

Despliegue de aplicaciones web

UT2. Servicios de red implicados en el despliegue



1. Despliegue de aplicaciones

Despliegue (Deployment)

- Las aplicaciones web requieren ser desplegadas.
- El despliegue serán las tareas que realizaremos para finalizar la fase de implementación/desarrollo, y trasladar nuestra aplicación probada y terminada a la fase de implantación/producción.



- Esto requiere la instalación y configuración de diferentes servicios de red o web que vaya a utilizar nuestra aplicación.

1. Despliegue de aplicaciones

Despliegue (Deployment)

- El despliegue es una tarea repetitiva con muchas acciones, que si no están automatizadas nos llevaran a cometer errores. Todas las dependencias deben ser actualizadas y los archivos de configuración copiados correctamente, sino la aplicación web desplegada no funcionará correctamente.
- El objetivo es automatizar lo máximo posible y comprobar que todo está funcionando correctamente.

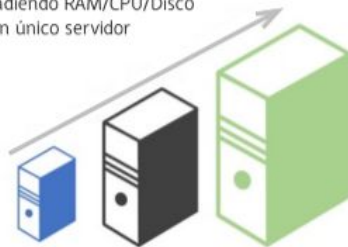
1. Despliegue de aplicaciones

1.1 Introducción al despliegue de aplicaciones web

- Las aplicaciones web quieren dar servicio concurrente a todos los usuarios, que pueden ser millones. Esto nos lleva a trabajar con sistemas distribuidos que forman un cluster de equipos coordinados. → esto aumenta la complejidad del despliegue
- Otro punto importante a considerar es la escalabilidad del sistema

Escalado vertical

Aumentar la capacidad
añadiendo RAM/CPU/Disco
a un único servidor



Escalado horizontal

Aumentar la capacidad añadiendo servidores



1. Despliegue de aplicaciones

1.2 Alojamiento interno (in-house)

- es el modelo clásico
- la organización aloja todos sus servidores, aplicaciones y servicios dentro de sus instalaciones
- la tendencia actual es trasladar total o parcialmente esta infraestructura propia a una nube pública, de esta forma será la infraestructura de computación en la nube la que administre la parte de infraestructura que contratemos.

1. Despliegue de aplicaciones

1.3 Computación en la nube

- este concepto surge en 2006
- AWS fue la primera en ofrecer este servicio de virtualización de servidores a gran escala EC2 y otro de almacenamiento masivo S3. El objetivo era ofrecer un servicio de CPD virtualizado y en pago por uso.
- las ventajas son:
 - escalabilidad
 - ausencia de inversión inicial
 - menor coste
- los servicios de computación en la nube se dividen en :
 - IaaS → nos encargaremos nosotros de administrar todo el equipo
 - SaaS → solo nos encargaremos de nuestra aplicación web
 - PaaS → es la opción intermedia que nos permitirá también desarrollar desde la nube

1. Despliegue de aplicaciones

1.3 Computación en la nube



Gestionado por
tu equipo



Gestionado por
tu proveedor

Internamente	IaaS	PaaS	SaaS
Aplicaciones	Aplicaciones	Aplicaciones	Aplicaciones
Datos	Datos	Datos	Datos
Tiempo de ejecución	Tiempo de ejecución	Tiempo de ejecución	Tiempo de ejecución
Middleware	Middleware	Middleware	Middleware
SO	SO	SO	SO
Virtualización	Virtualización	Virtualización	Virtualización
Servidores	Servidores	Servidores	Servidores
Almacenamiento	Almacenamiento	Almacenamiento	Almacenamiento
Equipos de red	Equipos de red	Equipos de red	Equipos de red

1. Despliegue de aplicaciones

1.3 Computación en la nube

- IaaS - Infraestructura como servicio
- PaaS - Plataforma como servicio
- SaaS - Software como servicio

IaaS - Infraestructura como servicio

Planes de hosting tradicionales como cloud hosting o el de servicio dedicado que se puede contratar de los IaaS hoy en día.

- En cloud hosting el proveedor del servicio nos facilita una consola de administración para que podamos gestionar el dominio y nuestras webs.
- En el servicio dedicado nos da acceso a una máquina sin ninguna ayuda.

Los principales proveedores IaaS son: AWS, Azure y Google

1. Despliegue de aplicaciones

1.3 Computación en la nube

PaaS - Plataforma como servicio

- El desarrollador se olvida del hardware y del sistema operativo, se preocupa solo de la aplicación y los datos por desplegar
- Se utiliza para desplegar aplicaciones web, bases de datos y middleware
- Un ejemplo de esto es Heroku

SaaS - Software como servicio

- El proveedor del servicio nos da soporte a todos los niveles
- El usuario se limita a usar la aplicación contratada
- Es lo opuesto a in-house
- Solo tenemos que contratar algún servicio (ERP, CMS, e-commerce, ...) a una de las muchas empresas de hosting

DNS

Domain Name System



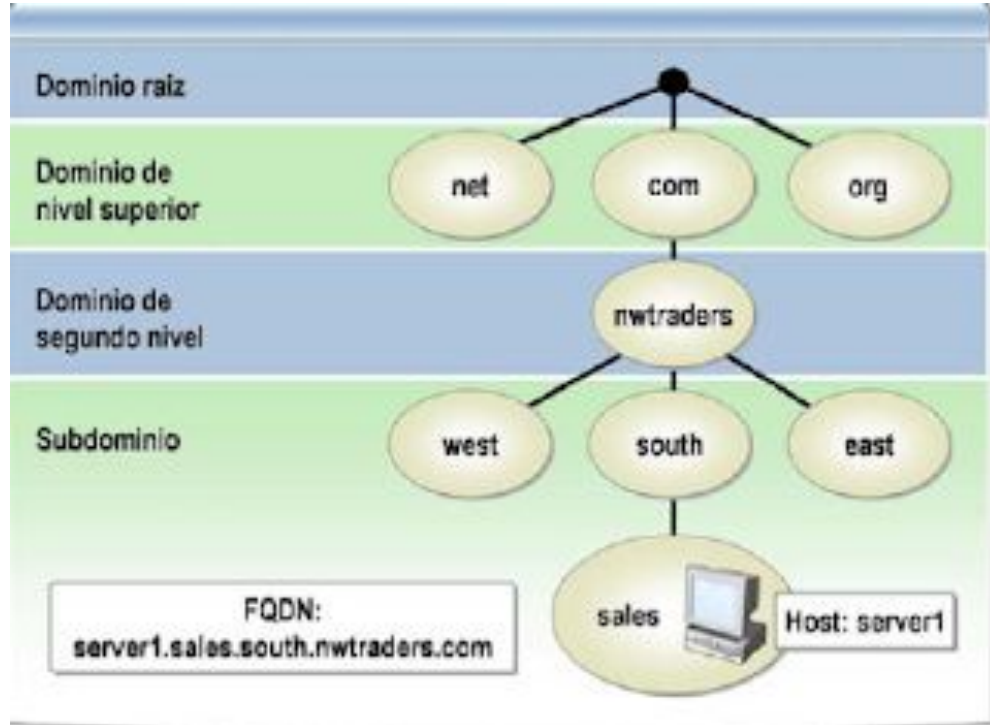
2. Sistema de nombres de dominio (DNS)

- DNS, o sistema de nombres de dominio (*Domain Name System*), es un protocolo de nivel de aplicación de la familia de protocolos TCP/IP que establece las normas de funcionamiento de un servicio de nombres jerárquico basado en dominios.
- DNS se centra en la administración de nombres de dominio, dando el servicio de resolucion su correspondiente IP.
- Se realiza de forma distribuida entre todos los servidores DNS.
- Un dominio o nombre de dominio es el nombre que identifica a un sitio web. Este dominio tiene que ser único en Internet.

2. Sistema de nombres de dominio (DNS)

2.1. Espacio de nombres de dominio

El servicio DNS se compone de una base de datos distribuida en la que se almacenan las asociaciones nombres de dominio y su dirección IP. Cada nombre de dominio es una rama en un árbol invertido llamado espacio de nombres de dominio.



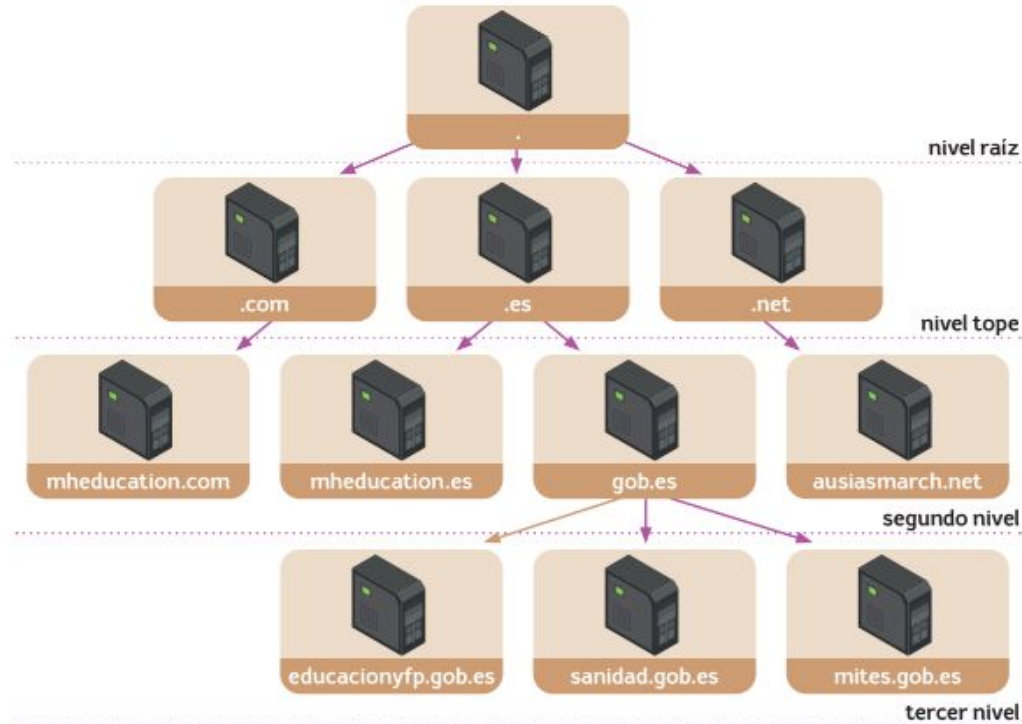
2. Sistema de nombres de dominio (DNS)

2.2. Dominios por niveles

- Dominio raíz
- Los dominios de primer nivel se llaman TLD (Top Level Domain), son los que se encuentran por debajo del dominio raíz. Todos tienen su centro de información de red propio NIC (Network Information Center). Los servidores de este nivel contiene información relativa a los servidores de dominio de segundo nivel.
- Los dominios de segundo nivel, están bajo los TLD, están registrados a nombre de una determinada entidad. Una entidad que desee tener un dominio debe registrarlo ante un registrador oficial autorizado por ICANN.
- Los subdominios forman parte de un dominio de segundo nivel.
- El nombre de dominio completo se llama FQDN (Fully Qualified Domain Name).

2. Sistema de nombres de dominio (DNS)

2.2. Dominios por niveles



2. Sistema de nombres de dominio (DNS)

2.3. Tipos de dominio

- Los dominios de primer nivel que ICANN establece en un principio eran dos:
 - ccTLD: dominios geográficos o de país (.es, .fr, ...)
 - gTLD: dominios de primer nivel genéricos:
 - sTLD: dominios patrocinados
 - uTLD: dominios no patrocinados (.com, .net, .org)
 - iTLD: dominios para temas lingüísticos (.cat, .gal, .eus, ...)
- Existe un dominio genérico especial para gestionar la infraestructura .arpa. sirve para realizar resoluciones inversas. Utilizamos dos:
 - in-addr.arpa para IPv4
 - ip6.arpa para IPv6.

2. Sistema de nombres de dominio (DNS)

2.4. Resoluciones directas y resoluciones inversas

- Existen dos tipos de resoluciones: iterativa y recursiva
- Una resolución directa es cuando se obtiene la dirección IP a partir del nombre DNS
- Una resolución inversa es cuando se obtiene el nombre DNS a partir de la dirección IP. El dominio .arpa es el especial de infraestructura para esta resolución.

2. Sistema de nombres de dominio (DNS)

2.5. Zonas primarias y zonas secundarias

- Si una zona se almacena en un servidor DNS, se dice que el servidor tiene autoridad sobre ella.
- Un servidor puede tener autoridad sobre una o varias zonas.
- La zona está constituida por varios registros. Los registros sirven para resolver, aunque también tienen otras funciones como indicar cuales son los servidores DNS de la zona.
- El caso más habitual es que una zona resuelva los nombres de un dominio, es decir que en un dominio haya solo una zona. Una zona puede contener información sobre un dominio y sus subdominios.

2. Sistema de nombres de dominio (DNS)

2.6. Tipos de servidores DNS

- Existen varios tipos de servidores y un mismo servidor puede ser a la vez de varios tipos.
- Los servidores DNS pueden ser de cuatro tipos:
 - Servidores autoritarios
 - primario (maestro)
 - secundario (esclavo)

Un servidor primario puede editar la zona.

Un servidor secundario obtiene la información de la zona primaria mediante transferencia de zona.

2. Sistema de nombres de dominio (DNS)

2.6. Tipos de servidores DNS

- servidores no autoritarios
 - caché
 - reenviador

Un servidor caché almacena temporalmente las últimas consultas realizadas.

Un servidor reenviador es el que ha sido designado por otro servidor para resolver nombres fuera del dominio en el que se encuentran.

2. Sistema de nombres de dominio (DNS)

2.7. Registros de recursos

<Propietario>	[<TTL>]	<Clase>	<Tipo>	<RDatos>
mcgrawDAW.lan.	0	IN	A	192.168.123.123

En cada campo se utilizan los siguientes parámetros:

- **Propietario:** contiene un nombre FQDN para especificar que el nombre completo del propietario llevará un punto al final. Si no es completo, es decir, es un nombre relativo, entonces no termina en ese punto final. En este caso, se añadirá el nombre de la zona que representa con el carácter @. Si dejamos una cadena vacía, el propietario tomará el valor del registro que se haya declarado anteriormente.
- **TTL (*Time To Live*):** marca el tiempo de vida del registro, que es un campo opcional. No se suele configurar, por lo que se deja vacío o con un valor 0, que indica lo mismo, como en nuestro ejemplo. Los servidores caché lo utilizan para saber cuándo tienen que actualizar los registros de las zonas descargadas y guardadas en su memoria. Dicho tiempo se puede expresar en días [d], horas [h], minutos [m] o segundos [s].
- **Clase:** indica la familia de protocolos para usar. Para nosotros siempre será "IN", ya que representa la familia de protocolos TCP/IP que se utiliza en Internet.
- **Tipo:** puede variar en función del campo Clase. En nuestro caso, para la clase IN en el siguiente subapartado de tipos de registros veremos cuáles están admitidos y para qué sirven.
- **RDatos:** es el registro con el dato por devolver en la consulta que nos planteen para este recurso.

2. Sistema de nombres de dominio (DNS)

2.8. Tipos de registros de recursos

Nombre de recurso	Tipo de registro	Función
Inicio de autoridad <i>(Start Of Authority)</i>	SOA	Debe ser el primer registro de la zona. Identifica al servidor como autoritario sobre dicha zona y le da los parámetros de configuración.
Servidor de nombres	NS	Indica qué equipo ejerce como servidor de nombres en esta zona.
Dirección IPv4	A	Sirve para obtener su dirección IPv4 del nombre de dominio FQDN de un equipo que se ha solicitado resolver.
Dirección IPv6	AAAA	Igual que la A, pero para direcciones IPv6. Cada A representa 32 bits, es decir, 4 bytes. Las direcciones IPv6 son de 128 bits. Las especificaciones de las extensiones para soportar IPv6 en DNS están recogidas en el RFC 1886. Disponible desde BIND 9.
Puntero	PTR	Sirve para las búsquedas inversas, donde, dada una IP, nos devolverá el nombre de dominio FQDN.

2. Sistema de nombres de dominio (DNS)

2.8. Tipos de registros de recursos

Correo	MX	Permite saber qué máquinas van a gestionar la correspondencia de correo de nuestro dominio; puede haber varios e indica prioridades a la hora de configurarlos. El valor 0 es el prioritario.
Nombres canónicos	CNAME	Nos permite asignar alias a nombres reales de máquina. Muy utilizado para renombrar dichas máquinas con los servicios que atienden; por ejemplo, <code>www</code> para el servicio de web.
Texto	TXT	Nos permite almacenar cualquier tipo de información. Se suele usar para dejar la marca identificativa de nuestro dominio cuando usamos servicios externos como el correo electrónico de terceros.
Servicio	SRV	Indica la ubicación de los servidores para un determinado servicio.

Ponte a prueba 2

¿Qué información podemos obtener de estas órdenes?

Veamos qué información podemos obtener de las diferentes herramientas que tenemos a nuestro alcance para lanzar todo tipo de consultas DNS desde nuestro equipo como clientes DNS.

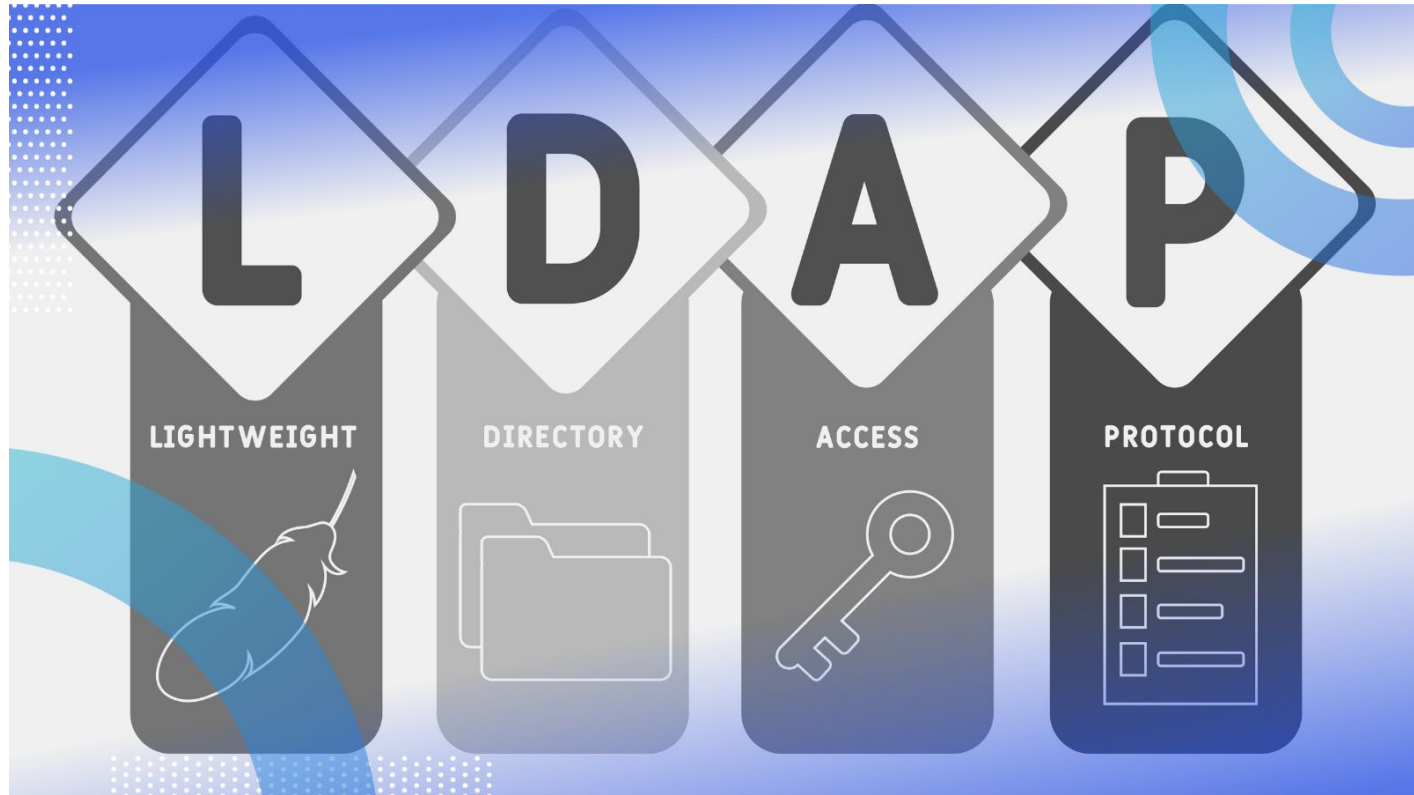


Las órdenes más utilizadas desde el terminal para realizar consultas DNS son:

- **ping** o **fping** con funciones extra
- **host**
- **dig**
- **nslookup**

Investiga su sintaxis y trata de hacer consultas directas con todas las órdenes para ver qué te puede aportar cada una de ellas.





3. Protocolo ligero de acceso a directorios (LDAP)

LDAP es un protocolo que nos va a permitir centralizar la autenticación y la autorización de usuarios de nuestros sistemas, en uno o más servidores maestros o esclavos para ofrecer tolerancia a fallos.

Algunas de sus funciones más importantes son la autenticación de usuarios en:

- aplicaciones web
- sistemas operativos
- encaminamiento y entrega de correo electrónico
- servicio FTP
- administración de zonas en Servidores DNS Bind9

3. Protocolo ligero de acceso a directorios (LDAP)

3.1. LDAP

LDAP surge con la idea de tener un directorio para que los administradores de sistemas puedan tener esta información de forma centralizada.

El directorio tiene información de objetos jerarquizada en categorías. Funciona como una base de datos para almacenar los objetos que componen el directorio activo. Estos objetos pueden ser usuarios, grupos, unidades organizativas y recursos (equipos clientes, impresoras, etc)

Es un protocolo para gestionar el acceso de usuarios.

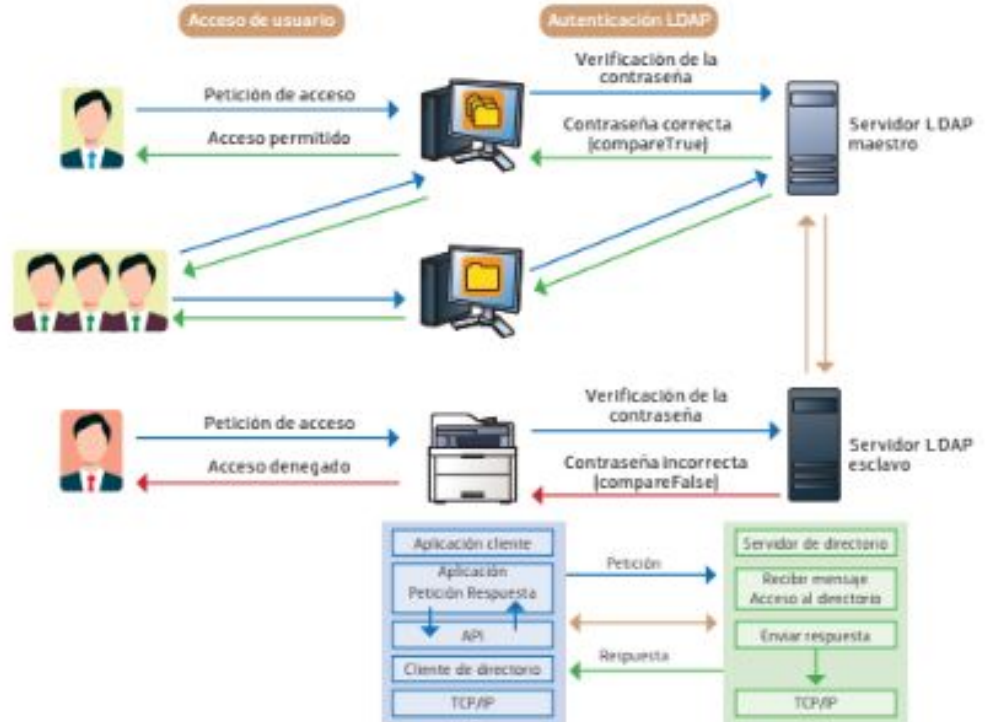
3. Protocolo ligero de acceso a directorios (LDAP)

3.1. LDAP

Funcionamiento

Un servidor LDAP(o varios) contienen los datos que conforman el directorio LDAP o base de datos troncal.

El cliente LDAP se conecta con el servidor LDAP y le hace una consulta. El servidor contesta la respuesta correspondiente.

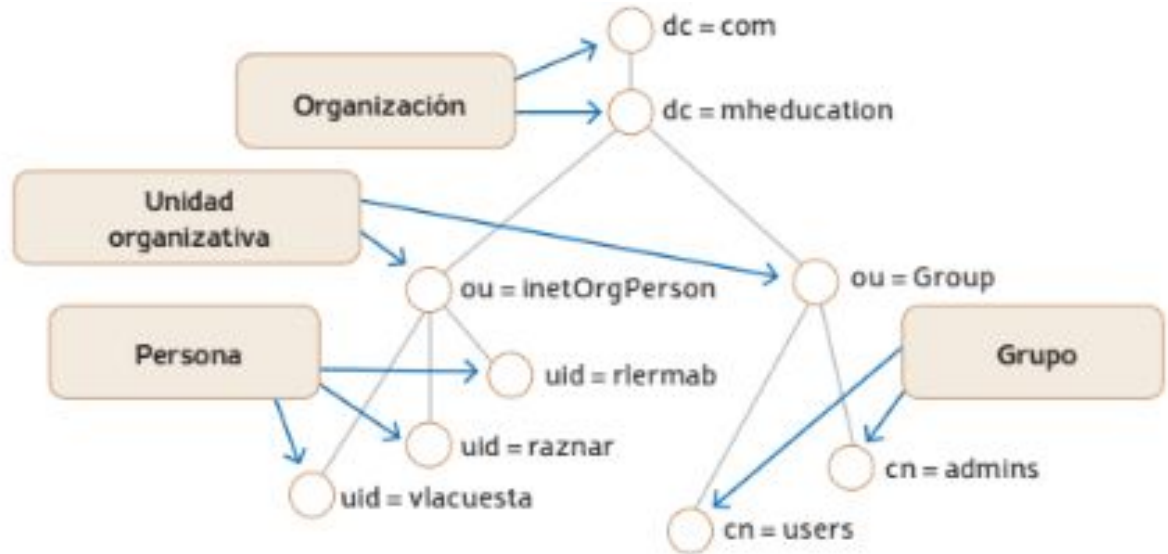


3. Protocolo ligero de acceso a directorios (LDAP)

3.1. LDAP

Esquema de jerarquía

La información está estructurada en árbol jerárquico invertido. Este árbol se denomina árbol de información de directorio.



3. Protocolo ligero de acceso a directorios (LDAP)

3.1. LDAP

Ventajas de un directorio LDAP

- está optimizado para la lectura atómica de registros
- replicar servidores distribuidos es muy sencillo y económico
- es compatible con la mayoría de servicios o aplicaciones a través de su API
- unicidad
- dispone de un modelo de nombres globales que asegura entradas únicas
- usa un sistema jerárquico de almacenamiento
- permite múltiples directorios independientes para diferentes servicios
- forma parte de de TCP/IP y admite seguridad SSL

3. Protocolo ligero de acceso a directorios (LDAP)

3.1. OpenLDAP

Surge como una implementación libre de código abierto de LDAP.

Es la alternativa para implementar estas características en sistemas basados en GNU/Linux.

