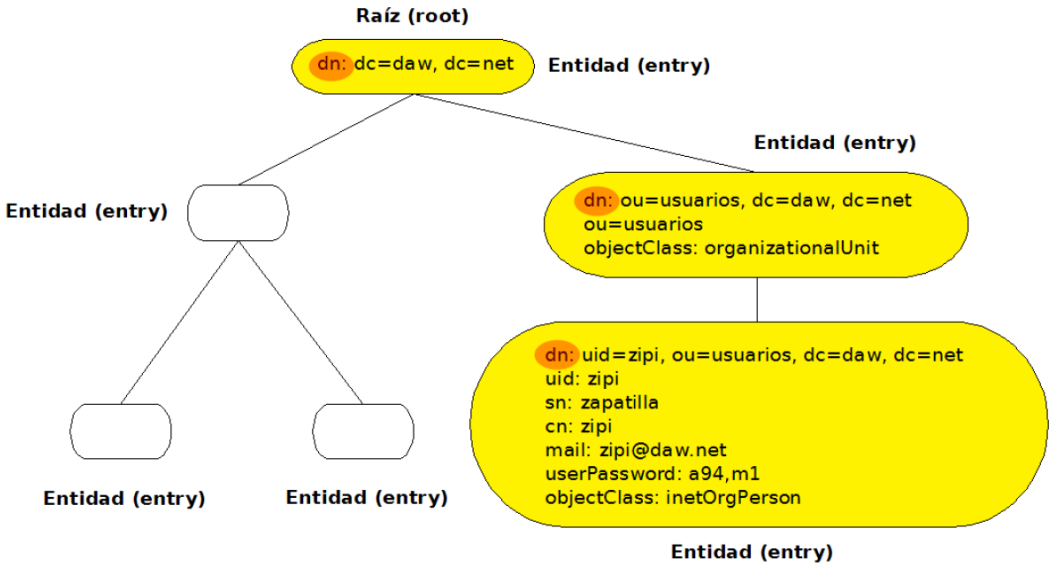
- Los esquemas (schemas) son paquetes que definen:
  - objectClass y atributos.
  - Un atributo definido en un esquema puede ser usado por objectClass de otros esquemas.
  - Podemos crear nuestros esquemas propios con los objectClass que nos interesen.

```
objectclass ( 2.5.6.6 NAME 'person' SUP top STRUCTURAL
  MUST ( sn $ cn )
  MAY ( userPassword $ telephoneNumber $ seeAlso $ description ) )

objectclass ( 2.5.6.7 NAME 'organizationalPerson' SUP person STRUCTURAL
  MAY ( title $ x121Address $ registeredAddress $ destinationIndicator $
   preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
   telephoneNumber $ internationaliSDNNumber $
   facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $
   postalAddress $ physicalDeliveryOfficeName $ ou $ st $ l ) )
```

LDAP: modelo de nombrado

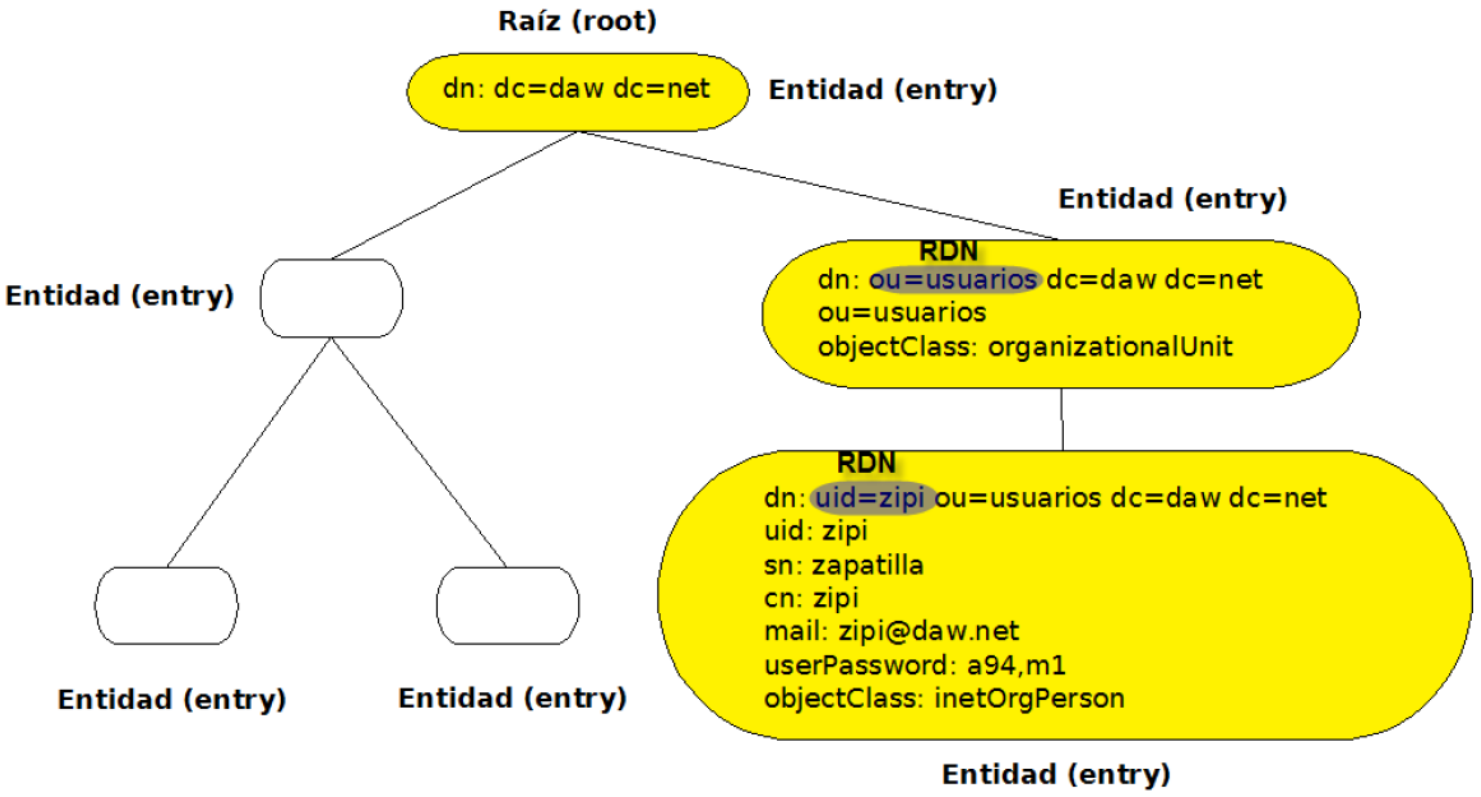
- **Modelo de nombrado**: define cómo se nombra y se identifica a la información almacenada en el directorio.
- La entradas se organizan el DIT en base a su **DN (*Distinguished Name*)**.



- DN (Distinguished Name): nombre único que identifica de forma unívoca a una entrada.
- Secuencias de **RDNs** (Relative Distinguished Names) y cada RDN se corresponde con una rama del DIT partiendo de la raíz hacia la entrada dentro del directorio.

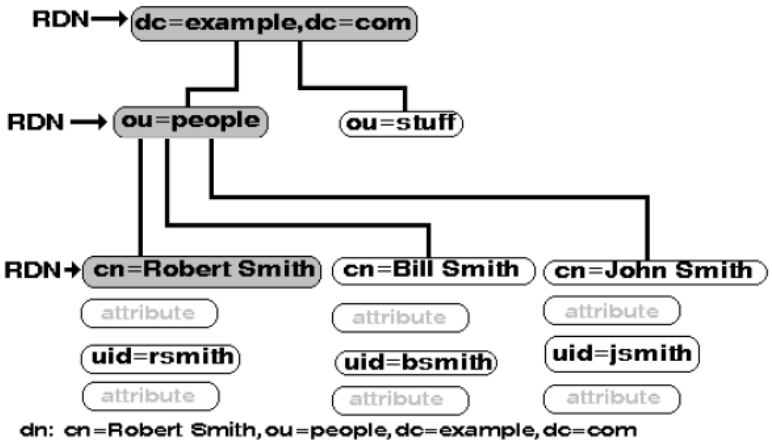
LDAP: modelo de nombrado

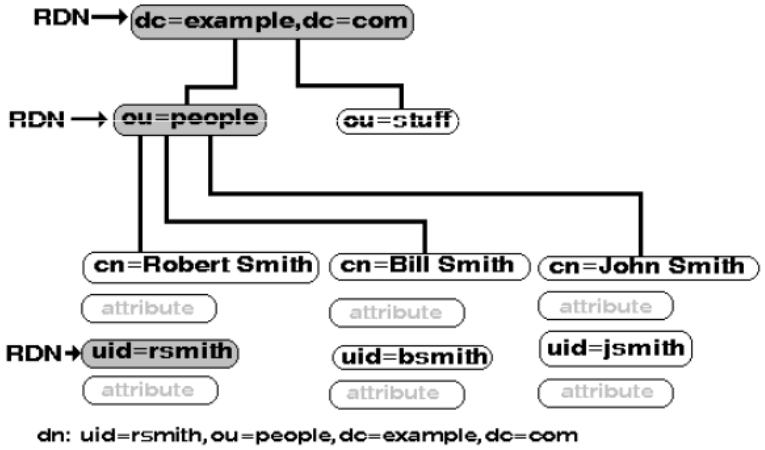
- DN = camino hasta la raíz + RDN (relative DN)



LDAP: modelo de nombrado

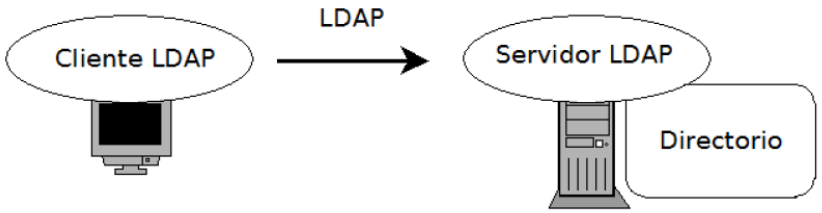
- Se puede elegir qué atributo de la entidad formará el RDN teniendo en cuenta que el DN **debe ser único**





LDAP: modelo de funcionamiento

- Arquitectura cliente-servidor. Puerto 389/TCP



- **Operaciones** sobre el servidor LDAP.
  - Consulta
    - Búsqueda y lectura (search)
  - Actualización
    - Añadir (add)
    - Borrar (delete)
    - Modificar (modify)
    - Renombrar un dn (rename).
  - Autenticación y control (bind, unbind, …)

LDAP

Usos

- Representar y almacenar información sobre organizaciones (departamentos, usuarios, equipamiento, …)
- Servicios centralizados
  - Usuarios/grupos.
  - Autenticación (Sistema, FTP, Correo, Web, WiFi, …)
  - Perfiles de usuarios
  - …

- **Servidores LDAP**
  - OpenLDAP
  - Active Directory (AD) (Microsoft)
  - Apache Directory.
  - Oracle Internet Directory
  - RedHat Directory Server
  - IBM Directory Server
  - Open DS
  - 389 Directory Server
  - …

- **Clientes LDAP**
  - Apache Directory Studio
  - JXplorer
  - phpLDAPadmin
  - LDAPExplorerTool
  - Fusiondirectory
  - OpenLDAP Tools
  - …

LDAP: autenticación y autorización en Apache

- Deberemos utilizar el módulo **authnz_ldap**
- Algunas directivas a utilizar serán las siguientes:
  - AuthType
  - AuthBasicProvider
  - AuthzLDAPAuthoritative
  - AuthName
  - AuthLDAPURL
  - AuthLDAPBindDN
  - AuthLDAPBindPassword
  - Require

LDAP: autenticación y autorización en Apache

- **Configuración**
  - Configurar el servidor LDAP con los usuarios y contraseñas adecuados.
  - Configurar la autenticación en Apache:

```
<Directory /var/www/html/profesor>
    Options Indexes
    AllowOverride None
    AuthType Basic
    AuthBasicProvider ldap
    AuthName "Introduce tu usuario y password"
    AuthLDAPURL "ldap://localhost/dc=daw,dc=com?uid?sub?(objectClass=*)"
    AuthLDAPBindDN "cn=admin,dc=daw,dc=com"
    AuthLDAPBindPassword usuario@1
    Require ldap-group cn=griegos,ou=grupos,dc=daw,dc=com
</Directory>
```

LDAP: autenticación y autorización en Apache

- ¿Cómo funciona?
  1. Se genera un filtro de búsqueda combinando el atributo y el filtro proporcionados en la directiva **AuthLDAPURL** con el nombre de usuario introducido por el usuario.
  2. Se establece una conexión al servidor LDAP con el usuario y password definidos en **AuthLDAPBindDN** y **AuthLDAPBindPassword** y se realiza una búsqueda con el filtro generado anteriormente. Si la búsqueda no retorna una entrada exactamente se deniega el acceso
  3. Se obtiene la entrada y se realiza una conexión al servidor usando el DN de la entrada y la password introducida por el usuario. Si la conexión es posible se permite el acceso y si no se deniega
  - Se utilizan las directivas Require para determinar si el usuario es autorizado o no.
    - Require ldap-user
    - Require ldap-dn
    - Require ldap-group
    - Require ldap-attribute
    - Require ldap-filter
  - Cada una de ellas tiene otro conjunto de directivas asociadas para controlar su comportamiento.

Hoja05_LDAP_04

LDAP: autenticación y autorización en Tomcat

1. Configurar el **Realm** en el ámbito que se considere más adecuado (<Engine>, <Host>, <Context>, …)

```
<Context>
    <Realm  className="org.apache.catalina.realm.JNDIRealm"
            connectionURL="ldap://localhost:389"
            userPattern="uid={0},ou=usuarios,dc=daw,dc=com"
            roleBase="ou=grupos,dc=daw,dc=com"
            roleName="cn"
            roleSearch="(uniqueMember={0})"
    />
</Context>
```

2. Proteger el recurso (en el descriptor de despliegue **web.xml** de la aplicación) (***security-constraint***)

```
<security-constraint>
    <web-resource-collection>
        <web-resource-name>JNDIRealm</web-resource-name>
        <url-pattern>/*</url-pattern>
    </web-resource-collection>
    <auth-constraint>
        <role-name>romanos</role-name>
    </auth-constraint>
</security-constraint>
```

3. Configurar el tipo autenticación (en el descriptor de despliegue **web.xml** de la aplicación) (***login-config***)