INTRODUCCIÓN AL SERVICIO DE DIRECTORIOS

AWS/AMS
Febrero 2021

1. ¿Qué es el Servicio de directorio?

Los directorios son un **tipo específico de bases de datos** con un propósito también específico: almacenar la información sobre un objeto (individuo, recurso de red, documento…).

Su papel es **clave** en cualquier organización que quiera tener la **información** sobre sus empleados, usuarios de red, etc., catalogada y accesible desde multitud de aplicaciones.

Además, el uso de un servicio de directorio facilita la **gestión de la identidad de los usuarios** de los sistemas de información en una organización.

En este núcleo formativo estudiaremos los conceptos básicos de los directorios, su diseño y su implantación.

1. ¿Qué es el Servicio de directorio?

Un **directorio** es una estructura jerárquica que organiza y almacena datos acerca de elementos. Es un tipo concreto de base de datos.

Un **servicio de directorio** es una plataforma que proporciona métodos para gestionar y almacenar los datos que contiene el directorio.

Un servicio de directorio permite la búsqueda de valores a partir de un determinado nombre (o identificador), de forma similar a lo que hace un diccionario.

Aunque el concepto de directorio se relacione con datos, bien cierto es que existen varias diferencias entre un servicio de directorio y una base de datos:

1. ¿Qué es el Servicio de directorio?

- En los directorios se realizan muchas más lecturas de datos que escrituras.
- Los directorios pueden modificar más fácilmente el diseño de las "entidades" que albergan.
- En cambio en una base de datos cambiar el diseño de ésta a posteriori puede ser más complejo.
- Los datos de los directorios suelen estar distribuidos y replicados con ma-yor frecuencia que en bases de datos.
- Los directorios permiten, en general, consultas simples, y no consultas que requieran la fusión de datos provenientes de varias tablas (consultas join de las bases de datos).

1.2 Tipos de Servicios de Directorio

Un tipo sencillo de directorios es el que está incluido en **aplicaciones de software**, como por ejemplo las libretas de direcciones.

Un paso más allá sería que esta aplicación de libreta de direcciones funcionara como una aplicación informática independiente, o quizás fuera un elemento más del sistema operativo. En este caso, sería preciso establecer **un estándar de intercambio de información** para que los demás programas pudieran hacer uso de esta libreta de direcciones.

Un ejemplo de estos sistemas sería **el LDIF** (LDAP, data interchange format) el cual es utilizado como medio habitual de exportación de datos de libreta de direcciones a un fichero de texto imprimible.

1.2 Tipos de Servicios de Directorio

Esta aplicación podría ser una aplicación de red ejecutándose en un servidor.
De este modo, la información de los contactos estaría disponible para todos los equipos clientes que consultaran al servidor.
En este caso, sería necesario establecer un protocolo de comunicaciones a nivel de aplicación para poder realizar distintas operaciones:

- Consultas sobre la información de un contacto.
- Mensaje de error en caso de que no se encontrara el contacto.
- Opcionalmente, un protocolo de identificación del usuario.
- Operaciones de alta, baja y modificación de contactos sólo ejecutables por un usuario con permisos de administrador, etc.

Los **directorios de sistemas operativos en red** almacenan datos de recursos de una red. Algunos ejemplos son **el Active Directory de Microsoft.**

1.2 Tipos de Servicios de Directorio

Ejemplo:

Otro ejemplo de directorio de propósito específico es el sistema de nombres para Internet.

El acceso a servicios basados en Internet se realiza mediante conexiones o envío de datagramas hacia una determinada dirección IP. El sistema DNS resuelve, a partir de un nombre, cuál es la dirección IP del recurso.

Así como DNS se puede ver como un servicio de directorio un tanto específico, los hay de propósito general.

Este es el caso del **servicio LDAP**. Aunque en ciertos casos su uso se remite a tener información sobre los usuarios de una serie de servicios en red (permitiendo, por ejemplo, el acceso a múltiples servicios mediante un único usuario y contraseña), LDAP permite definir soluciones para un amplio espectro de escenarios.

1.3 Espacio de nombres

ESPACIO DE NOMBRES:

En los servicios de directorio, cada objeto está identificado mediante un nombre.

El identificador de objeto debe ser un nombre único dentro del servicio de directorio.

Además de identificar objetos, los identificadores también pueden identificar grupos de objetos, con lo cual se puede diseñar una estructura jerárquica.

Veamos a continuación algunos ejemplos de espacios de nombres:

1.3 Espacio de nombres

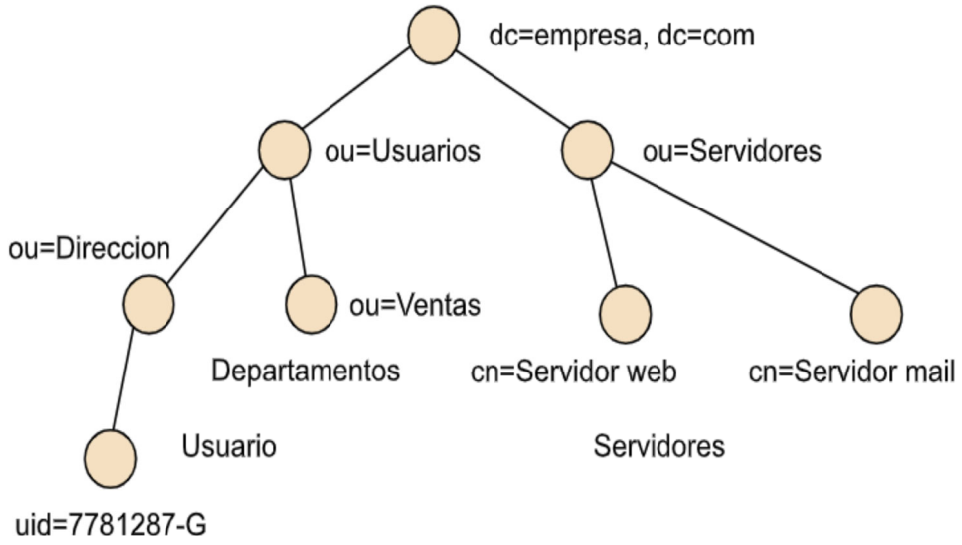
- Sistema DNS: En un sistema DNS, el espacio de nombres permite identificar unívocamente a un equipo conectado a Internet. De hecho, varios nombres pueden apuntar aun mismo equipo. Para identificar un equipo dentro del espacio de nombres, el DNS se ayuda de una estructuración jerárquica iniciada en el dominio '.', que es el dominio raíz.
- Sistema LDAP: El sistema de nombres que usa LDAP permite la organización de los objetos de forma jerárquica.

En la siguiente figura se puede observar como es un ejemplo de directorio LDAP:

Ejemplo de directorio LDAP

En la figura siguiente se puede observar un ejemplo de directorio LDAP.

Ejemplo de espacio de nombres de un directorio basado en LDAP



Espacio de nombres

En esta organización (*empresa.com*) hay definidos dos grupos de objetos: los usuarios y los servidores. Los usuarios se dividen en departamentos. Como veremos más adelante, LDAP y otras implementaciones derivadas permiten la agrupación de elementos y crear una jerarquía. LDAP no fija ninguna jerarquía ni ningún número determinado de niveles: el espacio de nombres permite dar flexibilidad para adaptarse a multitud de usuarios.

Cada objeto se plasma como una **entrada de directorio**. En el ejemplo hay un total de ocho entradas. Cada entrada tiene un nombre *distinguished name* (DN). Por ejemplo, la organización tiene como DN `dc=empresa, dc=com`.

Cada entrada de directorio está compuesta por una serie de **atributos**, cada uno de ellos describe varios aspectos del objeto que la entrada identifica. En el ejemplo se define un usuario. Este objeto podría tener los atributos descritos en la tabla siguiente.

Ejemplo de directorio LDAP

Ejemplo de directorio LDAP

Ejemplo de una entrada LDAP

Atributo	Valor
objectclass	person
cn	José María López García Pepe López García
sn	López García
telephoneNumber	1789
mail	josem.lopez@empresa.com
jpegPhoto	nU6KNyVIYS817zVdf5YKF1FrNb...

Ejemplo directorio LDAP

La definición de qué atributos forman parte del directorio se conoce como el esquema del directorio. A continuación se explica el significado de los anteriores atributos:

- objectclass. Especifica a qué clase pertenece el objeto.
- Common name (cn). Nombre del usuario, puede tener más de un valor. En este caso, se considera nombre del usuario tanto José María, como Pepe.
- Surname (sn). Es el apellido del usuario.
- telephoneNumber. Como su nombre indica, sirve para almacenar el número de teléfono del usuario.
- mail. Almacena la dirección de correo electrónico.
- jpegPhoto. Contiene una pequeña imagen del usuario.

> En LDAP cada objeto se identifica mediante un nombre, el ***distinguished name*** (DN), formado por varios atributos y sus valores.

1.4 Operaciones de cliente

En este subapartado definimos las operaciones más frecuentes a nivel de interacción de un cliente con un servicio de directorio:

- Operaciones de interrogación:

  La operación de **búsqueda** (*search*) permite buscar en el directorio y obtener información de las entradas.

  La herramienta puede ser mediante la línea de comandos, como **ldapsearch.**

```
ldapsearch –h ldap.ejemplo.com –s sub –b "ou=ingenieros" "(cn~=Juan Prados)"
```

En este caso se busca en el servidor ldap.ejemplo.com, dentro del apartado de ingenieros, la entrada correspondiente a un tal Juan Prados. La herramienta ldapsearch ha realizado una consulta al servicio de directorio que no ha precisado de una conexión autenticada o, lo que es lo mismo, ha usado una **conexión anónima**.

1.4 Operaciones de cliente

- Operaciones de actualización: LDAP dispone de cuatro operaciones de actualización de datos: añadir, borrar, modificar y renombrar/mover.
- Operaciones de autenticación: La conexión a un directorio no está exenta de las implicaciones en la seguridad del propio servicio de directorio.

Por ejemplo, puede ser que la consulta de información sea pública para cualquier usuario, mientras que la modificación sea sólo posible para ciertos usuarios con rol de administradores del servicio de directorio.
Para realizar una conexión (operación bind), se especifica el DN de quien realiza la conexión. Se puede usar una contraseña para autenticación, así como distintos métodos de seguridad.

1.4 Operaciones de cliente

- Acceso desde otras aplicaciones :Otra forma en la que se encuentra disponible LDAP es como interfaz pa-ra implementar programas (API). De este modo, por ejemplo, en lenguajeC es posible usar funciones contra un servicio de directorio LDAP, como ldap_search(), ldap_bind(), ldap_add(), etc.

2. Diseño del directorio

El espacio de nombres es un elemento estrechamente ligado con la organización de los objetos que incluye el directorio. En LDAP los objetos se identifican mediante una serie de valores específicos de atributos, en principio uno para cada nivel del espacio de nombres.
Será necesario tener claro qué elementos conviene almacenar para cada entrada, es decir, **cuáles serán los atributos que definirán los objetos del directorio.**

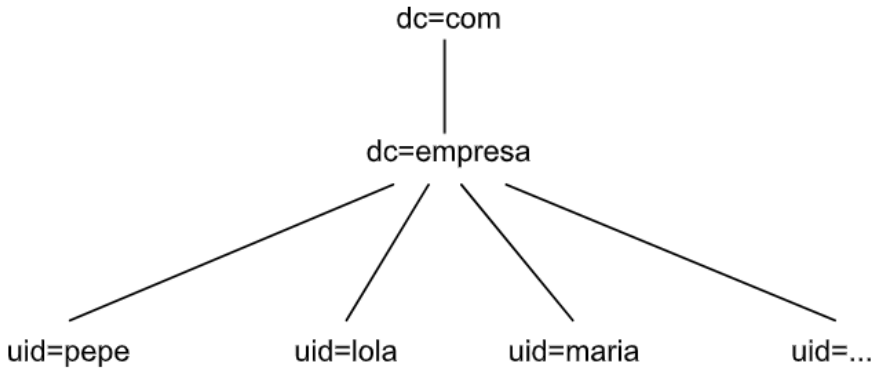
- **Elección del sufijo:** lo importante es que el sufijo (el nombre de la parte "raíz" del árbol del directorio) identifique unívocamente la organización. Se podría escoger la recomendación de la RFC 2247. En ella se especifica que es conveniente mapear en DN del directorio con el nombre DNS que la organización tenga asignado.

2. Diseño del directorio

- **Estructura del directorio plano:**

El espacio de nombres más simple sería un espacio de nombres plano, por ejemplo, sin departamentos ni grupos de usuarios.

La figura siguiente muestra un espacio de nombres plano.



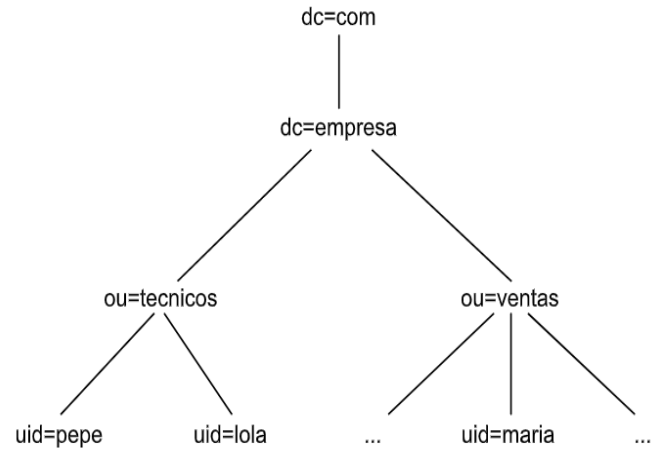
2. Diseño del directorio

- **Estructura del directorio jerarquizado:**

En organizaciones que contemplan departamentos o distintos perfiles de usuarios, se aconseja crear unidades organizacionales (identificadas con el atributo"ou").

Además, distribuir los recursos en grupos puede ser útil, por ejemplo, para temas de control de acceso a recursos.

Ejemplo de estructura jerárquica



2. Diseño del directorio

- **Identificación de objetos en el directorio:**

El último elemento que analizaremos sobre el diseño del espacio de nombres tiene que ver con la identificación de los objetos dentro del directorio.

Tal y como hemos visto, las restricciones que impone LDAP a la hora de identificar objetos son dos:

- El RDN de una entrada se formará a partir de un atributo (o más de uno, aunque no es recomendable).
- En RDN debe ser único entre las entradas "hermanas" (aquellas que penden del mismo padre en la estructura).

Aunque solamente haya establecidas estas dos restricciones, se aconseja engeneral que la identificación de los objetos (en especial cuando se trata depersonas o usuarios) se haga a través **de un identificador único**.

2. Diseño del directorio

- **Esquema del directorio (schema):**

- ✓ Hasta ahora hemos hablado de objetos en el sentido amplio de la palabra. Hemos visto que un directorio contiene entradas con sus distintos atributos.
- ✓ Una entrada representa a un objeto cuya información es almacenada en el directorio. Y se puede entrever que las entradas pueden ser de varios tipos: individuos, recursos, grupos, etc.

> El **esquema del directorio** es la definición de qué tipos de objeto guarda un directorio y qué atributos se utilizan para su definición.

- ✓ Las implementaciones de servicios de directorio ya suelen incluir sus propias definiciones de esquema.
- ✓ En el decurso de este subapartado vamos a tratar los conceptos de **atributo y clase**, esenciales en la definición del esquema del directorio.

2. Diseño del directorio

- **Atributos**:

Los atributos sirven directamente para guardar información (nombre de persona, número de teléfono, fotografía, etc.).

Para definir un atributo en LDAP, es preciso contar con una serie de información:

- Un nombre que identifica al atributo que se define. En caso de LDAP, ya hay algunos nombres estándar definidos (common name, telephoneNumber,etc.), algunos de ellos con una abreviatura también conocida (por ejemplo, "cn" para common name). LDAP no distingue entre mayúsculas y minúsculas.

- Un OID (identificador de objeto) que también identifique al atributo. Los OID son cadenas de números que permiten localizar de forma precisa un objeto de datos. Los OID también definen un espacio de nombres con una jerarquía.

.

Descripción de algunos de los atributos más habituales en LDAP

Atributo	Descripción
cn, commonName	Nombre del objeto. Si el objeto es una persona, sirve para especificar su nombre completo
sn, surname	Apellido de una persona
serialNumber	Número de serie de un dispositivo o recurso
c, countryName	Nombre del país usando dos caracteres, tal y como especifica la ISO 3166
st, stateOrProvinceName	Nombre del estado, provincia, comunidad autónoma, etc.
street, streetAddress	Dirección postal (calle, número, etc.)
o, organizationName	Nombre de la organización
ou, organizationalUnitName	Nombre del departamento u otra unidad organizacional

title	Título de la persona dentro de una organización (presidente, director, etc.)
description	Descripción del objeto, de forma comprensible para los humanos
postalCode	Código postal
telephoneNumber	Número de teléfono
preferredDeliveryMethod	Descripción de cómo quiere una persona que se le entregue información (por ejemplo, por fax o e-mail)
member	Se trata de un DN que indica de quién es miembro el objeto en el árbol del directorio
uid, userid	Identificador de usuario, en general usado para autenticarse en un servicio o sistema

2. Diseño del directorio

- **Clase de objeto:**

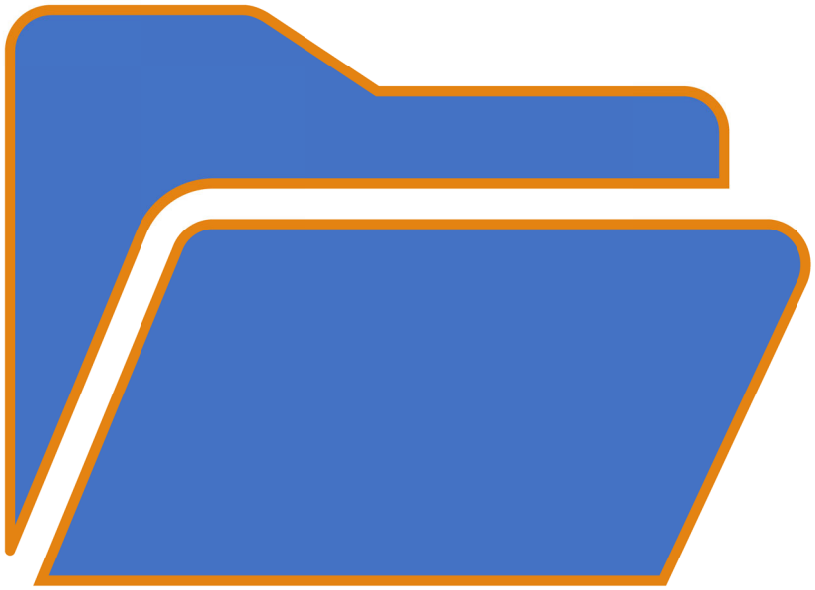
Una vez los atributos han sido definidos, se pueden utilizar clases de objetos para definir cómo son las entradas del directorio.

Una clase de objeto es un atributo de una entrada. Especifica los atributos que puede tener la entrada.

Un identificador de clase de objeto es un valor único que identifica una clase de objeto.

Para definir una clase de objeto, se debe especificar su identificador de objeto, un nombre, un nombre alternativo, una subclase, un tipo de clase de objeto, una lista de atributos debe contener y una lista de atributos puede contener o conjuntos de atributos, y una descripción.

.



3. Implementaciones de servicio de directorio

3. Implementacion de servicio de directorio

Después de realizar una aproximación teórica al concepto, diseño y usos de los servicios de directorio, y de haber estudiado el sistema LDAP, vamos a estudiar dos casos concretos de implementación:

- **El Active Directory**: Si Windows NT usaba el NetBIOS como mecanismo primario de comunicación de red (y el WINS como base de datos de nombre de objetos de red), el Active Directory requiere el uso de TCP/IP, así como del servicio DNS.
- **El OpenLDAP:** OpenLDAP es una implementación de un servicio de directorio basado en LDAP bajo la filosofía de software libre y código abierto. El proyecto OpenLDAP es quien se encarga de su desarrollo y sus productos se hallan disponibles en multitud de distribuciones GNU/Linux**.**

3. Implementacion de servicio de directorio

- **El OpenLDAP:** El servidor slapd ejecuta las funciones de servicio de directorio. Se encarga de interaccionar con el backend (almacén de datos) correspondiente.

OpenLDAP permite la replicación de contenidos. Actualmente, se utilizan los términos de proveedor y consumidor de actualizaciones. Esto permite definir mejores reglas de actualización, haciendo posible que un servidor pueda actuar de proveedor o bien de consumidor en función de la necesidad.

El motor de sincronización para OpenLDAP es **syncrepl,** que usa como protocolo el **LDAPSync**.

4. Conceptos relacionados

Una vez que disponemos de una idea global del concepto de directorio, es conveniente que hagamos un repaso de la terminología que vamos a emplear cuando hablemos de Servicios de directorio:

- **Dominio**: Un Dominio es una **colección de objetos** dentro del directorio que forman un subconjunto administrativo. Pueden existir diferentes dominios dentro de un **bosque**, cada uno de ellos con su propia colección de objetos y unidades organizativas.
- **Objeto:** La palabra Objeto se utiliza como nombre genérico para referirnos a cualquiera de los componentes que forman parte del directorio, como una impresora o una carpeta compartida, pero también un usuario, un grupo, etc.

Nota: Como veremos más adelante, las características específicas de cada tipo de objeto quedarán definidas en el Esquema de la base de datos

4. Conceptos relacionados

En general, los objetos se organizan en tres categorías:

- Usuarios: identificados a través de un nombre (y, casi siempre, una contraseña), que pueden organizarse en grupos, para simplificar la administración.
- Recursos: que son los diferentes elementos a los que pueden acceder, o no, los usuarios según sus privilegios. Por ejemplo, carpetas compartidas, impresoras, etc.
- Servicios: que son las diferentes funciones a las que los usuarios pueden tener acceso. Por ejemplo, el correo electrónico.

IMPORTANTE

Definición de usuario:

Desde un punto de vista informático, un usuario es un conjunto de permisos y de privilegios sobre determinados recursos.

En este sentido, un usuario no tiene que ser, necesariamente, una persona.

4. Conceptos relacionados

Controlador de dominio: Un Controlador de dominio (domain controller) contiene la **base de datos de objetos del directorio para un determinado dominio**, incluida la información relativa a la seguridad. Además, será responsable de la autenticación de objetos dentro de su ámbito de control.

En un dominio dado, puede haber varios controladores de dominio asociados, de modo que cada uno de ellos represente un rol diferente dentro del directorio. Sin embargo, a todos los efectos, todos los controladores de dominio, dentro del mismo dominio, tendrán la misma importancia.

RECUERDA

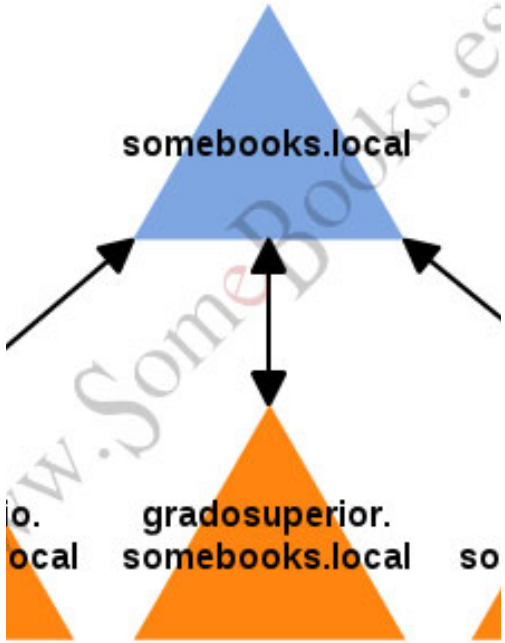
Cuando instalamos Active Directory en un ordenador con Windows Server 2008, convertimos a ese ordenador en un Controlador de dominio.

4. Conceptos relacionados

Árboles:

Un Árbol es simplemente una **colección de dominios** que dependen de una raíz común y se encuentra organizados como una determinada jerarquía.

De esta forma, sabremos que los dominios somebooks.es e informatica.somebooks.es forman parte del mismo árbol, mientras que sliceoflinux.com y somebooks.es no.

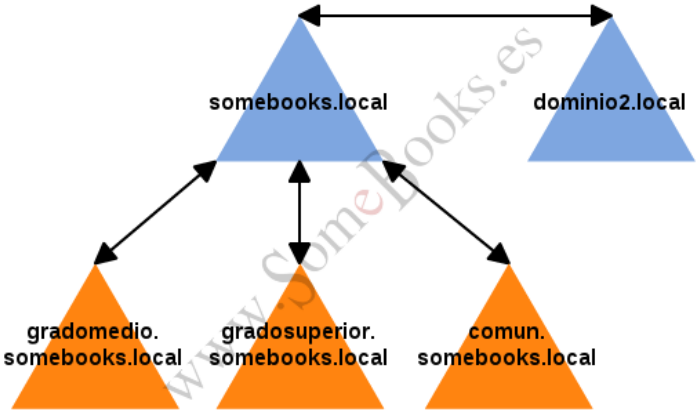


El objetivo de crear este tipo de estructura es fragmentar los datos del Directorio Activo, replicando sólo las partes necesarias y ahorrando ancho de banda en la red.

4. Conceptos relacionados

Bosque:

Un bosque es una agrupación de varios dominios Active Directory y posee un único esquema. El primer dominio instalado en un bosque se llama dominio raíz del bosque. Varios árboles de dominio cuyo espacio de nombres no es continuo representan un bosque.



Glosario

alias *m* Entrada que en realidad enlaza con otra entrada del directorio.

atributo *m* Parte de la entrada destinada a guardar una pieza de información, como por ejemplo un apellido o un número de teléfono.

base *f* En la operación de búsqueda especifica, desde qué entrada u objeto se quiere empezar a buscar información.

búsqueda *f* Operación básica de interrogación al servicio del directorio, mediante la cual se obtiene información conforme una serie de criterios.

DAP (*directory access protocol*) *m* Sistema primitivo de implementación y gestión de directorios, basado en la especificación X.500.

directorio *m* Tipo especializado de base de datos jerárquica que organiza y almacena datos acerca de elementos (entradas de directorio).

***distinguished name* (DN)** *m* Relación de DN relativos que identifican unívocamente una entrada en un directorio LDAP.

DN relativo *m* Atributo y su valor que identifican a una entrada para un nivel en concreto del árbol del directorio.

esquema *m* Definición de qué tipos de objeto guarda un directorio y los atributos usados en la definición de los objetos.

Glosario

esquema *m* Definición de qué tipos de objeto guarda un directorio y los atributos usados en la definición de los objetos.

LDAP (*lightweight DAP*) *f* Interfaz entre clientes de directorio y sistemas DAP, que luego evolucionó hacia un servicio de directorio.

objectclass *m* Atributo de una entrada de directorio basado en LDAP que especifica a qué clase pertenece la entrada (por ejemplo, "person").

servicio de directorio *m* Plataforma que proporciona métodos para gestionar y almacenar los datos que contiene el directorio.

sufijo *m* DN para la raíz del árbol de directorio.

user identifier *m* Atributo específico para guardar el "login" de un usuario de un sistema informático.