

A.S.I.R. 2019/2020

OpenLDAP y Monitorización Zabbix



ZABBIX

Alejandro Santoro Recio
Adrián Juárez Villarreal

ÍNDICE

1	Introducción	3
1.1	Sistemas y Software para implementar.....	4
1.2	Estructura de la red.....	4
2	OpenLDAP.....	5
2.1	Introducción.....	5
2.2	Instalación.....	5
2.2.1	Instalación en el servidor.....	5
2.2.2	Creación de estructura mediante archivos de configuración.....	10
2.2.3	Instalación Cliente Ubuntu.....	13
2.2.4	Instalación Cliente Windows.....	16
2.3	Instalación PhpLdapAdmin.....	20
2.4	SSL OpenLDAP.....	23
2.5	Estructura del LDAP.....	27
3	SAMBA	28
3.1	Introducción.....	28
3.2	Instalación.....	28
3.3	Configuración.....	28
3.4	Creación usuarios y grupos Samba.....	30
3.5	Crear carpetas compartidas usando NFS.....	30
3.5.1	¿Qué es NFS?.....	30
3.5.2	Instalación.....	31
3.5.3	Crear y compartir una carpeta con NFS.....	31

4	SSH.....	32
4.1	Introducción.....	32
4.2	Instalación.....	32
5	Scripts para automatización de tareas	33
5.1	Estructura.....	34
5.2	Scripts de automatización de tareas.....	34
6	Zabbix.....	38
6.1	Introducción.....	38
6.2	Instalación.....	38
6.2.1	Zabbix Server (en el servidor).....	38
6.2.2	Zabbix Agent en el cliente.....	43
6.3	Añadir clientes (Zabbix Agent / hosts) al servidor Zabbix.....	44
6.4	Monitorizar servicios a través de plantillas.....	46
6.4.1	Servicio SSH.....	46
6.4.2	Servicio Apache.....	47
6.4.3	Servicio LDAP.....	47
6.5	Ver el almacenamiento de los equipos.....	47
7	CONCLUSIÓN.....	50
8	BIBLIOGRAFÍA.....	51

1 Introducción

Este proyecto consiste en una simulación de la estructura de una red informática de un instituto basado en un dominio OpenLDAP. Estará constituida por diferentes grupos y usuarios, los cuales estarán organizados por profesores y alumnos, los cuales tendrán diferentes privilegios dentro de nuestra red. Esta estructura permitirá administrar los equipos del dominio desde el servidor tanto por interfaz gráfica como por línea de comandos.

Esta idea surge de que podamos implementar, en todos los equipos de la red, perfiles móviles para que todos los usuarios puedan acceder desde cualquiera de los equipos disponibles. Además, nos permitirá tener organizada dicha estructura de manera centralizada para facilitar tareas de administración.

Para la optimización de nuestro dominio, hemos implementado Samba, el cual es un servicio que nos permite compartir recursos entre los distintos usuarios que pertenecen a nuestra red.

Para supervisar el buen funcionamiento de la red, hemos implementado la herramienta Zabbix, la cual nos permite hacer un reporte en tiempo real a través de gráficas, datos y alertas visuales que muestran el estado y rendimiento de los servicios y equipos monitoreados.

Con esta herramienta también podemos monitorear el estado de los equipos, información acerca de ellos como, por ejemplo, la memoria utilizada, y los diferentes servicios que pueden estar implementados en el servidor como en los demás equipos.

INTRODUCTION

This project consists of a simulation of the structure of a computer network of an high school based on an OpenLDAP domain. It will be constituted by different groups and users, which will be organized by teachers and students, who will have different privileges within our network. This structure will allow the administration of the domain equipment from the server both by graphic interface and by command line.

This idea arises from the fact that we can implement, in all the computers of the network, mobile profiles so that all the users can access from any of the available computers. In addition, it will allow us to have this structure organized in a centralized way to facilitate administration tasks.

For the optimization of our domain, we have implemented Samba, which is a service that allows us to share resources between different users who belong to our network.

To monitor the proper functioning of the network, we have implemented the Zabbix tool, which allows us to make a real-time report through graphics, data and visual alerts that show the status and performance of services and equipment monitored.

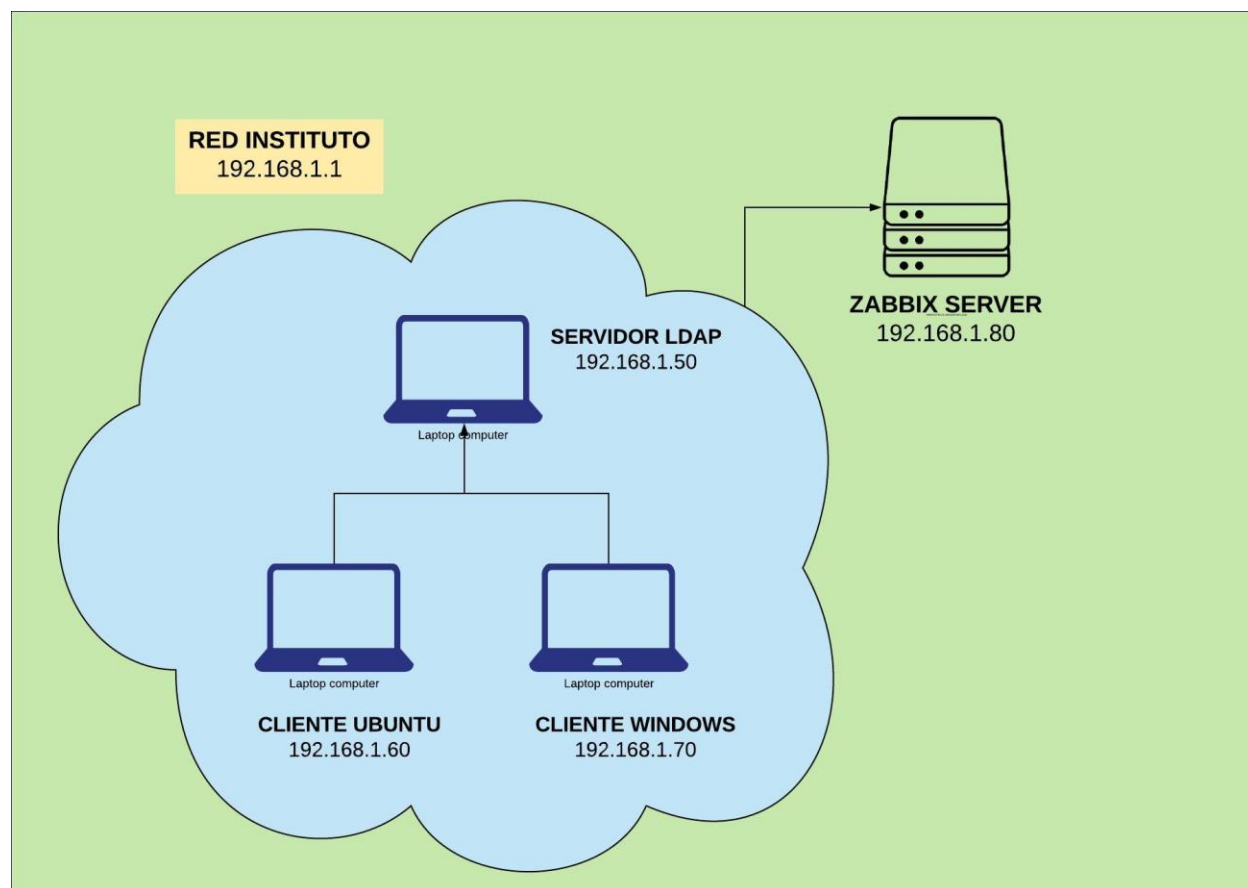
With this tool we can also monitor the state of the equipments, information about them such as the memory used, and the different services that can be implemented in the server as in the other equipment.

1.1 Sistemas y Software para implementar

Para la implementación de nuestro dominio OpenLDAP, hemos utilizado:

- 1 Ubuntu 18.04 Desktop, que actuará como servidor.
- 1 Ubuntu 18.04 Desktop, que actuará como cliente.
- 1 Windows 10, que actuará también como cliente.
- 1 Ubuntu 18.04 Desktop que desempeñará el papel de Servidor de Zabbix.
- Herramienta Zabbix, en la versión 4.4
- Servicio Apache.
- Servicio SSH.
- Servicio MySQL.
- Servicio Samba
- Servicio NFS.

1.2 Estructura de la red



La estructura de nuestra red se constituye de:

- Servidor LDAP → IP: 192.168.1.50
- Cliente Ubuntu → IP: 192.168.1.60
- Cliente Windows → IP: 192.168.1.70
- Servidor Zabbix → IP: 192.168.1.80

2 OpenLDAP

2.1 Introducción

OpenLDAP es una implementación libre y de código abierto del protocolo LDAP que se inició en el año 1998. Es un conjunto de protocolos abiertos usados para acceder a la información guardada centralmente a través de la red. Se trata de un sistema cliente/servidor en el cual el servidor puede usar una variedad de bases de datos para guardar un directorio, cada uno optimizado para operaciones de lectura rápidas y en gran volumen.

¿Por qué usar OpenLDAP?

La mayor ventaja de OpenLDAP es que se puede consolidar información para toda una organización dentro de un repositorio central, es decir, en vez de administrar listas de usuarios para cada grupo dentro de una organización, puede usar este sistema como directorio central, accesible desde cualquier parte de la red. Otra ventaja que dispone es que soporta la Capa de conexión segura (SSL) y la Seguridad de la capa de transporte (TLS), por lo que aumenta la seguridad de nuestro sistema.

OpenLDAP tiene tres componentes principales:

- slapd -Demonio del servidor y herramientas.
- Bibliotecas que implementan el protocolo LDAP.
- Programas para cliente: ldapsearch, ldapadd, ldapdelete, entre otros.

Sabiendo ya un poco en qué consiste OpenLDAP, vamos a proceder a la instalación.

2.2 Instalación

2.2.1 Instalación en el servidor

Antes de instalarlo tendremos que configurar la red de nuestra máquina de la siguiente manera:

```
GNU nano 2.9.3 /etc/netplan/01-network-manager-all.yaml

# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      addresses: [192.168.1.50/24]
      gateway4: 192.168.1.1
      nameservers:
        search: [mytcip.local]
        addresses: [192.168.1.1]
```

Lo primero que haremos será instalar el paquete slapd, que es donde se encuentra el servidor OpenLDAP, y el paquete que contiene las utilidades de administración de LDAP ldap-utils.

```
root@servidor:/home/servidor# apt-get install libnss-ldap -y
```

Nos aparecerá una ventana donde iremos insertando los datos de nuestro servidor.

- Introduciremos la ip de nuestro servidor.

Configuración de ldap-auth-config

Please enter the URI of the LDAP server to use. This is a string in the form of ldap://<hostname or IP>:<port>/. ldaps:// or ldapi:// can also be used. The port number is optional.

Note: It is usually a good idea to use an IP address because it reduces risks of failure in the event name service problems.

LDAP server Uniform Resource Identifier:

ldapi:///192.168.1.50

<Aceptar>

- Introduciremos la base de nuestra estructura de OpenLDAP.

Distinguished name of the search base:

dc=servidor,dc=local

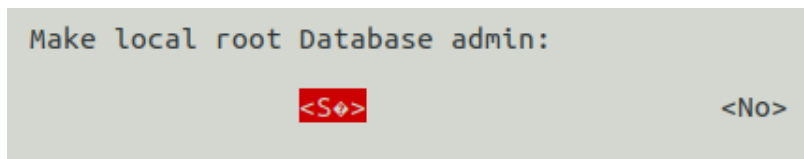
- Elegimos la versión 3.

LDAP version to use:

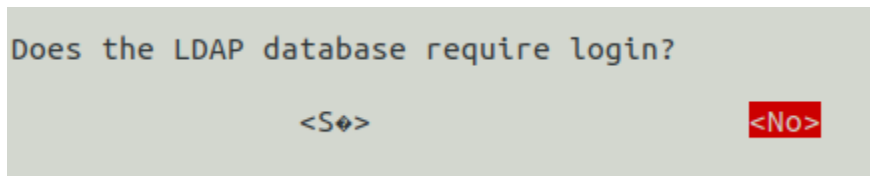
3

2

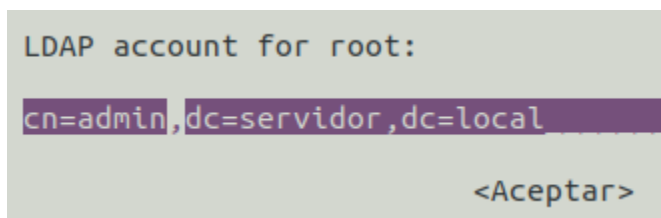
- Elegimos la opción “Sí”, la cual hará que se cree un administrador local de la base de datos.



- Le damos a que no va a necesitar login para el acceso a la base de datos.



- Introducimos la ruta del usuario root del OpenLDAP.



Ante cualquier problema que pueda surgir se podrá reconfigurar **ldap-auth-config** con el comando **dpkg-reconfigure ldap-auth-config**.

Configurar la autenticación para los clientes.

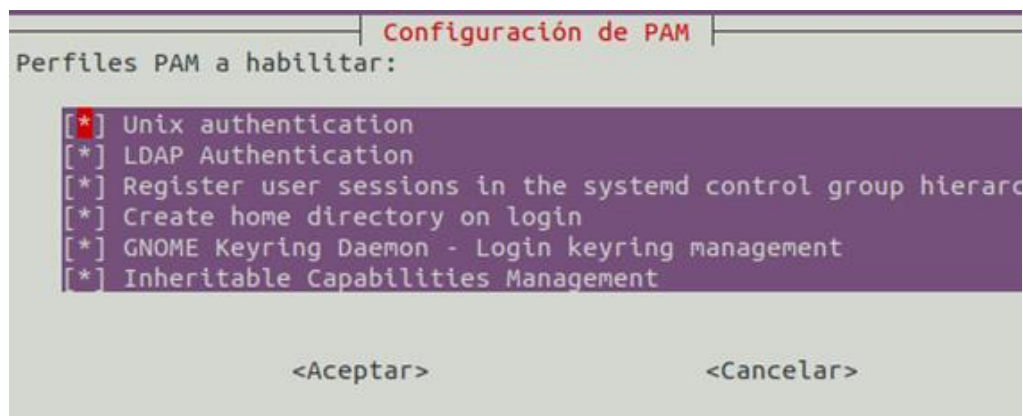
```
$ auth-client-config -t nss -p lac_ldap
```

Este comando ejecutará un script que nos ayudará a modificar los archivos de configuración de PAM y NSS.

Una vez ejecutado el anterior comando, actualizaremos la configuración de política de autenticación de PAM. PAM establece una interfaz entre los programas de usuario y distintos métodos de autenticación. De esta forma, el método de autenticación se hace transparente para los programas.

```
$ pam-auth-update
```

Cuando ejecutemos este comando, nos aparecerá otra ventana emergente donde nos aparecerán todos los módulos que podremos habilitar:



Una vez acabada la configuración automática, podremos editar el archivo de configuración de ldap (/etc/ldap.conf).

Nos aseguraremos de configurar las siguientes líneas de esta manera:

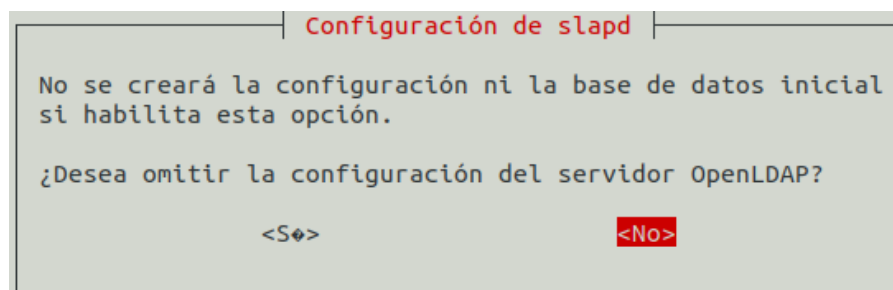
```
host 192.168.1.50
base dc=servidor,dc=local
uri ldapi:///192.168.1.50
ldap_version 3
rootbinddn cn=admin,dc=servidor,dc=local
bind_policy soft
```

A continuación, configuraremos el demonio **slapd**.

```
$ dpkg-reconfigure slapd
```

Nos aparecerá un asistente para evitar que tengamos que configurar a mano el archivo **slapd.conf**.

- Elegimos la opción **No** para empezar a configurar todo desde el principio:



- Introducimos el nombre de nuestro dominio.

Configuración de slapd

El nombre de dominio DNS se utiliza para construir el DN base del directorio LDAP. Por ejemplo, si introduce «foo.example.org» el directorio se creará con un DN base de «dc=foo, dc=example, dc=org».

Introduzca el nombre de dominio DNS:

servidor.local

- Seleccionamos el motor de base de datos que utilizaremos:

Configuración de slapd

Motor de base de datos a utilizar:

BDB

HDB

MDB

<Aceptar>

- En este punto, elegimos que no borre la base de datos cuando se borre el paquete slapd.

Configuración de slapd

¿Desea que se borre la base de datos cuando se purgue el paquete slapd?

<Si> **<No>**

- Y con esto elegimos que sobrescriba los datos anteriores para utilizar la base de datos creada recientemente.

Configuración de slapd

Existen ficheros en «/var/lib/ldap» que probablemente interrumpen el proceso de configuración. Si activa opción, se moverán los ficheros de las bases de datos antiguas antes de crear una nueva base de datos.

¿Desea mover la base de datos antigua?

<Si> <No>

2.2.2 Creación de estructura mediante archivos de configuración

La estructura de nuestro OpenLDAP se puede crear de dos maneras diferentes: por archivos de configuración y mediante la interfaz del PhpldapAdmin. En este caso vamos a utilizar la opción de los archivos de configuración.

Para comenzar, lo primero que tenemos que hacer será crear, en la ubicación que nosotros queramos, el archivo **base.ldif**, donde organizaremos nuestra estructura.

```
root@servidor:~# nano ~/base.ldif
```

```
GNU nano 2.9.3 base.ldif

dn: ou=Grupos,dc=servidor,dc=local
objectClass: organizationalUnit
ou: Grupos

dn: ou=Usuarios,dc=servidor,dc=local
objectClass: organizationalUnit
ou: Usuarios

dn: ou=Alumnos,ou=Usuarios,dc=servidor,dc=local
objectClass: organizationalUnit
ou: Alumnos

dn: ou=Profesores,ou=Usuarios,dc=servidor,dc=local
objectClass: organizationalUnit
ou: Profesores
```

Lo siguiente que tenemos que hacer será añadir esta configuración al sistema. Para eso tendremos que ejecutar el siguiente comando:

```
root@servidor:~# ldapadd -x -D cn=admin,dc=Servidor,dc=local -W -f base.ldif
Enter LDAP Password:
adding new entry "ou=Grupos,dc=servidor,dc=local"

adding new entry "ou=Usuarios,dc=servidor,dc=local"

adding new entry "ou=Alumnos,ou=Usuarios,dc=servidor,dc=local"

adding new entry "ou=Profesores,ou=Usuarios,dc=servidor,dc=local"
```

El comando anterior tiene distintas opciones:

- Opción -W es para que te pida la contraseña.
- Opción -f para el archivo que quieras que lea.
- Opción -x

- Opción -D para elegir el DN que tú quieras enlazar.

Con esto ya tenemos la base de nuestro OpenLDAP, ahora lo siguiente que haremos será crear el archivo **grupo.ldif**, donde pondremos todos los grupos que queramos crear dentro de la base de nuestro servidor.

```
GNU nano 2.9.3 grupo.ldif
dn: cn=Alumnos,ou=Grupos,dc=servidor,dc=local
objectClass: posixGroup
objectClass: top
cn: Alumnos
gidNumber: 500

dn: cn=Profesores,ou=Grupos,dc=servidor,dc=local
objectClass: posixGroup
objectClass: top
cn: Profesores
gidNumber: 501
```

Y hacemos lo mismo que con el archivo **base.ldif**

```
root@servidor:~# ldapadd -x -D cn=admin,dc=servidor,dc=local -W -f grupo.ldif
Enter LDAP Password:
adding new entry "cn=Alumnos,ou=Grupos,dc=servidor,dc=local"

adding new entry "cn=Profesores,ou=Grupos,dc=servidor,dc=local"
```

Por último, tendremos que añadir los usuarios de nuestro servidor. Crearemos el archivo **usuario.ldif** y lo editaremos para dejarlo como en la imagen siguiente:

```
GNU nano 2.9.3 usuario.ldif
dn: uid=ajarez,ou=Alumnos,ou=Usuarios,dc=servidor,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
uid: ajarez
sn: Juarez
givenName: Adrian
cn: Adrian Juarez
displayName: Adrian Juarez
uidNumber: 1041
gidNumber: 500
userPassword: servidor
gecos: Adrian Juarez
loginShell: /bin/bash
homeDirectory: /home/alumnado/ajarez
shadowExpire: -1
shadowFlag: 0
shadowWarning: 7
shadowMin: 8
shadowMax: 999999
shadowLastChange: 10877
mail: ajarezcdr@gmail.com
postalCode: 28030
o: servidor
```

Para aplicar los cambios y añadir los usuarios al servidor ejecutaremos lo siguiente:

```
root@servidor:~# ldapadd -x -D cn=admin,dc=servidor,dc=local -W -f usuario.ldif
Enter LDAP Password:
adding new entry "uid=ajarez,ou=Alumnos,ou=Usuarios,dc=servidor,dc=local"
```

Para asegurarnos de que todo ha salido correctamente, podremos ver la información del sistema con el comando **ldapsearch**.

```
server@servidor:~$ sudo ldapsearch -xLLL -b "ou=Grupos,dc=servidor,dc=local"
[sudo] contraseña para server:
dn: ou=Grupos,dc=servidor,dc=local
objectClass: organizationalUnit
objectClass: top
ou: Grupos

dn: cn=Alumnos,ou=Grupos,dc=servidor,dc=local
gidNumber: 500
cn: Alumnos
objectClass: posixGroup
objectClass: top

dn: cn=Profesores,ou=Grupos,dc=servidor,dc=local
gidNumber: 501
cn: Profesores
objectClass: posixGroup
objectClass: top
```

En la imagen anterior nos mostrará todo lo que esté dentro de la ruta que hemos introducido.

Otra opción para ver la información que tenemos es utilizar el comando **slapcat**, que nos muestra el contenido de todo el árbol.

```
server@servidor:~$ sudo slapcat
dn: dc=servidor,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: Instituto
dc: servidor
structuralObjectClass: organization
entryUUID: e145e0ee-11e0-103a-852d-610d163f2803
creatorsName: cn=admin,dc=servidor,dc=local
createTimestamp: 20200413144327Z
entryCSN: 20200413144327.401760Z#000000#000#000000
modifiersName: cn=admin,dc=servidor,dc=local
modifyTimestamp: 20200413144327Z

dn: cn=admin,dc=servidor,dc=local
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9R3NCNGZ2TWRFOWVidDUwd0dLdkhkV3hFR05sUUL0RWs=
structuralObjectClass: organizationalRole
entryUUID: e14cbe46-11e0-103a-852e-610d163f2803
creatorsName: cn=admin,dc=servidor,dc=local
createTimestamp: 20200413144327Z
entryCSN: 20200413144327.446743Z#000000#000#000000
modifiersName: cn=admin,dc=servidor,dc=local
modifyTimestamp: 20200413144327Z
```

2.2.3 Instalación Cliente Ubuntu

De la misma manera que en el servidor, en el cliente también tendremos una configuración de la red:

```
/etc/netplan/01-network-manager-all.yaml

# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      addresses: [192.168.1.60/24]
      gateway4: 192.168.1.1
      nameservers:
        search: [mytcip.local]
        addresses: [192.168.1.50,192.168.1.1]
```

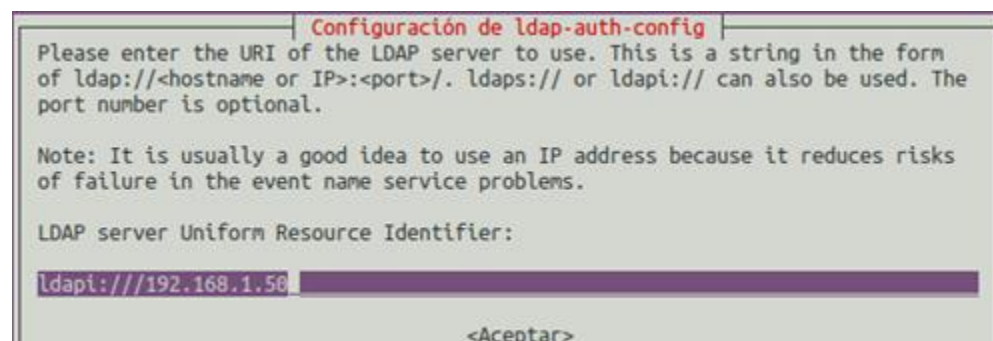
En el cliente necesitamos ajustar el comportamiento de los servicios **NSS** y **PAM**.

Primero instalamos los siguientes paquetes:

```
root@cliente:/home/cliente# sudo apt-get install libpam-ldap libnss-ldap nss-updatedb
libnss-db nscd ldap-utils -y
```

Y como en el servidor, aquí también nos aparecerá una ventana emergente para la configuración:

Introducimos la IP de nuestro servidor para conectarnos a él.



The screenshot shows a terminal window titled "Configuración de ldap-auth-config". It contains the following text:

```
Please enter the URI of the LDAP server to use. This is a string in the form
of ldap://<hostname or IP>:<port>/. ldaps:// or ldapi:// can also be used. The
port number is optional.

Note: It is usually a good idea to use an IP address because it reduces risks
of failure in the event name service problems.

LDAP server Uniform Resource Identifier:
ldapi:///192.168.1.50
<Aceptar>
```

Los demás datos a introducir en el resto de pasos serán idénticos a los del apartado 2.2.1.

Para finalizar la configuración completa, deberemos modificar algunos parámetros en los archivos de configuración del cliente.

Editaremos el archivo **/etc/ldap.conf** y modificaremos las siguientes líneas:

```
bind_policy soft

pam_password crypt

uri ldap://192.168.1.50
```

Ahora editaremos el archivo **/etc/ldap/ldap.conf** para dejarlo de la siguiente manera:

```
GNU nano 2.9.3 /etc/ldap/ldap.conf Modificado
#
# LDAP Defaults
#
# See ldap.conf(5) for details
# This file should be world readable but not world writable.

BASE    dc=servidor,dc=local
URI      ldap://ldap.servidor.local

SIZELIMIT    0
TIMELIMIT    0
DEREF        never

# TLS certificates (needed for GnuTLS)
TLS_CACERT    /etc/ssl/certs/ca-certificates.crt
```

El último archivo a configurar será el **/etc/nsswitch.conf**, que es el fichero de configuración de las Bases de Datos del Sistema.

```
GNU nano 2.9.3 /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the 'glibc-doc-reference' and 'info' packages installed, try:
# `info libc "Name Service Switch"' for information about this file.

passwd:      files ldap
group:       files ldap
shadow:      files ldap

hosts:       files mdns4_minimal [NOTFOUND=return] wins dns mdns4
networks:    files

protocols:   db files
services:    db files
ethers:      db files
rpc:         db files

netgroup:    nis
```

Actualizamos NSS y configuramos PAM para que utilicen LDAP.

```
root@cliente:/home/cliente# nss_updatedb ldap
passwd... done.
group... done.
```

El comando **getent** nos permite obtener entradas de varios archivos de texto del sistema, por ejemplo, de **passwd** y **group**. La ventaja es que consolida la información local con la obtenida a través de la red.

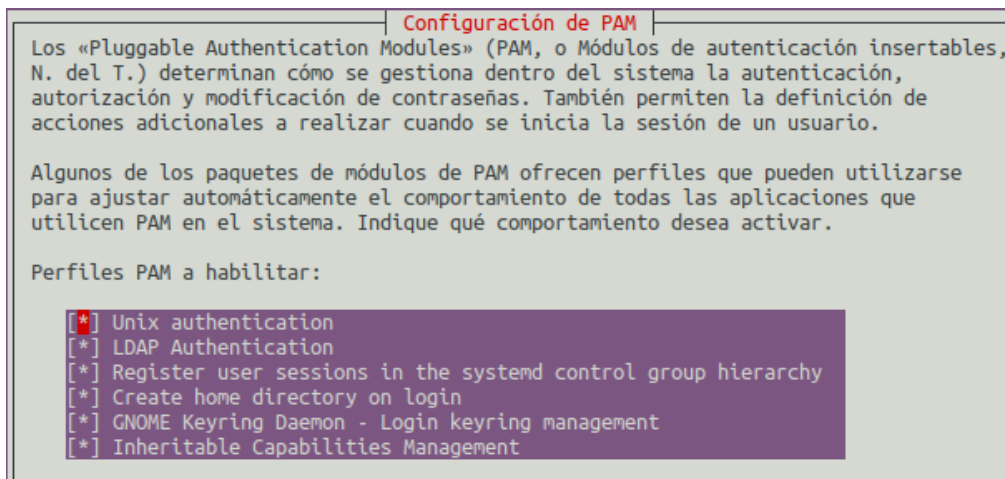

```
root@cliente:/home/cliente# getent passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
```

En la parte final del listado mostrado por dicho comando nos fijaremos en la parte final donde aparecen los usuarios creados por nosotros en el ldap (asantoro,ajuarez,rbautista,amora).

```
administrador:x:1001:1001::/home/administrador:/bin/sh
pepito:x:1002:1002::/home/pepito:/bin/sh
zabbix:x:127:131::/var/lib/zabbix:/usr/sbin/nologin
asantoro:*:1040:500:Alejandro Santoro:/home/alumnado/asantoro:/bin/bash
ajuarez:*:1041:500:Adrian Juarez:/home/alumnado/ajuarez:/bin/bash
rbautista:*:1020:501:ruben bautista:/home/profesorado/rbautista:/bin/bash
amora:*:1021:501:Amadeo Mora:/home/profesorado/amora:/bin/bash
```

Actualizamos la configuración de las políticas de autenticación predeterminadas de PAM.

```
root@cliente:/home/cliente# pam-auth-update
```



Hecho esto, el cliente ya estaría listo para autenticarnos con una cuenta creada a través del servidor LDAP. Sin embargo, las carpetas home de cada usuario no son creadas automáticamente. No obstante, si queremos que dicha carpeta se cree automáticamente al iniciar sesión con un nuevo usuario deberíamos añadir la siguiente directiva en el archivo **/etc/pam.d/common-session**.

```
session required pam_mkhomedir.so skel=/etc/skel/ umask=0022
```

Por último editaremos el archivo **/etc/pam.d/common-password** y buscamos la siguiente línea:

```
password [success=1 user_unknown=ignore default=die] pam_ldap.so use_authtok try_firs$
```

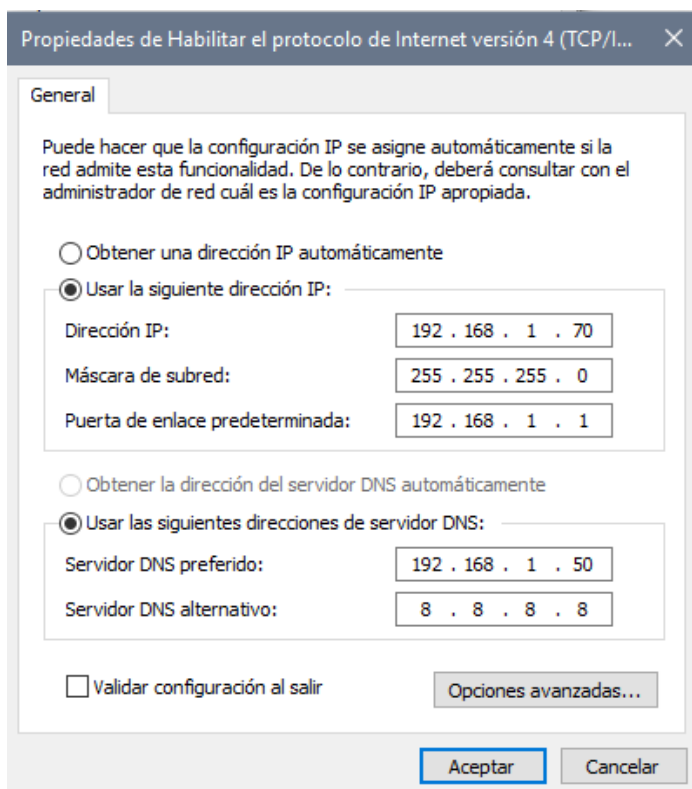

Y la sustituimos por la siguiente:

```
password [success=1 user_unknown=ignore default=die] pam_ldap.so
```

Con esto último, ya estaría todo configurado en el cliente Ubuntu.

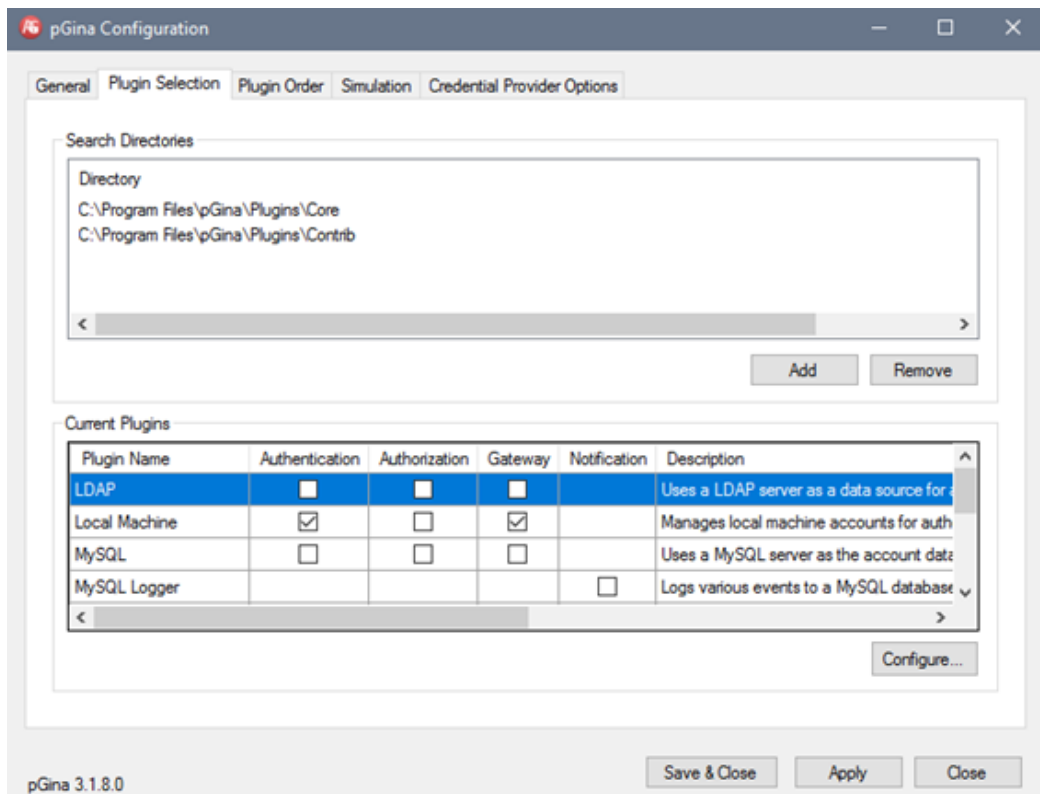
2.2.4 Instalación Cliente Windows

En esta máquina tendremos la siguiente configuración de red:

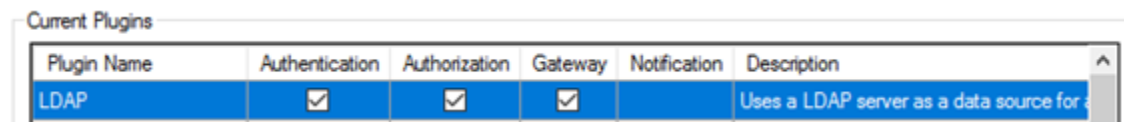


Para la instalación, lo primero que tendremos que hacer será instalar **pGina** (Versión Estable), que en este caso será la versión 3.1.8.0.

Una vez lo tengamos descargado e instalado, lo abrimos y nos vamos a la ventana de **Plugin Selection**.

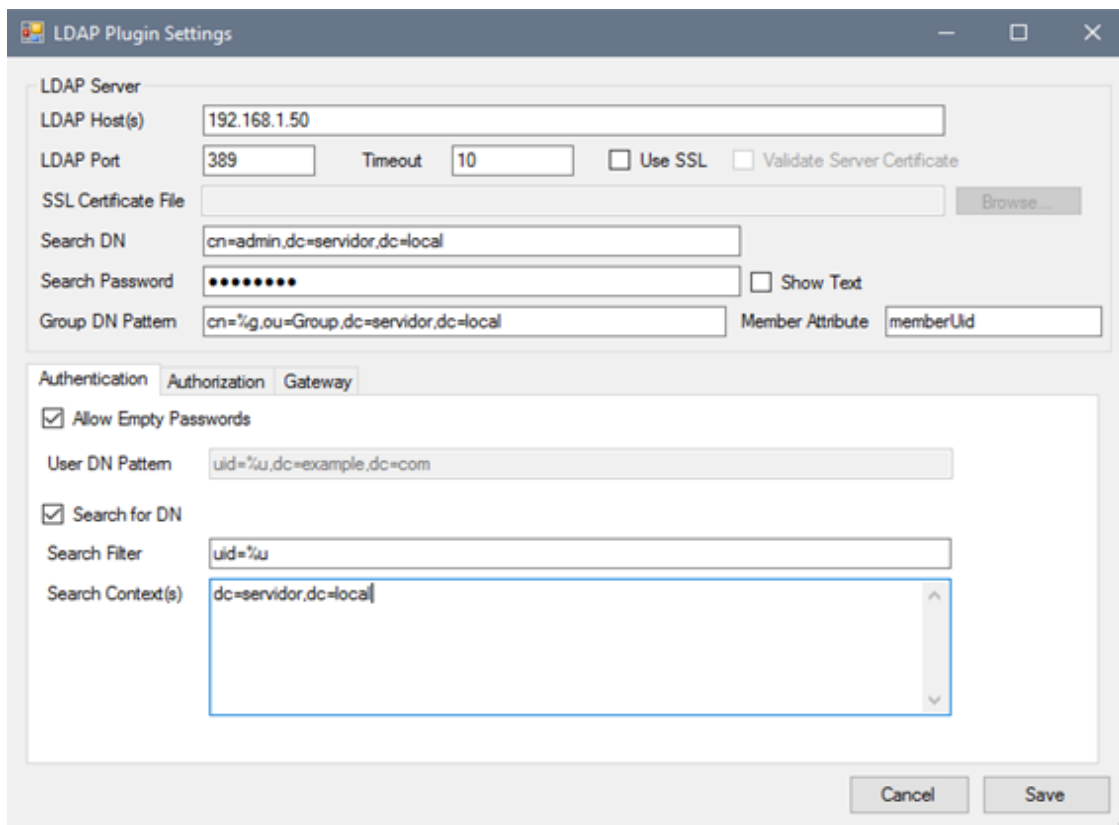


Y en la fila que pone LDAP, tendremos que activar las 3 casillas.



Y le damos a **Configure**.

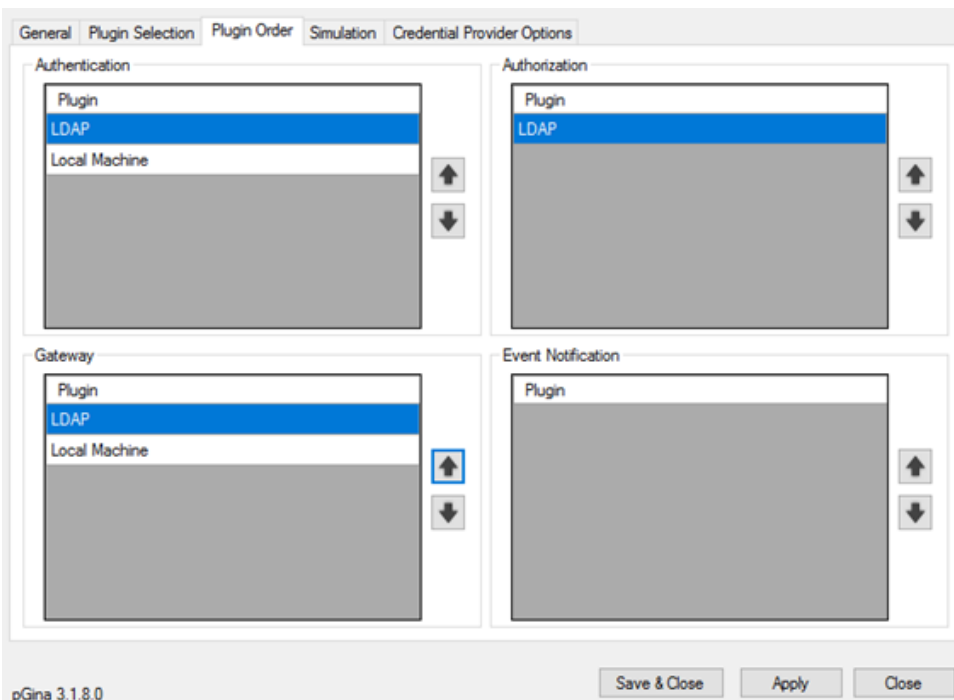
En la ventana que nos aparece a continuación tendremos que rellenar los datos de la siguiente manera:



The image shows the 'LDAP Plugin Settings' dialog box. It has a title bar with standard window controls. The main area is divided into sections. The 'LDAP Server' section contains fields for 'LDAP Host(s)' (192.168.1.50), 'LDAP Port' (389), 'Timeout' (10), 'Use SSL' (unchecked), 'Validate Server Certificate' (unchecked), 'SSL Certificate File' (empty), 'Search DN' (cn=admin,dc=servidor,dc=local), 'Search Password' (masked with dots), 'Show Text' (unchecked), 'Group DN Pattern' (cn=%g,ou=Group,dc=servidor,dc=local), and 'Member Attribute' (memberUid). Below this is a tabbed interface with 'Authentication', 'Authorization', and 'Gateway' tabs. The 'Authentication' tab is active, showing 'Allow Empty Passwords' (checked), 'User DN Pattern' (uid=%u,dc=example,dc=com), 'Search for DN' (checked), 'Search Filter' (uid=%u), and 'Search Context(s)' (dc=servidor,dc=local). At the bottom are 'Cancel' and 'Save' buttons.

Y le damos a **Save**.

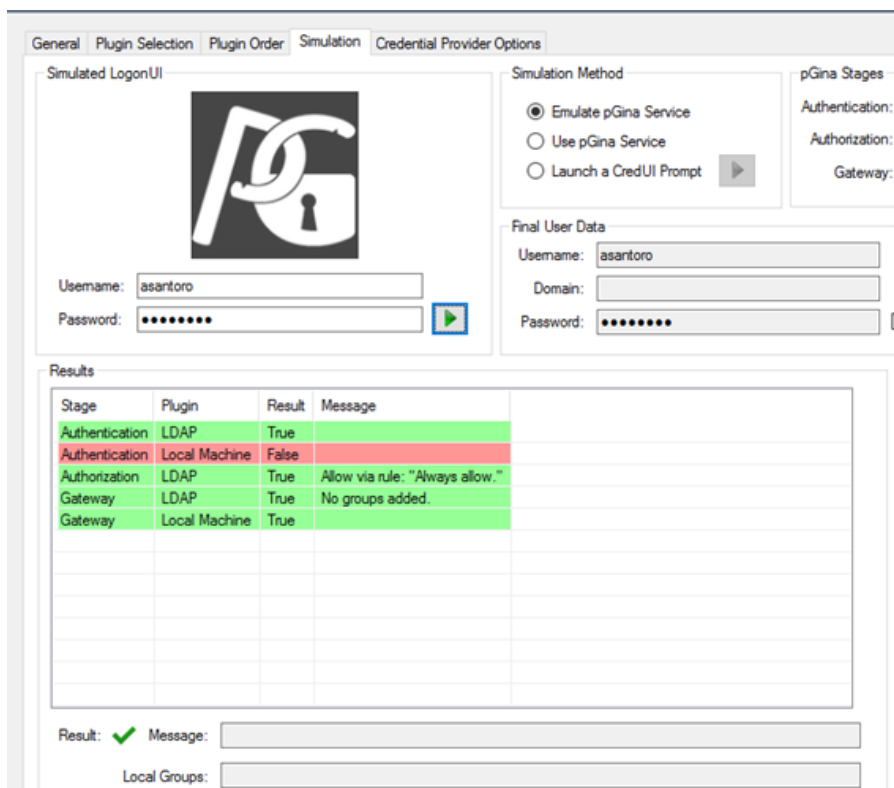
A continuación, nos iremos a la ventana **Plugin Order**, y lo configuramos como en la imagen.



The image shows the 'Plugin Order' dialog box. It has a title bar with standard window controls. The main area is divided into four sections: 'Authentication', 'Authorization', 'Gateway', and 'Event Notification'. Each section has a list of plugins. In 'Authentication', 'LDAP' is selected and 'Local Machine' is below it. In 'Authorization', 'LDAP' is selected. In 'Gateway', 'LDAP' is selected and 'Local Machine' is below it. In 'Event Notification', 'LDAP' is selected. Each list has up and down arrows for reordering. At the bottom are 'Save & Close', 'Apply', and 'Close' buttons. The version 'pGina 3.1.8.0' is displayed in the bottom left corner.

Ahora nos iremos a la ventana de **Simulation**.

Y probamos con un usuario del servidor, en nuestro caso será el usuario **asantoro**.

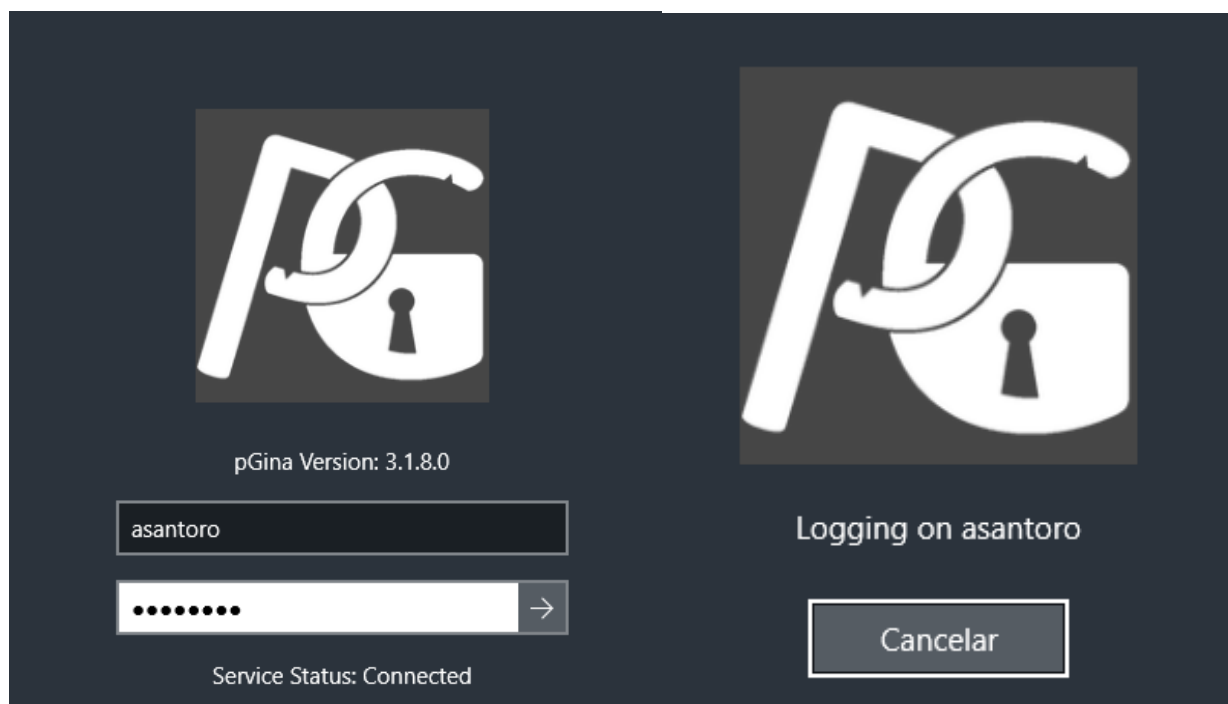


Stage	Plugin	Result	Message
Authentication	LDAP	True	
Authentication	Local Machine	False	
Authorization	LDAP	True	Allow via rule: "Always allow."
Gateway	LDAP	True	No groups added.
Gateway	Local Machine	True	

Result: ✔ Message:
 Local Groups:

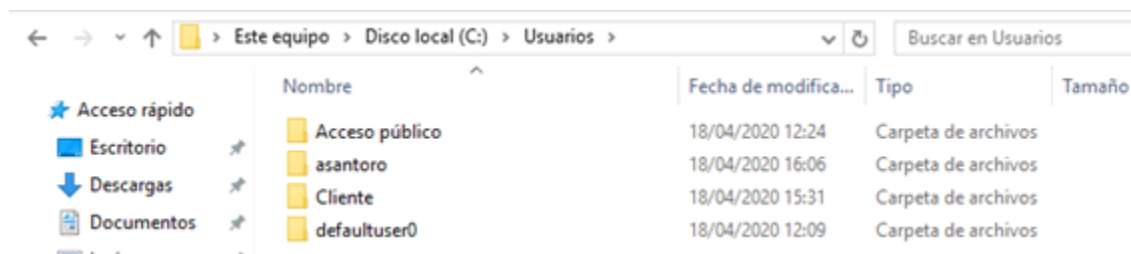
Hecho esto, reiniciamos la máquina y accedemos con el usuario **asantoro**.





Una vez hemos iniciado sesión, vamos al explorador de archivos > Disco local C: > Usuarios

Y ahí podemos ver que se nos ha creado una carpeta con el usuario **asantoro**.



2.3 Instalación PhpLdapAdmin

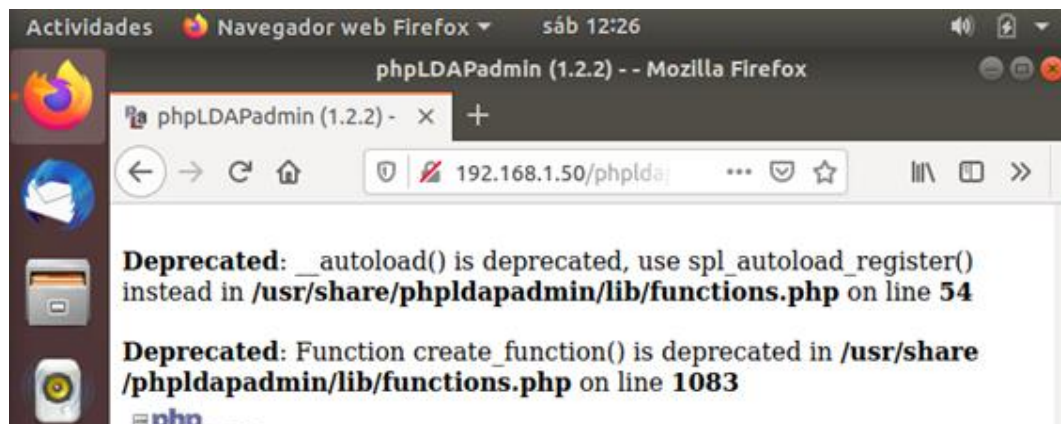
Otra manera de configurar la estructura de nuestro Ldap es a través de interfaz gráfica. Para ello tendremos que instalar PhpLdapAdmin, que es un cliente que permite administrar un servidor LDAP desde un navegador de una forma más sencilla:

```
root@servidor:~# apt-get install phpldapadmin -y
```

Una vez instalado, solo tendremos que acceder desde nuestro navegador a:

<http://192.168.1.50/phpldapadmin/>

Pero la primera vez que accedamos nos aparecerán estos dos errores:



Para solucionar estos errores, tendremos que editar el archivo **fuctions.php** y modificar las siguientes líneas:

#línea 54

```
* Loads class definition
*/
function my__autoload($className) {
    if (file_exists(HOOKSDIR."classes/$className.php"))
```

#línea 777

```
        return base64_encode($encrypt);
}
spl_autoload_register("my__autoload");
```

#línea 1083

```
        $code .= 'return $c;';

        $CACHE[$sortby] = __create_function('$a, $b', $code);
    }
}
```

Hecho esto, los errores estarían solucionados:



Le damos a conectar e introducimos los siguientes datos:



Para que nos aparezca de manera predeterminada nuestro servidor, tendremos que modificar el archivo

`/usr/share/phpldapadmin/config/config.php` y modificar las siguientes líneas:

```
/* Array of base DN's of your LDAP server. Leave this blank to have ph
auto-detect it for you. */
$servers->setValue('server','base',array('dc=servidor,dc=local'));
```

```
the directory for users (ie, if your LDAP server does not allow anonym
binds. */
$servers->setValue('login','bind_id','cn=admin,dc=servidor,dc=local');
```

Y ahora podemos verificar que cuando accedemos desde el navegador nos aparece nuestro servidor LDAP.

2.4 SSL OpenLDAP

La implementación de SSL para el OpenLDAP viene de la idea de implementar seguridad al PhpLdapAdmin cuando accedemos desde el navegador.

Para comenzar, nos dirigimos al archivo **/etc/phpldapadmin/config.php**, descomentamos y cambiamos la siguiente línea:

Con esto, lo que hacemos es desactivar los mensajes de warning:

```
/* Hide the warnings for invalid objectClasses/attributes in templates. */  
$config->custom->appearance['hide_template_warning'] = true;
```

Modificaremos las siguientes líneas para introducir los datos correspondientes a nuestro dominio.

```
(context_socket at /usr/local/var/run/ldap) /  
$servers->setValue('server','host','servidor.local');  
  
/* The port your LDAP server listens on (no quotes). 389 is standard. */  
// $servers->setValue('server','port',389);  
  
/* Array of base DNS of your LDAP server. Leave this blank to have phpLDAP  
auto-detect it for you. */  
$servers->setValue('server','base',array('dc=servidor,dc=local'));
```

A continuación, crearemos la carpeta para el autofirmado, donde se almacenarán el certificado y la key de OpenSSL.

```
$ mkdir /etc/ssl/autofirmado
```

Ahora vamos a crear el certificado y la key:


```

root@servidor:/etc/ssl# openssl req -x509 -nodes -days 365 -newkey rsa:4096 -keyout /etc/ssl/autofirma
do/cert_autofirmado.key -out /etc/ssl/autofirmado/cert_autofirmado.crt
Can't load /root/.rnd into RNG
139663706239424:error:2406F079:random number generator:RAND_load_file:Cannot open file:../crypto/rand/
randfile.c:88:Filename=/root/.rnd
Generating a RSA private key
.....++++
.....++++
writing new private key to '/etc/ssl/autofirmado/cert_autofirmado.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:MADRID
Locality Name (eg, city) []:MADRID
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:servidor.local
Email Address []:

```

Vamos a crear el fichero para la autenticación de contraseñas. Para ello necesitamos los siguientes paquetes de apache:

```
$ apt install apache2-utils
```

Ahora vamos a crear el archivo y especificar el usuario con el cual vamos a querer acceder.

```

root@servidor:/etc/ssl# htpasswd -c /etc/apache2/htpasswd admin
New password:
Re-type new password:
Adding password for user admin

```

Habilitamos el módulo ssl

```
$ a2enmod ssl
```

Editamos el archivo /etc/phpldapadmin/apache para agregar el alias con el cual queremos acceder al PhpLdapAdmin.

```

GNU nano 2.9.3 /etc/phpldapadmin/apache.conf

# Define /phpldapadmin alias, this is the default
<IfModule mod_alias.c>
    Alias /superldap /usr/share/phpldapadmin/htdocs
</IfModule>

```

Editamos el archivo /etc/apache2/sites-enabled/000-default.conf

GNU nano 2.9.3 /etc/apache2/sites-enabled/000-default.conf

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com
ServerName servidor.local

ServerAdmin webmaster@servidor.local
DocumentRoot /var/www/html
ServerName 192.168.1.50
Redirect permanent /superldap https://servidor.local/superldap

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
```

Recargamos la configuración añadida anteriormente

```
sudo a2ensite default-ssl.conf
```

Editamos nuevamente el archivo /etc/apache2/sites-enabled/default-ssl.conf

GNU nano 2.9.3 /etc/apache2/sites-enabled/default-ssl.conf

```
<IfModule mod_ssl.c>
<VirtualHost _default_:443>
    ServerAdmin webmaster@servidor.local
    ServerName servidor.local

    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

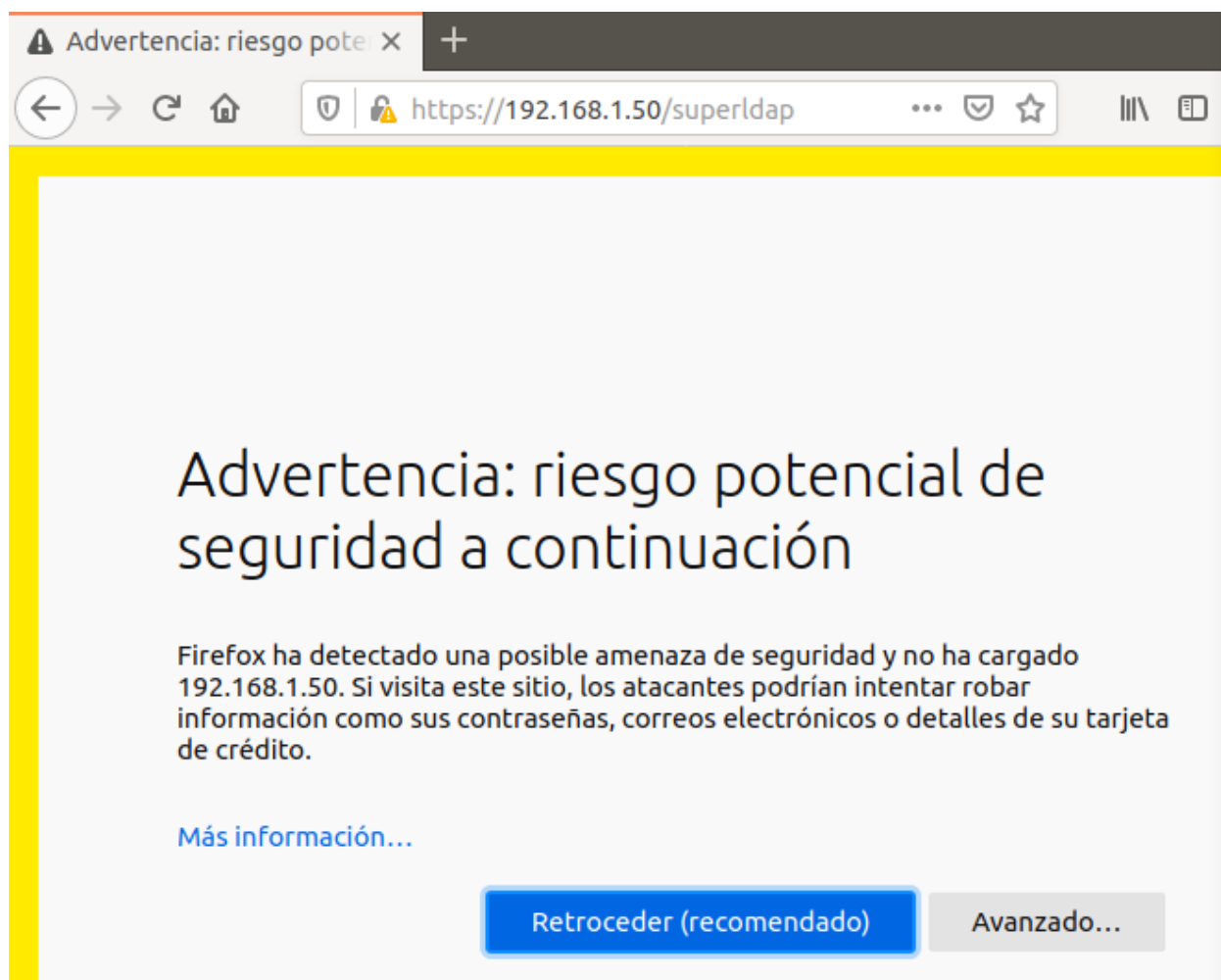
    ErrorLog ${APACHE_LOG_DIR}/superldap_error.log
    CustomLog ${APACHE_LOG_DIR}/superldap_access.log combined
```

```
SSLCertificateFile /etc/ssl/autofirmado/cert_autofirmado.crt
SSLCertificateKeyFile /etc/ssl/autofirmado/cert_autofirmado.key
```

```
<Location /superldap>
  AuthType Basic
  AuthName "Restricted Files"
  AuthUserFile /etc/apache2/htpasswd
  Require valid-user
</Location>
```


Reiniciamos el servicio de apache para que se apliquen los cambios.

Desde nuestro navegador accedemos a 192.168.1.50/superldap, donde nos saldrá una advertencia de la cual no hace falta preocuparse ya que la muestra porque el certificado ha sido creado por nosotros mismos. Avanzado... → Aceptar el riesgo y continuar.



A partir de ahora al acceder a nuestro PhpLdapAdmin tendremos que autenticarnos con la contraseña que hemos puesto antes en htpasswd.

Se requiere autenticación - Mozilla Firefox



https://192.168.1.50 solicita su nombre de usuario y contraseña. El sitio dice:
"Restricted Files"

Nombre de usuario:

Contraseña:

2.5 Estructura del LDAP

En cuanto a la estructura de nuestro LDAP hemos decidido dividirla principalmente en dos unidades organizativas que serán grupos y usuarios (dividida a su vez en alumnos y profesores). Los grupos se utilizarán posteriormente en samba para la asignación de permisos en las carpetas.

Para la organización de los usuarios, hemos utilizado diferentes rangos del UID según el tipo de usuario que sea:

- Usuarios locales (1000-1019).
- Usuarios Profesores (1020-1039).
- Usuarios Alumnos (1040-1059).



3 SAMBA

3.1 Introducción

Samba es un proyecto de software libre que implementa el protocolo de archivos compartidos de Windows para Sistemas operativos de tipo UNIX.

El servidor Samba ofrece los siguientes servicios:

- Compartir uno o varios sistemas de archivos
- Compartir uno o varios sistemas de archivos distribuidos
- Compartir impresoras instaladas en el servidor entre los clientes Windows de la red
- Ayudar a los clientes permitiéndoles navegar por la red
- Autenticar a los clientes que ingresan en un dominio Windows

3.2 Instalación

Para la instalación en el servidor, solo hará falta instalar lo siguiente:

```
$ apt-get install samba
```

Con esta instalación deberemos tener activos los servicios `smbd`.

Para la instalación en el **cliente**, solo hará falta instalar lo siguiente:

Antes de todo, explicar, que en la instalación de samba en el cliente se hará de diferente manera a la instalación en el servidor para evitar un futuro fallo al obtener la lista de compartición del servidor.

```
$ apt-get install samba samba-common smbclient winbind
```

Winbind es un componente de la suite de programas Samba que resuelve los problemas de inicio de sesión unificados.

3.3 Configuración

-En el servidor

Iremos a la ruta `/etc/samba` y modificaremos el archivo de configuración **smb.conf**.

Al final del archivo añadiremos las directivas correspondientes a nuestra carpeta compartida:

```
[Recursos Compartidos]
comment = Recursos compartidos de profesores y alumnos
path = /home/server/Escritorio/Recursos
browseable = yes
writeable = yes
guest ok = no
valid users = @Profesores @Alumnos
write list = @Profesores
create mask = 0755
directory mask = 0755
```

También habrá que comprobar quién es el propietario de la carpeta, tanto el usuario como el grupo.

Por defecto, aparecerá que el propietario tanto de usuario como grupo será **server**, que es con quien hemos creado esta carpeta.

Nosotros lo modificaremos de tal manera que solo tenga de propietario al grupo **Profesores**, para que todos los usuarios Profesores que pertenezcan a ese grupo pueda gestionar dicha carpeta.

Para ello usaremos el siguiente comando:

```
$ chown nobody:Profesores Recursos
```

Y se vería así:

```
root@servidor:/home/server/Escritorio# ls -la
total 48
drwxrwxrwx  3 server server    4096 abr 27 12:08 
drwxrwxrwx 20 server server    4096 may 23 16:40 
-rw-rw-r--  1 server server   36746 abr 27 12:07 fondo.jpg
drwxrwxr-x  4 nobody Profesores 4096 may 20 16:22 Recursos
```

- En el cliente

Editamos el archivo **/etc/samba/smb.conf**:

```
$ gedit /etc/samba/smb.conf
```

En cuanto a la siguiente directiva que aparece comentada, eliminamos el ; o sino aparece la agregamos dentro de la sección [global].

```
name resolve order= lmhosts hosts wins bcast
```

Editamos el archivo **/etc/nsswitch.conf**.

```
$ gedit /etc/nsswitch.conf
```

Editamos la línea hosts para dejarlo de la siguiente manera.

```
Hosts: files mdns4_minimal [NOTFOUND=return] wins dns mdns4
```

Por último, reiniciamos los demonios **samba** y **winbind**.

```
$ /etc/init.d/winbind restart
```

```
$ /etc/init.d/samba restart
```

3.4 Creación usuarios y grupos Samba

Para la creación de usuarios de Samba lo primero que tenemos que tener en cuenta es, que los usuarios deben estar ya creados previamente de manera local.

Para crear un usuario de Samba hay que usar el siguiente comando:

```
$ smbpasswd -a asantoro
```

Una vez creados los usuarios que queramos tener, para poder ver un listado de ellos, usaremos el comando **\$ pdbedit -L**

```
server@servidor:~/Escritorio$ sudo pdbedit -L
rbautista:1020:ruben bautista
pepito:1002:
amora:1021:Amadeo Mora
administrador:1001:
asantoro:1040:Alejandro Santoro
ajuarez:1041:Adrian Juarez
```

Estos usuarios los queremos dividir en los grupos **Profesores** y **Alumnos**, los cuales tienen que haber sido creados previamente.

Para añadir los usuarios a los diferentes grupos usaremos el siguiente comando:

```
$ usermod -a -G Alumnos ajuarez
```

```
$ usermod -a -G Profesores amora
```

De esta manera ya tendremos a cada usuario en su respectivo grupo.

3.5 Crear carpetas compartidas usando NFS

3.5.1 ¿Qué es NFS?

NFS (Network File System - Sistema de Archivos en Red) es el sistema nativo utilizado por Linux para compartir carpetas en una red. Desde los PCs de los usuarios se puede acceder a dichas carpetas compartidas y el resultado es el mismo que si estuvieran en su propio disco duro.

3.5.2 Instalación

Para la instalación de NFS en Linux necesitamos los siguientes paquetes: `nfs-kernel-server`, `nfs-common`, `rpcbind`. Estos paquetes serán instalados en el servidor.

```
$ apt-get install nfs-kernel-server nfs-common rpcbind
```

Una vez instalado, comprobamos que se ha instalado correctamente.

```
root@servidor:/home/servidor# grep nfsd /proc/filesystems
nodev    nfsd
```

En el cliente únicamente instalaremos los paquetes **nfs-common** y **rpcbind**.

```
$ apt-get install nfs-common rpcbind
```

El paquete **nfs-common** son archivos de apoyo a NFS comunes al cliente y al servidor.

rpcbind es un dominio del paquete NFS quien permite que los clientes NFS descubran qué puerto está utilizando el servidor NFS.

3.5.3 Crear y compartir una carpeta con NFS.

Creamos la carpeta, asignamos propietario y grupo, y asignamos los permisos a la carpeta en función de nuestras necesidades.

```
$ mkdir /compartido
$ chown nobody:nogroup /compartido
$ chmod -R 755 /compartido
```

A continuación, exportamos el contenido de las carpetas añadiendo en el archivo `/etc/exports` las siguientes líneas:

```
/home      *(rw,sync,no_root_squash,no_subtree_check)
/compartido *(rw,sync,no_subtree_check)
```

Los valores que hemos añadido son:

- **rw (read-write)**: El usuario podrá realizar cambios en el contenido de la carpeta compartida.
- **sync**: Evita responder peticiones antes de escribir los cambios pendientes en disco. Es la opción predeterminada.
- **root_squash**: Evita que los usuarios con privilegios administrativos los mantengan, sobre la carpeta compartida, cuando se conectan remotamente. En su lugar, se les trata como a un usuario remoto más. Es la opción predeterminada. (Esta la hemos negado para deshabilitar lo anterior).

- **subtree_check**: Cuando el directorio compartido es un subdirectorio de un sistema de archivos mayor, NFS comprueba los directorios por encima de éste para verificar sus permisos y características. Es la opción predeterminada. (Esta la hemos negado para deshabilitar lo anterior).

Montar las carpetas **home** y **compartido** en el cliente

Creamos el punto de montaje de las carpetas compartidas:

```
root@cliente:~# mkdir -p /mnt/nfs/home
root@cliente:~# mkdir -p /mnt/nfs/compartido
```

Para montar las carpetas utilizaremos el comando:

```
$ mount.
```

```
root@cliente:~# mount 192.168.1.50:/home /mnt/nfs/home
root@cliente:~# mount 192.168.1.50:/home /mnt/nfs/compartido/
```

En caso de que nos salga un error al montar diciéndonos que el acceso está denegado únicamente deberemos de ejecutar el comando **exportfs -a** utilizado para comparticiones puntuales. Para comprobar que se ha montado correctamente podemos ejecutar el comando **mount** o **df -h**.

Ahora haremos que las carpetas se monten automáticamente siempre que iniciemos el cliente.

Para esto editaremos el archivo **/etc/fstab** añadiendo las siguientes líneas:

```
192.168.1.50:/home /mnt/nfs/home nfs auto,noatime,nolock,bg,nfsvers=3,intr,tcp,actimeo=1800 0 0
192.168.1.50:/compartido /mnt/nfs/compartido/ nfs
auto,noatime,nolock,bg,nfsvers=3,intr,tcp,actimeo=1800 0 0
```

Reiniciamos el cliente y comprobamos que seguimos teniendo acceso a las carpetas compartidas.

4 SSH

4.1 Introducción

Secure Shell es un protocolo seguro de administración remota que le permite a los usuarios controlar y modificar sus servidores remotos a través de Internet a través de un mecanismo de autenticación. SSH establece conexiones seguras entre los dos sistemas.

4.2 Instalación

Para instalarlo tendremos que usar el siguiente comando:

```
$ apt install ssh
```

Para verificar que se instaló correctamente, vamos a probar a conectarnos desde el servidor al usuario **ajuarez** del cliente ubuntu:

```
server@servidor:~$ ssh ajuarez@192.168.1.60
The authenticity of host '192.168.1.60 (192.168.1.60)' can't be established.
ECDSA key fingerprint is SHA256:XsegSa9xElklNMC3WpeVYbZuyloa0he2ERxPnE7c3Ms.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.60' (ECDSA) to the list of known hosts.
ajuarez@192.168.1.60's password:
Creating directory '/home/alumnado/ajuarez'.
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-99-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

Pueden actualizarse 0 paquetes.
0 actualizaciones son de seguridad.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

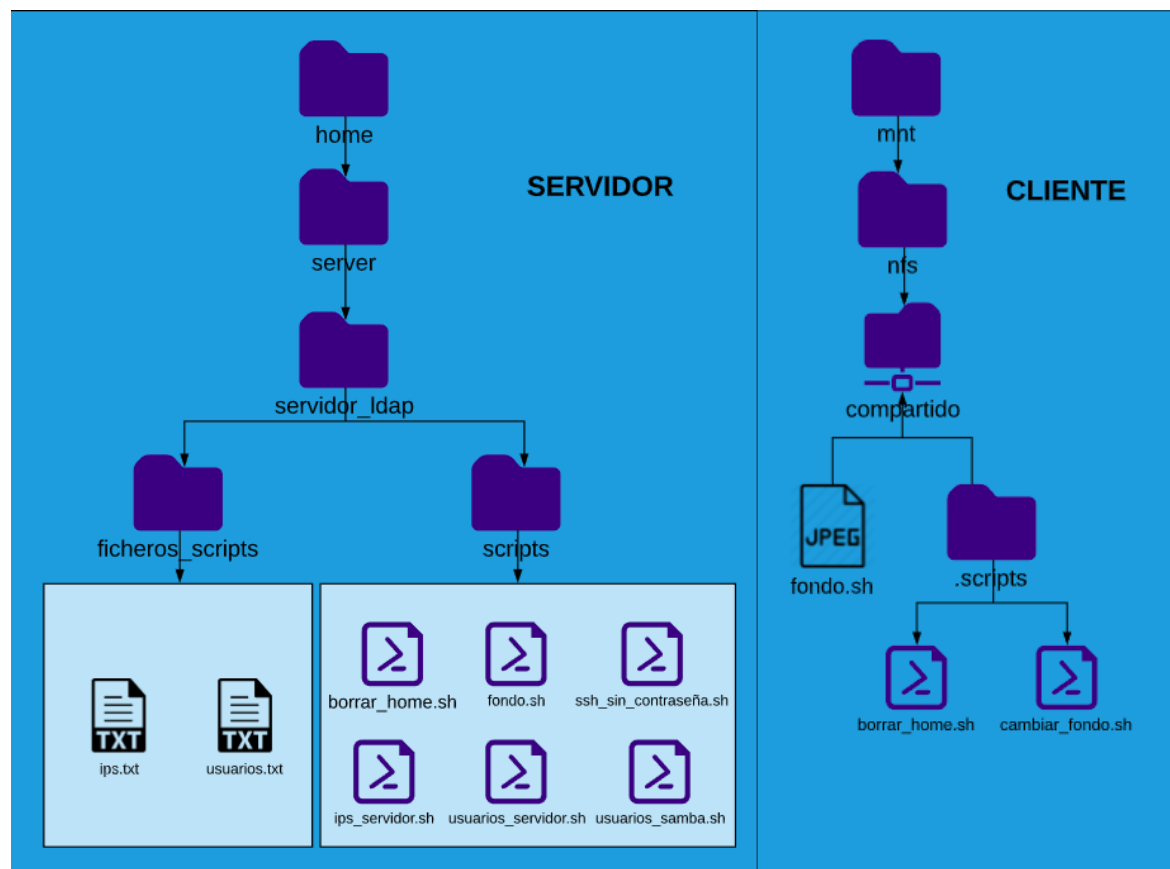
Last login: Thu Apr 30 14:18:17 2020 from 192.168.1.50
ajuarez@cliente:~$
```

5 Scripts para automatización de tareas

Los scripts principalmente han sido creados en el servidor ya que es desde donde se ejecutarán, aunque para que algunos funcionen correctamente ha sido necesario crear scripts complementarios ejecutados a través de ssh.

Como medida de seguridad hemos establecido los permisos para que los usuarios tales como los alumnos no tengan permisos para modificar estos scripts.

5.1 Estructura



5.2 Scripts de automatización de tareas

1. Script establecer fondo de escritorio en todos los usuarios conectados al servidor.

Script servidor (fondo.sh):

```
#!/bin/bash

# usuarios del servidor
./usuarios_servidor.sh

# ips conectadas al servidor
./ips_servidor.sh

# Cambiar fondo
while read usuario
do
    while read ips
    do
        ssh -tt ${usuario}@${ips} /mnt/nfs/compartido/.scripts/
prueba.sh exit
        echo "Fondo establecido"
        echo ""

    done < /home/server/servidor_ldap/ficheros_scripts/ips.txt
done < /home/server/servidor_ldap/ficheros_scripts/usuarios.txt

exit 0
```

Script cliente (cambiar_fondo.sh):

```
#!/bin/bash
dconf write "/org/gnome/desktop/background/picture-uri" "'/mnt/nfs/compartido/
fondo.jpg'"
```

2. Script ssh para que se conecte a los clientes sin pedir contraseña (ssh_sin_contraseña.sh).

```
#!/bin/bash

usuarios=`ldapsearch -z 0 -H ldap://localhost:389 -w servidor -D
"cn=admin,dc=servidor,dc=local" -b "dc=servidor,dc=local" "(uid=*)" | grep uid:
| cut -d " " -f2 > /home/server/servidor_ldap/ficheros_scripts/usuarios.txt`

guardar_ip=`netstat -atn | grep 50:389 | tr -s " " | cut -d " " -f5 | cut -d ":"
-f1 | uniq > /home/server/servidor_ldap/ficheros_scripts/ips.txt`

ssh-keygen -b 4096 -t rsa
|
while read usuario
do
    while read ips
    do
        ssh-copy-id ${usuario}@${ips}
    done < /home/server/servidor_ldap/ficheros_scripts/ips.txt
done < /home/server/servidor_ldap/ficheros_scripts/usuarios.txt

exit 0
```

3. Borrar home de usuarios eliminados (borrar_home.sh).

```
#!/bin/bash

./ips_servidor.sh

touch /compartido/usuario.txt
chmod +x /compartido/usuario.txt
touch /compartido/opcion.txt
chmod +x /compartido/opcion.txt

echo "¿Que deseas borrar?: alumno/profesor/otro: "
read opcion
echo ""
if [ $opcion == "alumno" ]
then
    echo "Has elegido borrar a un alumno."
    echo "¿De qué alumno quieres borrar el home:"
    read usuario
    echo ""
    echo "alumnado" > /compartido/opcion.txt
    echo "$usuario" > /compartido/usuario.txt

elif [ $opcion == "profesor" ]
then
    echo "Has elegido borrar a un profesor."
    echo "¿De qué profesor quieres borrar el home:"
    read usuario
    echo ""
    echo "profesorado" > /compartido/opcion.txt
    echo "$usuario" > /compartido/usuario.txt

elif [ $opcion == "otro" ]
then
    echo "Has elegido borrar otro."
    echo "carpeta donde se guarda su home"
    read carpeta
    echo "¿De qué usuario quieres borrar el home:"
    read usuario
    echo ""
    echo $carpeta > /compartido/opcion.txt
    echo "$usuario" > /compartido/usuario.txt
fi

while read ips
do
    echo cliente | ssh -tt cliente@${ips} "sudo /mnt/nfs/
compartido/.scripts/borrar_home.sh" exit
done < /home/server/servidor_ldap/ficheros_scripts/ips.txt

rm /compartido/usuario.txt
rm /compartido/opcion.txt
```

Script en el cliente (borrar_home.sh).

```
#!/bin/bash

while read usuario
do
    while read opcion
    do
        cd /home/$opcion
        rm -r $usuario
    done < /mnt/nfs/compartido/opcion.txt
done < /mnt/nfs/compartido/usuario.txt
```

4. Script para sacar las ips conectadas al servidor (ips_servidor.sh).

```
#!/bin/bash
netstat -atn | grep 50:389 | tr -s " " | cut -d " " -f5 | cut -d ":" -f1 | uniq
> /home/server/servidor_ldap/ficheros_scripts/ips.txt
```

5. Usuarios conectados al servidor (usuarios_servidor.sh)

```
#!/bin/bash
ldapsearch -z 0 -H ldap://localhost:389 -w servidor -D
"cn=admin,dc=servidor,dc=local" -b "dc=servidor,dc=local" "(uid=*)" | grep uid:
| cut -d " " -f2 > /home/server/servidor_ldap/ficheros_scripts/usuarios.txt
```

6. Añadir usuarios del servidor a samba (usuarios_samba.sh).

```
#!/bin/bash

passwd=servidor

# usuarios del servidor
./usuarios_servidor.sh

#Añadir usuarios a samba
while read usuario
do
    yes ${passwd} | head -2 | smbpasswd -a ${usuario}
    echo ${usuario} añadido
done < /home/server/servidor_ldap/ficheros_scripts/usuarios.txt

exit 0
```

6 Zabbix

Antes de todo debemos tener en cuenta la versión que vamos a instalar, en nuestro caso instalaremos la versión 4.4.8 en todas las máquinas.

6.1 Introducción

Zabbix es un sistema para monitorear la capacidad, el rendimiento y la disponibilidad de los servidores, equipos, aplicaciones y bases de datos. Además, ofrece características avanzadas de monitoreo, alertas y visualización, que incluso, algunas de las mejores aplicaciones comerciales de este tipo no ofrecen.

Usa MySQL, PostgreSQL, SQLite, Oracle o IBM DB2 como base de datos. Su backend está escrito en C y el frontend web está escrito en PHP.

¿Qué se puede hacer con Zabbix?

- Agregar y monitorear servidores, equipos, servicios, aplicaciones específicas, dispositivos físicos como impresoras, routers, entre otros.
- Reporte en tiempo real a través de gráficas, datos y alertas visuales que muestran el estado y rendimiento de los servicios y equipos monitoreados.
- Inventario de equipos para mantener al día la infraestructura tecnológica
- Configuración de permisos por usuarios y grupos.
- Mapas de la red.
- Configuración de notificaciones vía correo electrónico.
- Perfiles de usuarios para el uso del administrador Web.

Ventajas de Zabbix.

- Interfaz basada en la web.
- Reportes detallados.
- Fácil configuración.
- Estadísticas en tiempo real del estado de los servidores/máquinas.
- Reduce los costos de operación al evitar el tiempo de inactividad.

6.2 Instalación.

6.2.1 Zabbix Server (en el servidor).

El servidor de zabbix será el responsable de monitorizar a todos los clientes de zabbix. Esto se hará gracias a una interfaz que proporciona el propio Zabbix donde se irá mostrando a tiempo real, todo lo que va sucediendo en cada cliente de este servidor.

Desde la página oficial de Zabbix podemos descargar el ejecutable .deb.

```
$ wget https://repo.zabbix.com/zabbix/4.4/ubuntu/pool/main/z/zabbix-release/zabbix-release_4.4-1+bionic_all.deb
```

Instalamos el paquete con el siguiente comando:

```
$ dpkg -i zabbix-release_4.4-1+bionic_all.deb
```

Actualizamos la lista de paquetes.

```
$ apt update
```

Instalamos los siguientes paquetes: **zabbix-server-mysql** para la base de datos que manejará el servidor de Zabbix, **zabbix-frontend-php** para la instalación del frontend (interfaz gráfica), **zabbix-apache-conf** para la configuración de apache y **zabbix-agent** que es para gestión de hosts.

```
$ apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent
```

Creamos la base de datos que va a utilizar Zabbix, creamos el usuario que va a administrar dicha base de datos y le concedemos todos los permisos a dicho usuario.

```
root@zabbix:/home/zabbix# mysql -uroot -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 2516
Server version: 10.1.44-MariaDB-0ubuntu0.18.04.1 Ubuntu 18.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]> █
```

```
MariaDB [(none)]> create database zabbix character set utf8 collate utf8_bin;
```

```
MariaDB [(none)]> create user zabbix@localhost identified by 'servidor';
```

```
MariaDB [(none)]> grant all privileges on zabbix.* to zabbix@localhost;
```

Importamos los datos del esquema inicial y de los datos de Zabbix.

```
$ zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix
```

Ahora vamos a modificar el archivo de configuración del Zabbix server (**/etc/zabbix/zabbix_server.conf**) para configurar el usuario y la contraseña con los cuales accederemos a Zabbix server. Tendremos que modificar las siguientes líneas:

```
DBUser=zabbix
```



```
DBPassword=servidor
```

Editamos el archivo **/etc/zabbix/apache.conf** para cambiar los valores de los módulos de php para evitar futuros errores tales como de memoria, zona geográfica...

```
GNU nano 2.9.3 /etc/zabbix/apache.conf

# Define /zabbix alias, this is the default
<IfModule mod_alias.c>
    Alias /zabbix /usr/share/zabbix
</IfModule>

<Directory "/usr/share/zabbix">
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all

    <IfModule mod_php5.c>
        php_value max_execution_time 600
        php_value memory_limit 128M
        php_value post_max_size 32M
        php_value upload_max_filesize 16M
        php_value max_input_time 600
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone Europe/Madrid
    </IfModule>
    <IfModule mod_php7.c>
        php_value max_execution_time 600
        php_value memory_limit 128M
        php_value post_max_size 32M
        php_value upload_max_filesize 16M
        php_value max_input_time 600
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone Europe/Madrid
    </IfModule>
</Directory>
```

Reiniciamos y habilitamos tanto el servicio de Zabbix como el servicio de apache.

```
$ systemctl restart zabbix-server zabbix-agent apache2
```

```
$ systemctl enable zabbix-server zabbix-agent apache2
```

Accedemos a **localhost/zabbix** desde nuestro navegador para instalar el frontend.

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Welcome to

Zabbix 4.4

[Back](#)
[Next step](#)

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Check of pre-requisites

	Current value	Required	
PHP version	7.2.24-0ubuntu0.18.04.4	5.4.0	OK
PHP option "memory_limit"	128M	128M	OK
PHP option "post_max_size"	32M	16M	OK
PHP option "upload_max_filesize"	16M	2M	OK
PHP option "max_execution_time"	600	300	OK
PHP option "max_input_time"	600	300	OK
PHP option "date.timezone"	Europe/Madrid		OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK

[Back](#)
[Next step](#)

Introducimos la contraseña del usuario Zabbix y el resto lo dejamos como viene predeterminado.

ZABBIX

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type	<input type="text" value="MySQL"/>
Database host	<input type="text" value="localhost"/>
Database port	<input type="text" value="0"/> 0 - use default port
Database name	<input type="text" value="zabbix"/>
User	<input type="text" value="zabbix"/>
Password	<input type="password" value="....."/>

[Back](#) [Next step](#)

ZABBIX

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Zabbix server details

Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional).

Host	<input type="text" value="localhost"/>
Port	<input type="text" value="10051"/>
Name	<input type="text"/>

[Back](#) [Next step](#)

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Database type MySQL
Database server localhost
Database port default
Database name zabbix
Database user zabbix
Database password *****

Zabbix server localhost
Zabbix server port 10051
Zabbix server name

[Back](#)
[Next step](#)

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Install

Congratulations! You have successfully installed Zabbix frontend.

Configuration file "/usr/share/zabbix/conf/zabbix.conf.php" created.

[Back](#)
[Finish](#)

6.2.2 Zabbix Agent en el cliente.

En cada cliente que queramos monitorizar, tendremos que instalar este agente. Estos agentes irán enviando información a tiempo real al servidor de Zabbix, el cual irá mostrando en su interfaz toda esta información.

Descargamos el paquete de la página web de Zabbix como hicimos con el servidor.

Instalamos el agente de Zabbix:

```
$ apt install zabbix-agent
```

Una vez instalado, activamos el agente de Zabbix.

```
$ update-rc.d zabbix-agent enable
```

Y arrancamos el servicio.

```
$ /etc/init.d/zabbix-agent start
```

Ahora vamos a editar el archivo `/etc/zabbix/zabbix_agentd.conf` para conectarlo a nuestro servidor zabbix. Para ello modificaremos las siguientes líneas:

```
Server=192.168.1.80
```

```
ServerActive=192.168.1.80
```

Habilitamos el puerto 10050 por el que se conectará Zabbix.

```
$ ufw allow 10050/tcp
```

Y ya solo nos quedaría reiniciar el servicio y habilitarlo. Para ello usaríamos los comandos:

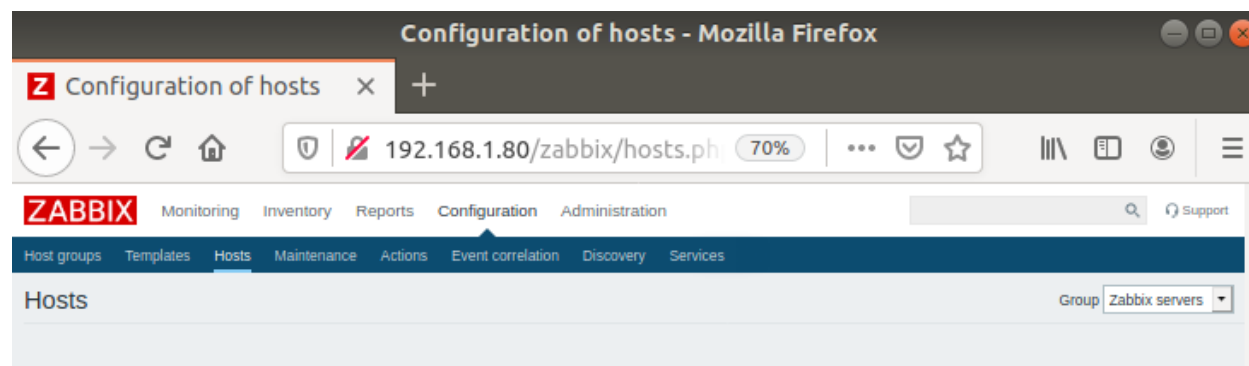
```
$ systemctl restart zabbix-agent
```

```
$ systemctl enable zabbix-agent
```

6.3 Añadir clientes (Zabbix Agent / hosts) al servidor Zabbix


Para ello, entramos con nuestro usuario Admin en nuestro servidor Zabbix.

Vamos a la pestaña de **Configuration** → **Hosts**.



En la parte derecha, en la opción de **Group**, elegimos el grupo **Zabbix servers** y le damos a **Create host**:

Group Zabbix servers Create host Import

Filter 

Introducimos el nombre que deseemos dar al cliente y añadimos la ip de la máquina.

Hosts

Host
Templates
IPMI
Macros
Host inventory
Encryption

* Host name

cliente

Visible name

cliente

* Groups

Zabbix servers ×

type here to search

Select

* At least one interface must exist.

Agent interfaces

IP address	DNS name	Connect to	Port	Default
192.168.1.60		IP DNS	10050	<input checked="" type="radio"/> Remove

Add

Vamos a la pestaña de **Templates** y seleccionamos la plantilla **Template OS Linux by Zabbix agent**, tras seleccionarla le damos a **Select**.

☐ Template OS HP-UX

☐ Template OS Linux by Prom

☒ Template OS Linux by Zabbix agent

☐ Template OS Linux by Zabbix agent active

Para finalizar le damos a **Add** para añadir el nuevo host.

En la ventana de hosts nos aparecerán los hosts que vayamos agregando.

<input type="checkbox"/>	Name ▲	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availa
<input type="checkbox"/>	cliente ldap	Applications 12	Items 64	Triggers 19	Graphs 11	Discovery 4	Web	192.168.1.60:10050		Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)	Enabled	ZBX
<input type="checkbox"/>	servidor ldap	Applications 12	Items 50	Triggers 18	Graphs 9	Discovery 3	Web	192.168.1.50:10050		Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)	Enabled	ZBX
<input type="checkbox"/>	Zabbix server	Applications 15	Items 108	Triggers 56	Graphs 19	Discovery 3	Web	127.0.0.1:10050		Template App Zabbix Server, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent,	Enabled	ZBX

6.4 Monitorizar servicios a través de plantillas.

En este caso, nos interesa saber en todo momento el estado de los servicios de Apache, SSH y LDAP.

6.4.1 Servicio SSH

Para ver su estado tendremos que añadir su plantilla correspondiente. Para ello iremos a la pestaña de **Hosts** y en el listado con todos los hosts que tenemos, seleccionar al que queremos añadir dicha plantilla.

En el listado de **Templates**, visto anteriormente, tendremos que elegir la siguiente opción:



Una vez seleccionada, solo nos quedará darle a **Update** para aplicar los cambios realizados.

Y con esto ya podríamos saber el estado del servicio en los equipos a los que se lo hayamos añadido.

6.4.2 Servicio Apache

Como hicimos con el servicio SSH, solo tendremos que añadir una plantilla para monitorear el servicio Apache. Iremos de nuevo a la pestaña de **Hosts**, y una vez ahí elegir dentro del listado de **Templates** la siguiente plantilla:

☒ Template App Apache by Zabbix agent

Le daremos a **Update** y se aplicarán los cambios realizados.

6.4.3 Servicio LDAP

Se podría decir que el servicio más relevante a monitorizar es LDAP, ya que necesitaremos saber en todo momento si está corriendo correctamente, o en su defecto si está caído.

Lo único que debemos hacer es añadir la plantilla **Template App LDAP Service** en el host servidor Idap para empezar a monitorizar dicho servicio.



Después le volveremos a dar a **Update** y se aplicarán de nuevo los cambios realizados.

6.5 Ver el almacenamiento de los equipos.

Para ver en todo momento el almacenamiento ocupado de los equipos, vamos a crear nuestro propio gráfico.

Vamos a ir a la pestaña de **Configuration** -> **Hosts**. Elegimos el equipo al cual queremos aplicar nuestro futuro gráfico.

En este caso, nosotros elegiremos el **servidor Idap**.

<input type="checkbox"/>	Name	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy
<input type="checkbox"/>	cliente ldap	Applications 14	Items 73	Triggers 25	Graphs 12	Discovery 4	Web 192.168.1.60:10050		
<input type="checkbox"/>	servidor ldap	Applications 17	Items 91	Triggers 30	Graphs 18	Discovery 4	Web 192.168.1.50:10050		
<input type="checkbox"/>	Zabbix server	Applications 15	Items 108	Triggers 56	Graphs 19	Discovery 3	Web 127.0.0.1:10050		

Una vez seleccionemos **servidor ldap**, le daremos a **Graphs**.

All hosts / servidor ldap Enabled ZBX SNMP JMX IPMI Applications 17 Items 91 Triggers 30 Graphs 18 Discovery rules 4 Web scenarios

Host Templates IPMI Tags Macros Inventory Encryption

* Host name servidor ldap

Elegimos el grupo y el host que queremos y le damos a **Create graph**.

Group Zabbix servers Host servidor ldap Create graph

Nos aparecerán los siguientes campos para rellenar:

Graph Preview

* Name

* Width 900

* Height 200

Graph type Normal

Show legend ☒

Show working time ☒

Show triggers ☒

Percentile line (left) ☐

Percentile line (right) ☐

Y axis MIN value Calculated

Y axis MAX value Calculated

* Items

Name	Function	Draw style	Y axis side	Colour	Action
Add					

Add Cancel

Rellenamos el nombre que queramos darle a nuestro gráfico, el tamaño del gráfico y los items que queremos que muestre. En nuestro caso nos mostrará lo siguiente:

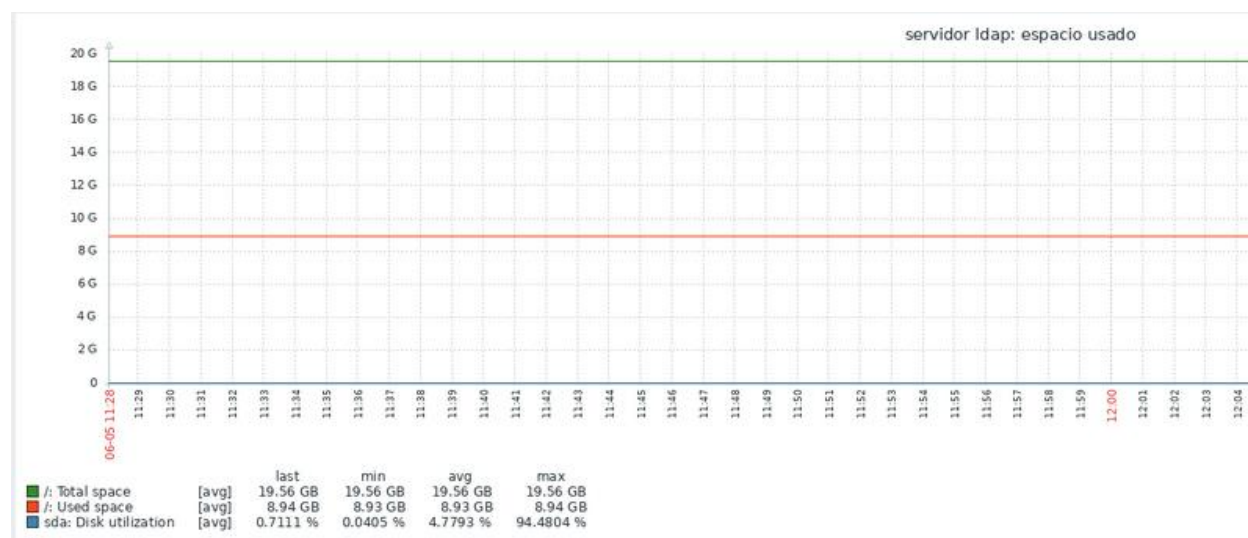
* Items	Name	Function	Draw style	Y axis side	Colour	Action
1:	servidor ldap: /: Total space	avg	Line	Left	1A7C11	Remove
2:	servidor ldap: /: Used space	avg	Line	Left	F63100	Remove
3:	servidor ldap: sda: Disk utilization	avg	Line	Left	2774A4	Remove

Elegidos todos los que queramos que se muestren, le damos a **Add** y ya estaría creado.

Para mostrarlo, tendremos que ir a **Monitoring -> Graphs** y en la esquina superior derecha elegir lo siguiente:

Group **Zabbix servers** Host **servidor ldap** Graph **espacio usado** View as **Graph**

Le damos a **Apply** y nos aparecerá el gráfico:



Y esto lo realizaríamos con todos los equipos en los que queramos mostrar el espacio usado.

7 CONCLUSIÓN

Una vez terminado el proyecto, se puede decir que la mayoría de los objetivos del trabajo se han cumplido perfectamente. Cuando planteamos la idea del TFG, pensábamos en algo que nos pudiese resultar útil de cara a un posible desempeño que podríamos realizar en una empresa.

Después de haberlo realizado, hemos podido observar que en cuanto a la estructura de una red de un instituto/empresa, es una gran idea utilizar OpenLDAP, debido a que es una manera muy sencilla de organizar todo. Además, para el tema de administración, hemos podido ver que es bastante intuitiva, lo cual facilita mucho las cosas.

En cuanto a la herramienta Zabbix, hemos observado que, para el tema de monitorización, es una de las mejores que existen en la actualidad, ya que es bastante completa.

8 BIBLIOGRAFÍA

Página LDAP: <https://www.openldap.org/>

Instalación Server LDAP: <http://somebooks.es/12-7-instalar-y-configurar-openldap-en-el-servidor-ubuntu/>

Instalación Cliente Ubuntu LDAP: <http://somebooks.es/12-9-configurar-un-equipo-cliente-con-ubuntu-para-autenticarse-en-el-servidor-openldap/>

Instalación Cliente Windows LDAP: <https://www.youtube.com/watch?v=6ls6kRvGkrU>

Página pGina: <http://pgina.org/>

Consulta usuarios LDAP: <https://www.linuxito.com/gnu-linux/nivel-alto/1023-consultas-a-directorios-ldap-utilizando-ldapsearch>

SSL LDAP: <https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-openldap-and-phpldapadmin-on-an-ubuntu-14-04-server#create-an-ssl-certificate>

Página samba: <https://www.samba.org/>

Instalar y configurar SAMBA: <https://www.cambiatealinux.com/instalar-y-configurar-samba-en-ubuntu-linux>

Grupos SAMBA: <https://eltallerdelbit.com/gestion-usuarios-grupos-permisos-samba/>

Fallo al obtener la lista de compartición del servidor: <https://lug-novatos.lugmen.org.narkive.com/w4m0ldRg/consulta-fallo-al-obtener-la-lista-de-comparticion-del-servidor>

Carpetas compartidas NFS: <http://somebooks.es/10-4-acceder-a-la-carpeta-compartida-con-nfs-desde-un-cliente-con-ubuntu/>

Página SSH: <https://www.ssh.com/ssh/>

SSH sin contraseña: <https://blog.desdelinux.net/ssh-sin-password-solo-3-pasos/>

Zabbix: <https://www.zabbix.com/>

Instalación Zabbix: <https://www.zabbix.com/download>