UNIVERSIDAD SAN CARLOS DE GUATEMALA

CENTRO UNIVERSITARIO DE OCCIDENTE

DIVISIÓN CIENCIAS DE LA INGENIERÍA

CARRERA DE INGENIERÍA CIENCIAS Y SISTEMAS

LABORATORIO DE SISTEMAS OPERATIVOS 1

"**SÉPTIMO SEMESTRE"**

ING.: FRANCISCO ROJAS

ESTUDIANTES:

LUIS ESTUARDO BOLAÑOS GONZÁLEZ - 201731766

MARIO MOISES RAMIREZ TOBAR - 201830007

YEFER RODRIGO MIGUEL ALVARADO TZUL - 201731163

TRABAJO: GESTIÓN DE USUARIOS (PROYECTO FINAL)

FECHA: 13 de mayo de 2,021

DESCRIPCIÓN DEL PROYECTO

Se solicita la creación de 3 servidores y 3 clientes dentro de una red local, la finalidad de cada uno se explican a continuación.

1er. Servidor:

Funcionalidad de autenticación y manejo de políticas de permisos dentro de la red (LDAP). La configuración de este servidor se realiza sin ninguna herramienta web, es configuración directa por archivos de texto.

2do. Servidor:

Configuración para dar el servicio de correo electrónico con interfaz web, todas las autenticaciones para acceder a las cuentas, así como para crear y modificar se hacen directamente con el servidor de autenticación LDAP (servidor 1). La configuración de este servidor se realiza sin ninguna herramienta web, es configuración directa por archivos de texto y solo el uso del servicio de correo se realiza por medio de webmail.

3er. Servidor:

Servidor de respaldos basado en bacula y en este mismo se hacen los respaldos localmente. Para poder acceder a la administración de este servidor se hace mediante su interfaz web y los usuarios son autenticados también por el servidor de autenticación LDAP. La programación de este servidor contempla un respaldo completo cada 5 minutos y un respaldo diferencial cada minuto (esto es para poder comprobar si se realizan correctamente), sobre las carpetas de almacenamiento del servidor web.

1er. Cliente:

Sistema operativo Windows 10 con autenticación a través del servidor de autenticaciones LDAP (los usuarios están configurados dentro del servidor 1).

2do. Cliente:

Sistema operativo Linux modo gráfico con autenticación a través del servidor de autenticaciones LDAP (los usuarios están configurados dentro del servidor 1).

3er. Cliente:

Es el equipo de administración que es el único que debe tener permisos para acceder a los tres servidores. Cualquier otro cliente no puede tener acceso a los servidores.

REQUERIMIENTOS TECNICOS

- Uso del sistema operativo Centos en modo texto para la configuración del servidor uno y tres.
- Uso del sistema operativo Centos en modo gráfico para la configuración del servidor dos.
- Creación de una red que emule una red local para la interacción entre servidores y clientes.
- Llevar un control para los permisos de comunicación entre clientes y servidores.
- Guardar respaldos de seguridad para cada uno de los usuarios dentro de la red.
- Envío de emails entre usuarios según las respectivas restricciones.

HERRAMIENTAS

- Sistema operativo CentOS 7 en modo texto.
- Sistema operativo Centos 7 en modo gráfico.
- Sistema operativo Windows 10.
- Herramientas de respaldo Bacula.
- Paquete net - tools.
- Editor de texto nano.
- Scripts en lenguaje Bash.
- Logmein Hamachi.
- nano

OBJETIVOS

Objetivos Generales:

- Aplicar los conocimientos sobre seguridad, servicios y respaldos hacia sistemas operativos.
- Implementar una red capaz de comunicarse entre los nodos que la conformen.
- Estudiar diferentes softwares útiles para el manejo del dominio de sistemas operativos.

Objetivos Específicos:

- Montar una red para la comunicación entre clientes y servidores.
- Lograr un buen manejo en los permisos de comunicación entre clientes y servidores.
- Aplicar conocimientos sobre Bacula para la creación del servidor de respaldos.
- Aplicar conocimientos sobre LDAP para la creación del servidor de autenticación.
- Correcto envío de emails entre usuarios.
- Generar respaldos y restauraciones de manera correcta de los datos de los usuarios.

MARCO TEÓRICO

Servidor

Un servidor es un conjunto de computadoras capaz de atender las peticiones de un cliente y devolverle una respuesta en concordancia. Los servidores se pueden ejecutar en cualquier tipo de computadora, incluso en computadoras dedicadas a las cuales se les conoce individualmente como «el servidor». En la mayoría de los casos una misma computadora puede proveer múltiples servicios y tener varios servidores en funcionamiento. La ventaja de montar un servidor en computadoras dedicadas es la seguridad. Por esta razón la mayoría de los servidores son procesos diseñados de forma que puedan funcionar en computadoras de propósito específico.

Los servidores operan a través de una arquitectura "cliente-servidor". Los servidores son programas de computadora en ejecución que atienden las peticiones de otros programas: los clientes. Por tanto, el servidor realiza otras tareas para beneficio de los clientes; les ofrece la posibilidad de compartir datos, información y recursos de hardware y software. Los clientes usualmente se conectan al servidor a través de la red, pero también pueden acceder a él a través de la computadora donde está funcionando. En el contexto de redes Internet Protocol (IP), un servidor este es un programa que opera como oyente de un socket.

Comúnmente, los servidores proveen servicios esenciales dentro de una red, ya sea para usuarios privados dentro de una organización o compañía, o para usuarios públicos a través de Internet. Los tipos de servidores más comunes son servidor de base de datos, servidor de archivos, servidor de correo, servidor de impresión, servidor web, servidor de juego, y servidor de aplicaciones.

Un gran número de sistemas usa el modelo de red cliente-servidor, entre ellos los sitios web y los servicios de correo. Un modelo alternativo, el modelo red peer-to-peer, permite a todas las computadoras conectadas actuar como clientes o servidores.

Cliente-Servidor

La arquitectura cliente-servidor es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

Algunos ejemplos de aplicaciones que usan el modelo cliente-servidor son el Correo electrónico, un Servidor de impresión y la World Wide Web. La arquitectura cliente-servidor es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

Algunos ejemplos de aplicaciones que usan el modelo cliente-servidor son el Correo electrónico, un Servidor de impresión y la World Wide Web.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la

centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un solo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

Una disposición muy común son los *sistemas multicapa* en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

La red cliente-servidor es una red de comunicaciones en la cual los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados.Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc. Este tipo de red puede utilizarse conjuntamente en caso de que se esté utilizando en una red mixta.

Máquinas Virtualizadas (KVM)

kernel-based Virtual Machine o KVM, (en español, Máquina virtual basada en el núcleo) es una solución para implementar virtualización completa con Linux. Está formada por un módulo del núcleo (con el nombre kvm.ko) y herramientas en el espacio de usuario, siendo en su totalidad software libre. El componente KVM para el núcleo está incluido en Linux desde la versión 2.6.20.

KVM permite ejecutar máquinas virtuales utilizando imágenes de disco que contienen sistemas operativos sin modificar. Cada máquina virtual tiene su propio hardware virtualizado: una tarjeta de red, discos duros, tarjeta gráfica, etc.

KVM fue creado por Qumranet. En 2008 esta empresa fue adquirida por Red Hat Inc y actualmente el software es mantenido por Openshift.

LDAP

El protocolo ligero de acceso a directorios (en inglés: Lightweight Directory Access Protocol, también conocido por sus siglas de LDAP) hace referencia a un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Un directorio es un conjunto de objetos con atributos organizados de una manera lógica y jerárquica. El ejemplo más común es el directorio telefónico, que consiste en una serie de nombres (personas u organizaciones) que están ordenados alfabéticamente, con cada nombre teniendo una dirección y un número de teléfono adjuntos. Para entender mejor, es un libro o

carpeta, en la cual se escriben nombres de personas, teléfonos y direcciones, y se ordena alfabéticamente.

Un árbol de directorio LDAP a veces refleja varios límites políticos, geográficos u organizacionales, dependiendo del modelo elegido. Los despliegues actuales de LDAP tienden a usar nombres de Sistema de Nombres de Dominio (DNS por sus siglas en inglés) para estructurar los niveles más altos de la jerarquía. Conforme se desciende en el directorio pueden aparecer entradas que representan personas, unidades organizacionales, impresoras, documentos, grupos de personas o cualquier cosa que representa una entrada dada en el árbol (o múltiples entradas).

Habitualmente, almacena la información de autenticación (usuario y contraseña) y es utilizado para autenticarse aunque es posible almacenar otra información (datos de contacto del usuario, ubicación de diversos recursos de la red, permisos, certificados, etc). A manera de síntesis, LDAP es un protocolo de acceso unificado a un conjunto de información sobre una red.

WebMail

Un **correo web** es un cliente de correo electrónico, que provee una interfaz web que permite crear cuentas de e-mail que pueden ser revisadas a través de la web. Este servicio lo ofrecen muchos sitios web, en especial los portales y también los proveedores de acceso a internet (ISPs). Otras formas de acceder al correo electrónico pueden ser:

- Conectándose con un cliente de correo local a un servidor de correo remoto utilizando un protocolo *ad hoc* de transporte de correo, como IMAP o POP, descargar los correos y almacenarlos localmente.
- Utilizando un cliente de correo por consola (por ejemplo, Mutt).

El webmail permite listar, desplegar y borrar mediante un navegador web los correos almacenados en un servidor remoto. Los correos pueden consultarse posteriormente desde otro ordenador conectado a la misma red (por ejemplo, Internet) y que disponga de un navegador web.

Generalmente, también permite la redacción y el envío de correos, y no está limitado a la lectura de correo electrónico.

Roundcube

Roundcube es un cliente de correo para ver los mensajes de correo (correo electrónico) a través de una página web, pudiendo acceder desde cualquier navegador con acceso a internet. Desde él es posible realizar todas las operaciones necesarias para gestionar los correos e incluso usarlo como agenda de contactos y calendario.

Roundcube está liberado bajo la licencia GPL, por lo que es software libre.

Principales Características

- Disponible en 65 idiomas.
- Drag-&-drop para gestionar los *e-mails*.
- Soporte completo para mensajes MIME y HTML.
- Sistema de protección de la privacidad sofisticado.
- Redactar mensajes con archivos adjuntos.
- Múltiples identidades del remitente.
- Composición de *e-mails* en HTML enriquecido.
- Reenviar mensajes con archivos adjuntos.

- Búsqueda de mensajes y contactos.
- Corrección ortográfica.
- Administración de carpetas IMAP.
- Soporte para servidores SMTP externos.
- Caché acceso al buzón rápido.
- Número ilimitado de usuarios y mensajes.
- Plantilla sistema de skins personalizados.

Bacula

Bacula es una colección de herramientas de respaldo capaz de cubrir las necesidades de respaldo de equipos bajo redes IP. Se basa en una arquitectura Cliente-servidor que resulta eficaz y fácil de manejar, dada la amplia gama de funciones y características que brinda; copiar y restaurar ficheros dañados o perdidos. Además, debido a su desarrollo y estructura modular, Bacula se adapta tanto al uso personal como profesional, desde un equipo hasta grandes parques de servidores.

Componente

Los componentes de Bacula: generalmente usado en sistemas u organizaciones donde la información es ingresada desde un dispositivo o punto final de red (PC de escritorio), transporta parte de sus datos a un servidor directamente desde la dirección IP. Todo el conjunto de elementos que forman Bacula trabaja en sincronía y es totalmente compatible con bases de datos como MySQL, SQLite y PostgreSQL.

Bacula-director daemon

Es el demonio que gestiona la lógica de los procesos de backup y los demás servicios. El servidor de la base de datos debe estar accesible desde la máquina que ejecuta este demonio (o también puede estar en la misma máquina y escuchar en localhost).

En el archivo de configuración de este demonio se especifica dónde y cómo acceder al resto de demonios y recursos, la contraseña para el acceso mediante bacula-console y los trabajos o jobs.

Bacula-storage daemon

Este demonio es el encargado de manejar los dispositivos de almacenamiento; esto exige que este demonio esté instalado en la máquina que posea la conexión física a los dispositivos de almacenamiento, tales como discos locales, grabadoras de CD o DVD, unidades de cinta, volúmenes NAS o SAN, autocargadores o librerías de cinta.

Bacula-file daemon

Mediante este demonio, Bacula obtiene los ficheros que necesita respaldar. Así pues, este es el componente que hay que instalar en las máquinas que necesiten respaldo. Realiza la misma función que los "agentes" en otros sistemas de backup.

Este archivo de configuración es el más simple de todos: simplemente especifica qué directores pueden realizar peticiones.

Para poder interactuar con el servicio de backup es necesario un cliente.

Bacula-console

Es un programa que permite comunicarse con bacula-director (mientras esté funcionando). Tiene dos versiones, una estilo terminal y otra estilo interfaz gráfica para Gnome. Permite realizar consultas y disparar tareas, por ejemplo.

Hamachi

Es una aplicación comercial que configura redes privadas virtuales capaz de establecer vínculos directos entre computadoras que están bajo firewalls de NAT sin necesitar reconfiguración alguna (en la mayoría de los casos). En otras palabras, establece una conexión a través de Internet y simula una red de área local formada por computadoras remotas. Actualmente está disponible la versión para Microsoft Windows y la versión beta para Mac OS X y Linux. El 8 de agosto de 2006 se anunció que Hamachi era adquirida por LogMeIn.

Cómo funciona

Hamachi es un sistema VPN de administración centralizada que consiste en un clúster servidor administrado por el vendedor del sistema y el software cliente, el cual es instalado en los ordenadores de los usuarios.

El software cliente agrega una interfaz de red virtual al ordenador que es utilizada tanto para interceptar el tráfico VPN saliente como para inyectar el tráfico VPN entrante. El tráfico saliente enviado por el sistema operativo a esta interfaz es entregado al software cliente, que lo cifra y lo autentifica y luego lo envía al nodo VPN de destino a través de una conexión UDP iniciada a tal efecto. Hamachi se encarga del tunelamiento del tráfico IP, incluido el broadcast

(difusión) y el multicast (multidifusión). La versión Windows reconoce y tunela, además, el tráfico IPX.

Cada cliente establece y mantiene una conexión de control con el Cluster servidor. Cuando la conexión está establecida, el cliente entra en una secuencia de identificación de usuario, seguida de un proceso de descubrimiento y sincronización de estado. El paso de autentificación de usuario autentifica al cliente contra el servidor y viceversa. El descubrimiento es utilizado para determinar la topología de la conexión a Internet del cliente, y más concretamente para detectar la presencia de dispositivos cortafuegos y servidores NAT. El paso de sincronización extrae una vista del cliente de sus redes privadas sincronizadas con los otros miembros de esas redes.

Cuando un miembro de una red se conecta o se desconecta, el servidor da instrucciones a los otros nodos de la red para que inicien o detengan túneles con dicho miembro. Cuando se establecen túneles entre los nodos, Hamachi utiliza una técnica de NAT transversal asistido por servidor, similar al "UDP hole punching" ("perforadora de agujeros UDP"). Hasta la fecha, no se ha publicado información detallada de cómo funciona realmente. El vendedor afirma que "...atraviesa con éxito las conexiones P2P en el 95% de los casos, aproximadamente..." Este proceso no funciona en ciertas combinaciones de dispositivos NAT, que requieren que el usuario abra un puerto para la conexión. Además de esto, la versión 1.0 del software cliente es capaz de retransmitir el tráfico a través de los 'servidores de retransmisión' que mantiene el vendedor. En octubre de 2014, la aplicación comenzó a tener problemas graves, dado que mucha gente abandonó hamachi para abrir puertos inalámbricos de otras maneras alternativas. En el caso de que se pierda la conexión con el servidor de manera inesperada, el cliente mantiene todos sus túneles e inicia una comprobación de sus estados. Cuando el servidor pierde una conexión de

cliente de manera inesperada, se informa a los nodos clientes sobre el hecho y se espera a que inicien sus comprobaciones. Todo esto hace inmune a los túneles Hamachi frente a problemas de red transitorios en el camino entre el cliente y el servidor, y de igual modo quedan operativos en los breves intervalos de indisponibilidad completa del servidor.

El bloque de direcciones 5.0.0.0 está reservado por la IANA y no está actualmente en uso en el dominio de encaminamiento de Internet, pero no está garantizado que esto continúe así en el futuro. Se espera que el fondo común se agotará en abril de 2090. Si este rango es asignado, los usuarios de Hamachi no podrán conectarse a ninguna dirección IP de Internet dentro de ese rango mientras estén utilizando el cliente Hamachi.

Además, utilizar un prefijo de red crea un único dominio de difusión entre todos los clientes. Esto hace posible la utilización de protocolos que dependen de la difusión IP para descubrir y anunciar servicios sobre las redes Hamachi.

Hamachi es habitualmente utilizada para jugar en red y para la administración remota. El vendedor provee servicios básicos gratis y otras características extra pagando.

Direccionamiento

A cada cliente Hamachi se le asigna una dirección IP desde el bloque de direcciones 5.0.0.0/8 cuando inicia una sesión en el sistema por primera vez, y es en adelante asociada con la clave de cifrado pública del cliente. Mientras el cliente retenga esta clave, puede autentificarse en el sistema y utilizar esa dirección IP 5.X.X.X.

DESARROLLO

Configuración Servidor LDAP

Instalacion y Configuracion de Hamachi en Centos 7

Actualizar los repositorios del sistema operativo.

```
# yum update
```

Instalamos los paquetes necesarios para la instalación

```
# sudo yum install wget
```

Descargar los paquetes de hamachi

```
# sudo wget http://www.vpn.net/installers/logmein-hamachi-2.1.0.203-1.x86_64.rpm
# sudo yum install logmein-hamachi-2.1.0.203-1.x86_64.rpm
```

Iniciar sesión con el servidor hamachi

```
# Hamachi login
```

Ingresamos el nombre del cliente para hamachi, en este caso se llamará “servidor-1”

```
# sudo hamachi set-nick servidor-1
```

Vincular el correo registrado en Logmein

```
# sudo hamachi attach nama@gmail.com
```

Ingresamos el ID de la red hamachi, nos pedirá un password este será la contraseña de la máquina virtual.

```
# sudo hamachi join 451-760-249
```

Instalacion y Activacion de OpenLDAP

En esta ocasión el nombre del dominio ldap se llama “apex”, por esto antes de iniciar la instalación es necesario crear una archivo hosts para que se guarda la información de la conexión.

Creamos un archivo con la siguiente ruta

```
# /etc/hosts
```

Dentro del archivo colocamos el siguiente texto

```
ipVirtualServidor server.dominio.com server

# 25.16.221.130 server.apex.com server
```

Instalación de los paquetes de OpenLDAP

```
# yum -y install openldap compat-openldap openldap-clients
openldap-servers openldap-servers-sql openldap-devel net-tools nano
```

Iniciamos el Servicio slapd y lo activamos

```
# systemctl start slapd
# systemctl enable slapd
```

Verificación el LDAP

```
tcp        0      0 0.0.0.0:389             0.0.0.0:*               LISTEN      1520/slapd
tcp6       0      0 :::389                  :::*                    LISTEN      1520/slapd
```

Configurar la Contraseña de administrador LDAP

Reemplazar ldppassword con su contraseña

```
# slappasswd -h {SSHA} -s ldppassword
```

El comando anterior generará un hash cifrado de la contraseña ingresada que debe usar en el archivo de configuración LDAP. Así que tome nota de esto y déjelo a un lado.

Ejemplo de la salida:

```
{SSHA} d / thexcQUuSfe3rx3gRaEhHpNJ52N8D3
```

Configurar el Servidor OpenLDAP

Cree un archivo .ldif

```
# nano db.ldif
```

Reemplace la contraseña encriptada ({SSHA} d / thexcQUuSfe3rx3gRaEhHpNJ52N8D3) con la contraseña que generó en el paso anterior.

```
dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=apex,dc=com

dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=admin,dc=apex,dc=com

dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcRootPW
olcRootPW: {SSHA}d/thexcQUuSfe3rx3gRaEhHpNJ52N8D3
```

Una vez que haya terminado con el archivo ldif, envíe la configuración al servidor LDAP.

```
# ldapmodify -Y EXTERNAL  -H ldapi:/// -f db.ldif
# ldapmodify -Y EXTERNAL  -H ldapi:/// -f permiss-modify.ldif
```

Realice cambios en el archivo /etc/openldap/slapd.d/cn=config/olcDatabase={1}monitor.ldif (No editar manualmente) para restringir el acceso del monitor solo al usuario raíz de ldap (ldapadm), no a otros.

```
# nano monitor.ldif
```

Utilice la siguiente información.

```
dn: olcDatabase={1}monitor,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by
dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external, cn=auth"
read by dn.base="cn=admin,dc=apex,dc=com" read by * none
```

Una vez que haya actualizado el archivo, envíe la configuración al servidor LDAP.

```
# ldapmodify -Y EXTERNAL  -H ldapi:/// -f monitor.ldif
```

Configurar la base de datos LDAP

Copie el archivo de configuración de la base de datos de muestra /var/lib/ldap y actualice los permisos del archivo.

```
# cp /usr/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG
```

Agregue los esquemas LDAP coseno y nis.

```
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
```

Generamos el archivo base.ldif

```
# nano base.ldif
```

Utilice la siguiente información. Puede modificarlo según sus necesidades.

```
dn: dc=apex,dc=com
dc: apex
objectClass: top
objectClass: domain

dn: cn=admin ,dc=apex,dc=com
objectClass: organizationalRole
cn: admin
description: LDAP Manager

dn: ou=People,dc=apex,dc=com
objectClass: organizationalUnit
ou: People

dn: ou=Group,dc=apex,dc=com
objectClass: organizationalUnit
ou: Group
```

Construye la estructura del directorio.

```
# ldapadd -x -W -D "cn=admin,dc=apex,dc=com" -f base.ldif
```

El comando ldapadd le pedirá la contraseña de ldapadm (usuario raíz de LDAP).

Salida:

```
Enter LDAP Password:
adding new entry "dc=apex,dc=local"

adding new entry "cn=admin ,dc=apex,dc=com"

adding new entry "ou=People,dc=apex,dc=com"

adding new entry "ou=Group,dc=apex,dc=com"
```

Crear usuario LDAP

En lugar de crear un nuevo usuario, puede migrar los usuarios locales a LDAP . Creemos un archivo LDIF para un nuevo usuario llamado juan.

```
# nano jaun.ldif
```

Pegue las siguientes líneas en el archivo LDIF anterior.

```
dn: uid=juan,ou=People,dc=apex,dc=com
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
cn: juan
uid: juan
uidNumber: 9999
gidNumber: 100
homeDirectory: /home/juan
loginShell: /bin/bash
userPassword: {crypt}x
shadowLastChange: 17058
shadowMin: 0
shadowMax: 99999
shadowWarning: 7
```

Utilice el comando ldapadd con el archivo anterior para crear un nuevo usuario llamado "juan" en el directorio OpenLDAP.

```
# ldapadd -x -W -D "cn=admin,dc=apex,dc=com" -f juan.ldif
```

Salida: - Ingrese la contraseña de admin.

```
Enter LDAP Password:
adding new entry "uid=juan,ou=People,dc=apex,dc=com"
```

Asignar una contraseña al usuario.

```
# ldappasswd -s password123 -W -D "cn=admin,dc=apex,dc=com" -x
"uid=juan,ou=People,dc=apex,dc=com"
```

Firewall

Agregue el servicio LDAP al firewall (TCP 389).

```
firewall-cmd --permanent --add-service=ldap
firewall-cmd --reload
```

Habilitar el registro LDAP

Configure Rsyslog para registrar eventos LDAP en el archivo de registro /var/log/ldap.log

```
# nano /etc/rsyslog.conf
```

Agregue la siguiente línea al archivo /etc/rsyslog.conf.

```
# local4.* /var/log/ldap.log
```

Reinicie el servicio rsyslog.

```
# systemctl restart rsyslog
```

Configuración del cliente LDAP para utilizar el servidor LDAP

Instale los paquetes de cliente LDAP necesarios en la máquina cliente.

```
yum install -y openldap-clients nss-pam-ldapd net-tools
```

Ejecute el siguiente comando para agregar la máquina cliente al servidor LDAP para el inicio de sesión único. Reemplace "192.168.1.10" con la dirección IP o el nombre de host de su servidor LDAP.

```
authconfig --enableldap --enableldapauth --ldapserver=192.168.1.10
--ldapbasedn="dc=apex,dc=com" --enablemkhomedir --update
```

Reinicie el servicio de cliente LDAP.

```
# systemctl restart nslcd
```

Verificar inicio de sesión LDAP

Utilice el comando getent para obtener las entradas LDAP del servidor LDAP.

```
# getent passwd juan
```

salida:

```
juan:x:9999:100:/home/juan:/bin/bash
```

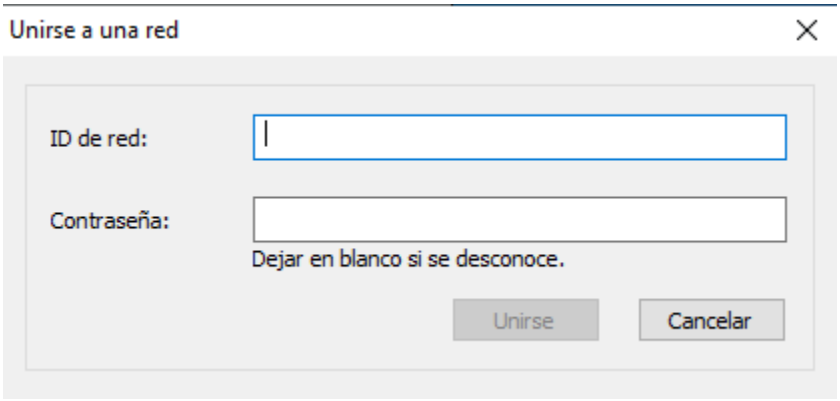
Instalacion Logmein hamachi en Windows 10

Debemos de iniciar sesión en Logmein, posteriormente nos dirigimos a la sección de red. Y seleccionamos la opción de agregar cliente.

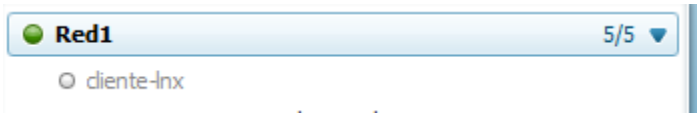
Posteriormente seleccionamos instalar LogMein Hamachi en este ordenar, automáticamente descargar el archivo .exe

Después de la instalación procederemos a unirnos con la red existente.

Ingresamos el ID de la red y en caso de que tuviera contraseña la colocamos, pero si no se debe dejar en blanco.

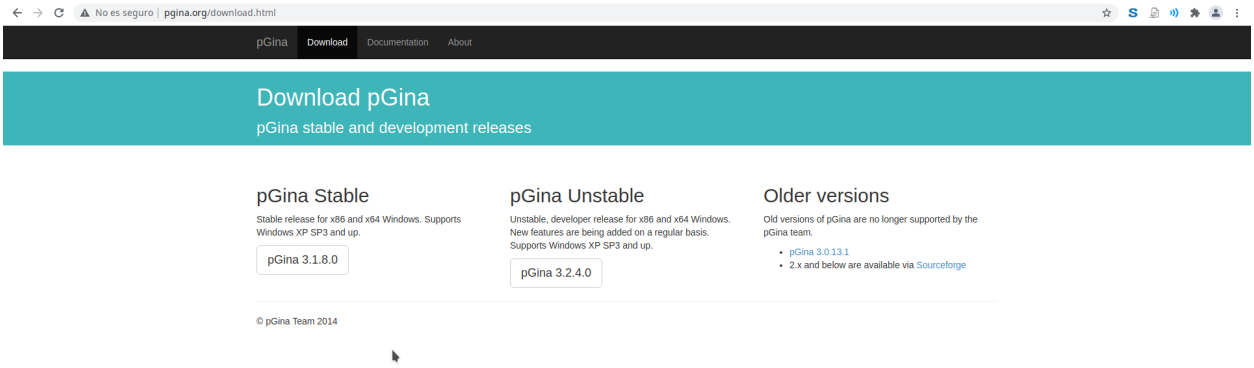


Una vez hecho esto nos aparecerá la conexión activa.



Instalación de PGina Windows 10

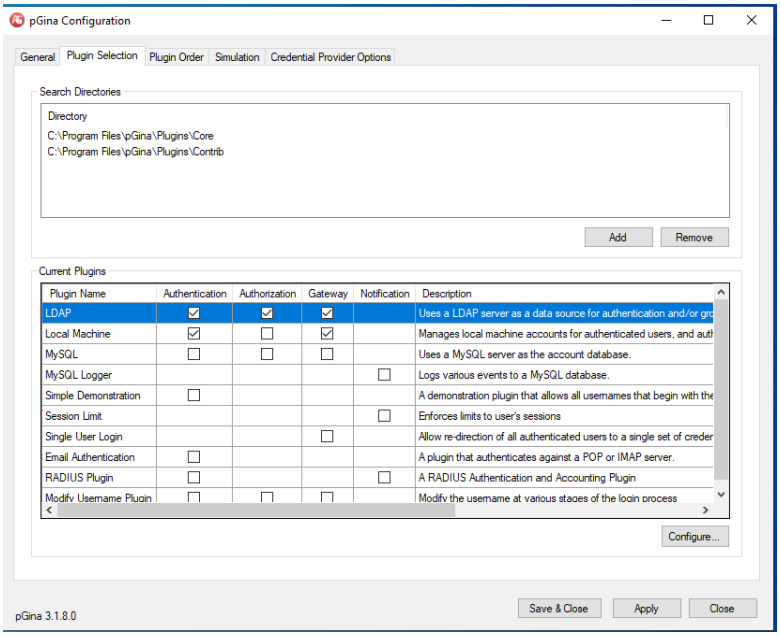
En el navegador buscamos Pgina y accedemos a la primera opción de download. Y descargamos la versión pGina Stable, este descargara el archivo .exe



Posteriormente instalamos el archivo.

Una vez instalado abrimos como administrador el programa, y procederemos a la configuración.

Activamos la opción Authentication, Authorization y Gateway, de LDAP y aplicamos los cambios, posteriormente nos dirigimos a la opción de Configure...



LDAP Host(s): <ipServer>

Search DN: cn=admin,dc=<dominio>,dc=com

Search Password: <contraseña del ldap admin>

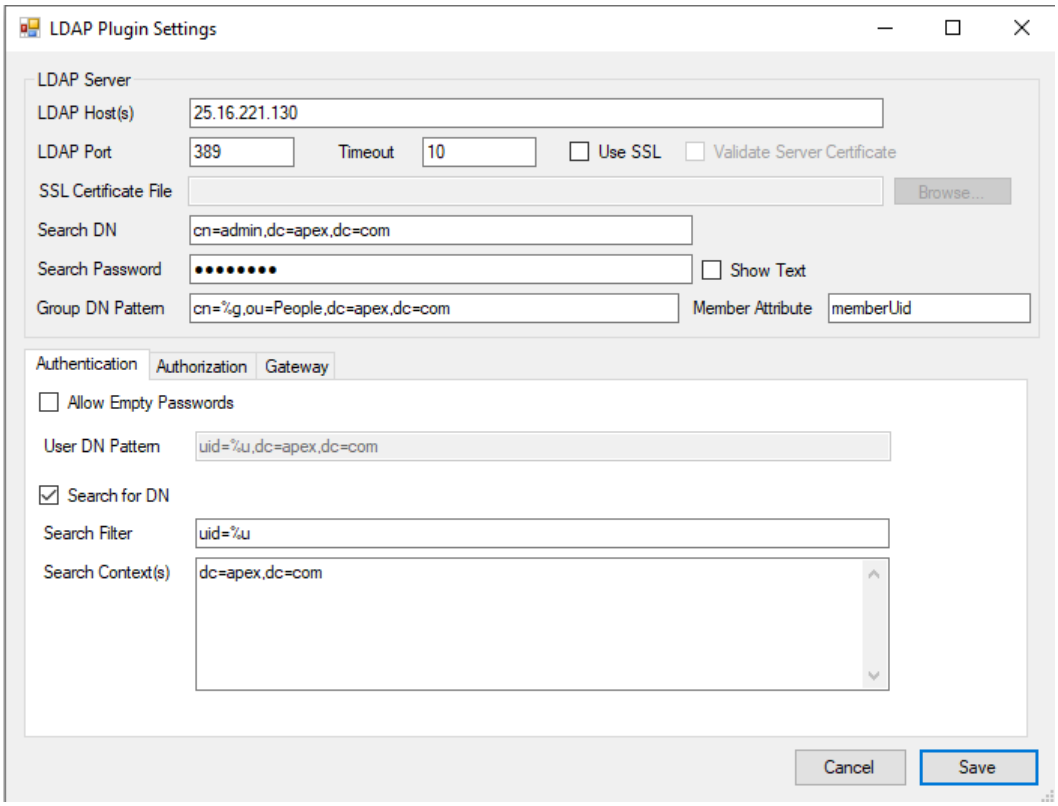
Group DN Pattern: cn=%g,ou=People,dc=<dominio>,dc=com

Sección Authentication

Seleccionamos Search for DN

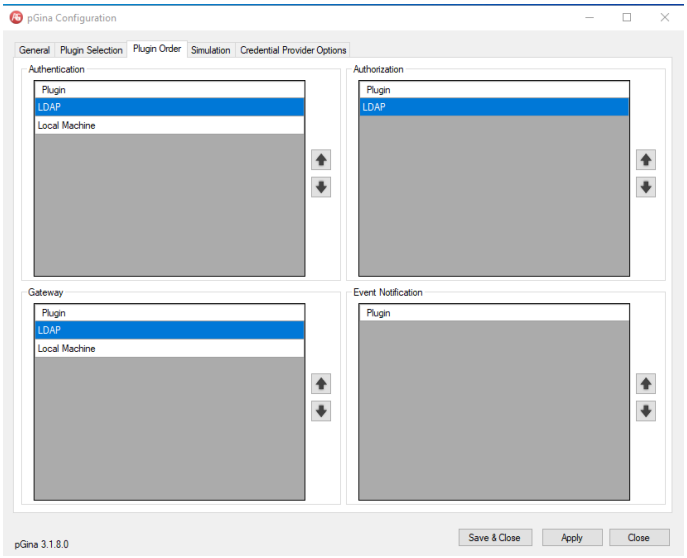
Search Filter: uid=%u

Search Context(s): dc=<dominio>,dc=com



Al finalizar la configuración guardamos los cambio

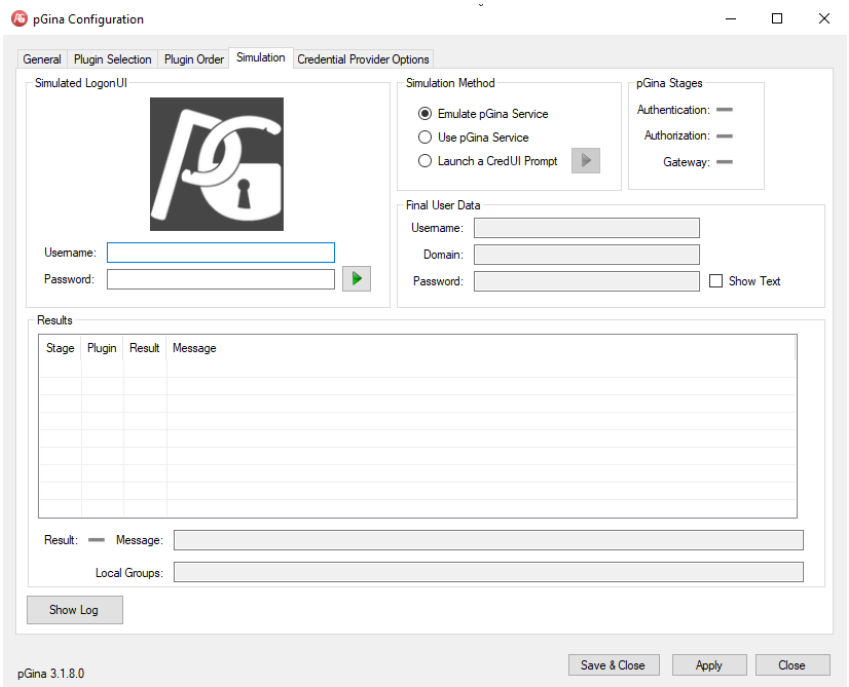
Nos dirigimos a la pestaña Plugin Order. Y ordenamos los LDAP de manera que queden de primero



Al finalizar esta configuración aplicamos los cambios.

Por último nos dirigimos a la pestaña Simulation

En esta sección ingresamos un usuario y contraseña, posteriormente se hace click en play, y si en el recuadro con el nombre “pina Stages” aparecen tres check verdes, significa que todo está correcto y podemos guardar y salir. El inicio de sesión estará funcionando correctamente.



Creación del Servidor WebMail

Instalación Inicial

Actualizar repositorios

```
#actualizar repositorios
yum update -y
```

Instalar paquetes

```
yum install httpd httpd-tools mariadb-server mariadb php php-fpm
php-mysqlnd php-opcache php-gd php-xml php-mbstring php-json php-intl
php-ldap
```

```
yum install
https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rp
m
yum update && yum install epel-release
yum install http://rpms.remirepo.net/enterprise/remi-release-7.rpm
yum install yum-utils
```

Configuración Inicial

```
echo "Ingrese la versión de su php Ejemplo si es 7.4 o 7.4.1 ingrese
74\n"
php -v
echo "\n"
```

```
read version_php
yum-config-manager --enable remi-php$version_php
yum install php-opcache
```

```
systemctl start httpd
```

```
systemctl start mariadb
```

```
systemctl enable httpd
```

```
systemctl enable mariadb
```

```
firewall-cmd --permanent --add-service=http
```

```
firewall-cmd --reload
```

Configurar la base de datos

```
mysql_secure_installation
```

Escribir en el bash de mariadb, los siguientes comandos:

```
create database roundcubedb;
create user roundcubeuser@localhost identified by 'roundcubepwd';
grant all on roundcubedb.* to roundcubeuser@localhost;
flush privileges;
exit;
```

Configuración RoundCube

```
cd /var/www/html/
```

```
wget
https://github.com/roundcube/roundcubemail/releases/download/1.4.1/ro
undcubemail-1.4.1-complete.tar.gz
tar -xvf roundcubemail-1.4.1-complete.tar.gz
```

```
mv roundcubemail-1.4.1 webmail
```

```
cp config.inc.php /var/www/html/webmail/config/config.inic.php
```

```
chown -R apache:apache /var/www/
chmod -R 755 /var/www/html/webmail
systemctl restart httpd
```


```
chown -R apache:apache /var/www/
chmod -R 755 /var/www/html/webmail
systemctl restart httpd
```


Configurar el archivo config-inc.php

```
<?php

/* Local configuration for Roundcube Webmail */

// ----------------------------------
// SQL DATABASE
// ----------------------------------
// Database connection string (DSN) for read+write operations
// Format (compatible with PEAR MDB2):
db_provider://user:password@host/database
// Currently supported db_providers: mysql, pgsql, sqlite, mssql,
sqlsrv, oracle
// For examples see
http://pear.php.net/manual/en/package.database.mdb2.intro-dsn.php
// Note: for SQLite use absolute path (Linux):
'sqlite:////full/path/to/sqlite.db?mode=0646'
//    or (Windows): 'sqlite:///C:/full/path/to/sqlite.db'
// Note: Various drivers support various additional arguments for
connection,
```

```
//    for Mysql: key, cipher, cert, capath, ca, verify_server_cert,
//    for Postgres: application_name, sslmode, sslcert, sslkey,
sslrootcert, sslcrl, sslcompression, service.
//    e.g.
'mysql://roundcube:@localhost/roundcubemail?verify_server_cert=false'
$config['db_dsnw'] =
'mysql://roundcubeuser:roundcubepwd@localhost/roundcubedb';

// The IMAP host chosen to perform the log-in.
// Leave blank to show a textbox at login, give a list of hosts
// to display a pulldown menu or set one host as string.
// To use SSL/TLS connection, enter hostname with prefix ssl:// or
tls://
// Supported replacement variables:
// %n - hostname ($_SERVER['SERVER_NAME'])
// %t - hostname without the first part
// %d - domain (http hostname $_SERVER['HTTP_HOST'] without the first
part)
// %s - domain name after the '@' from e-mail address provided at
login screen
// For example %n = mail.domain.tld, %t = domain.tld
$config['default_host'] = '25.7.96.199';

// SMTP port (default is 25; use 587 for STARTTLS or 465 for the
// deprecated SSL over SMTP (aka SMTPS))
$config['smtp_port'] = 25;

// SMTP username (if required) if you use %u as the username
Roundcube
// will use the current username for login
$config['smtp_user'] = '';

// SMTP password (if required) if you use %p as the password
Roundcube
// will use the current user's password for login
$config['smtp_pass'] = '';

// provide an URL where a user can get support for this Roundcube
installation
```

```
// PLEASE DO NOT LINK TO THE ROUNDCUBE.NET WEBSITE HERE!
$config['support_url'] = '';

// this key is used to encrypt the users imap password which is
stored
// in the session record (and the client cookie if remember password
is enabled).
// please provide a string of exactly 24 chars.
// YOUR KEY MUST BE DIFFERENT THAN THE SAMPLE VALUE FOR SECURITY
REASONS
$config['des_key'] = 'fpYV6KROVO1z1G5JJnpUVeCX';

// This domain will be used to form e-mail addresses of new users
// Specify an array with 'host' => 'domain' values to support
multiple hosts
// Supported replacement variables:
// %h - user's IMAP hostname
// %n - http hostname ($_SERVER['SERVER_NAME'])
// %d - domain (http hostname without the first part)
// %z - IMAP domain (IMAP hostname without the first part)
// For example %n = mail.domain.tld, %t = domain.tld
$config['mail_domain'] = '%d';

// List of active plugins (in plugins/ directory)
$config['plugins'] = array('archive', 'zipdownload');

// $config['imap_auth_type'] = 'plain';
# $config['imap_auth_type'] = 'PLAIN';
$config['enable_installer'] = true;
```

Iniciar Servidor

```
sudo yum install dovecot -y
```

```
sudo systemctl enable dovecot
```

```
sudo systemctl start dovecot
```

```
setenforce 0
```

```
systemctl restart httpd
systemctl restart dovecot
sudo service dovecot restart
```

Configurar el archivo config-inc.php.dist

```
....
// LDAP, LDAP_SIMPLE and LDAP_EXOP Driver options
// ----------------------------------
// LDAP server name to connect to.
// You can provide one or several hosts in an array in which case the hosts are
tried from left to right.
// Exemple: array('ldap1.exemple.com', 'ldap2.exemple.com');
// Default: 'localhost'
$config['password_ldap_host'] = '25.16.221.130';

// LDAP server port to connect to
// Default: '389'
$config['password_ldap_port'] = '89';

// TLS is started after connecting
// Using TLS for password modification is recommended.
// Default: false
$config['password_ldap_starttls'] = false;

// LDAP version
// Default: '3'
$config['password_ldap_version'] = '3';

// LDAP base name (root directory)
// Exemple: 'dc=example,dc=com'
$config['password_ldap_basedn'] = 'dc=apex,dc=com';

// LDAP connection method
```

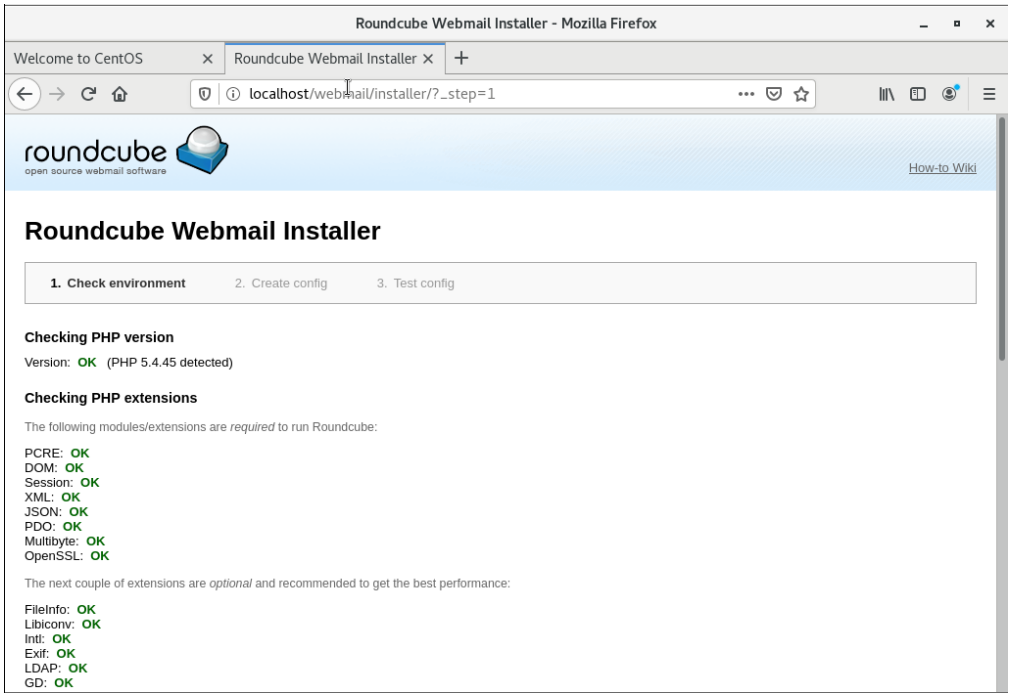
```
// There are two connection methods for changing a user's LDAP password.
// 'user': use user credential (recommended, require password_confirm_current=true)
// 'admin': use admin credential (this mode require password_ldap_adminDN and
password_ldap_adminPW)
// Default: 'user'
$config['password_ldap_method'] = 'user';

// LDAP Admin DN
// Used only in admin connection mode
// Default: null
$config['password_ldap_adminDN'] = null;

// LDAP Admin Password
// Used only in admin connection mode
// Default: null
$config['password_ldap_adminPW'] = null;

// LDAP user DN mask
// The user's DN is mandatory and as we only have his login,
// we need to re-create his DN using a mask
// '%login' will be replaced by the current roundcube user's login
// '%name' will be replaced by the current roundcube user's name part
// '%domain' will be replaced by the current roundcube user's domain part
// '%dc' will be replaced by domain name hierarchal string e.g.
"dc=test,dc=domain,dc=com"
// Exemple: 'uid=%login,ou=people,dc=exemple,dc=com'
$config['password_ldap_userDN_mask'] = 'uid=%login,ou=People,dc=apex,dc=com';

// LDAP search DN
// The DN roundcube should bind with to find out user's DN
// based on his login. Note that you should comment out the default
// password_ldap_userDN_mask setting for this to take effect.
// Use this if you cannot specify a general template for user DN with
// password_ldap_userDN_mask. You need to perform a search based on
// users login to find his DN instead. A common reason might be that
// your users are placed under different ou's like engineering or
// sales which cannot be derived from their login only.
$config['password_ldap_searchDN'] = 'cn=roundcube,ou=services,dc=apex,dc=com';

// LDAP search password
// If password_ldap_searchDN is set, the password to use for
// binding to search for user's DN. Note that you should comment out the default
// password_ldap_userDN_mask setting for this to take effect.
// Warning: Be sure to set approperiate permissions on this file so this password
// is only accesible to roundcube and don't forget to restrict roundcube's access
to
// your directory as much as possible using ACLs. Should this password be
```

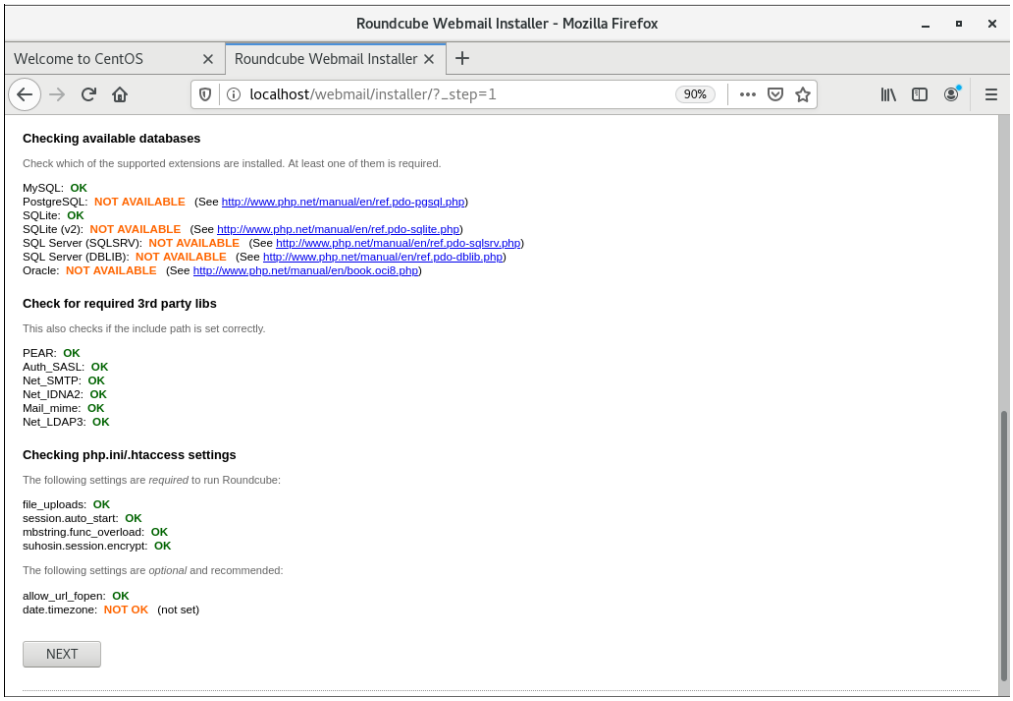
```
compromised
// you want to minimize the damage.
$config['password_ldap_searchPW'] = 'secret';

// LDAP search base
// If password_ldap_searchDN is set, the base to search in using the filter below.
// Note that you should comment out the default password_ldap_userDN_mask setting
// for this to take effect.
$config['password_ldap_search_base'] = 'ou=People,dc=apex,dc=com';

// LDAP search filter
// If password_ldap_searchDN is set, the filter to use when
// searching for user's DN. Note that you should comment out the default
// password_ldap_userDN_mask setting for this to take effect.
// '%login' will be replaced by the current roundcube user's login
// '%name' will be replaced by the current roundcube user's name part
// '%domain' will be replaced by the current roundcube user's domain part
// '%dc' will be replaced by domain name hierarchal string e.g.
"dc=test,dc=domain,dc=com"
// Example: '(uid=%login)'
// Example: '(&(objectClass=posixAccount)(uid=%login))'
$config['password_ldap_search_filter'] = '(uid=%login)';

// LDAP password hash type
// Standard LDAP encryption type which must be one of: crypt,
// ext_des, md5crypt, blowfish, md5, sha, smd5, ssha, ad, cram-md5 (dovecot style)
or clear.
// Set to 'default' if you want to use method specified in password_algorithm
option above.
// Multiple password Values can be generated by concatenating encodings with a +.
E.g. 'cram-md5+crypt'
// Default: 'crypt'.
$config['password_ldap_encodage'] = 'crypt';

// LDAP password attribute
// Name of the ldap's attribute used for storing user password
// Default: 'userPassword'
$config['password_ldap_pwattr'] = 'userPassword';

// LDAP password force replace
// Force LDAP replace in cases where ACL allows only replace not read
// See
http://pear.php.net/package/Net_LDAP2/docs/latest/Net_LDAP2/Net_LDAP2_Entry.html#me
thodreplace
// Default: true
$config['password_ldap_force_replace'] = true;
```

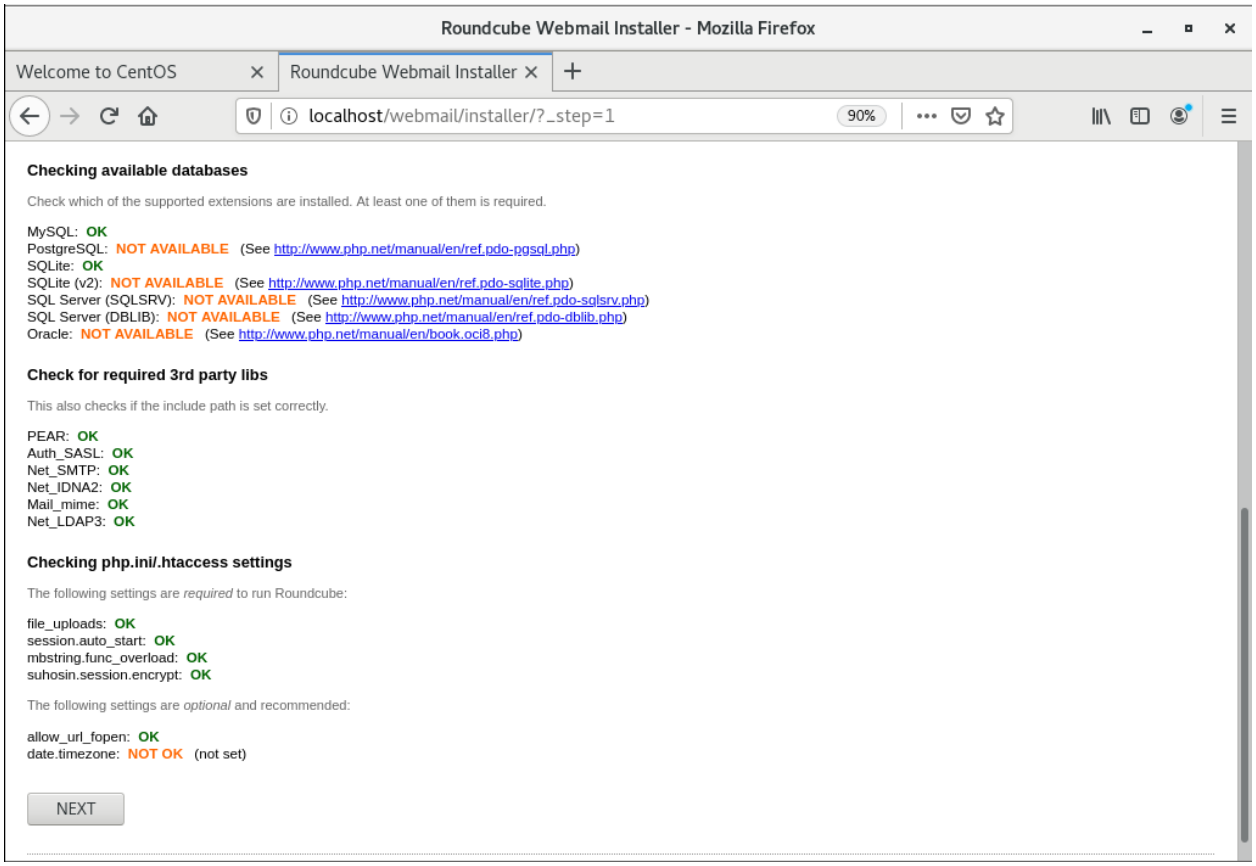
```
// LDAP Password Last Change Date
// Some places use an attribute to store the date of the last password change
// The date is meassured in "days since epoch" (an integer value)
// Whenever the password is changed, the attribute will be updated if set (e.g.
shadowLastChange)
$config['password_ldap_lchattr'] = '';

// LDAP Samba password attribute, e.g. sambaNTPassword
// Name of the LDAP's Samba attribute used for storing user password
$config['password_ldap_samba_pwattr'] = '';

// LDAP Samba Password Last Change Date attribute, e.g. sambaPwdLastSet
// Some places use an attribute to store the date of the last password change
// The date is meassured in "seconds since epoch" (an integer value)
// Whenever the password is changed, the attribute will be updated if set
$config['password_ldap_samba_lchattr'] = '';

// LDAP PPolicy Driver options
// ---------------------------------

// LDAP Change password command - filename of the perl script
// Example: 'change_ldap_pass.pl'
$config['password_ldap_ppolicy_cmd'] = 'change_ldap_pass.pl';

// LDAP URI
// Example: 'ldap://ldap.example.com/ ldaps://ldap2.example.com:636/'
$config['password_ldap_ppolicy_uri'] = 'ldap://25.16.221.130/';

// LDAP base name (root directory)
// Exemple: 'dc=exemple,dc=com'
$config['password_ldap_ppolicy_basedn'] = 'dc=apex,dc=com';

$config['password_ldap_ppolicy_searchDN'] = 'cn=someuser,dc=apex,dc=com';

$config['password_ldap_ppolicy_searchPW'] = 'secret';

// LDAP search filter
// Example: '(uid=%login)'
// Example: '(&(objectClass=posixAccount)(uid=%login))'
$config['password_ldap_ppolicy_search_filter'] = '(uid=%login)';

// CA Certificate file if in URI is LDAPS connection
$config['password_ldap_ppolicy_cafile'] = '/etc/ssl/cacert.crt';
```

Verificar la integra instalación de webmail

Se abre en el navegador la siguiente dirección: **http://loalchost/webmail/installer** apareciendo esto:







Conectar Bacula con WebMail

Instalar el cliente del bacula

```
sudo yum install bacula-client
```

Crear una contraseña aleatoria, para el servidor

```
date +%s | sha256sum | base64 | head -c 33 ; echo
```

Abrir el archivo /etc/bacula/bacula-fd.conf

```
sudo nano /etc/bacula/bacula-fd.conf
```

Escribir los siguientes campos:

```
Director {
  Name = BackupServer-dir # nombre del servidor con -dir
  Password = "Y2Q5ODUyMWM0YTFhYjA3NTcwYmU5OTA4Y"
}
FileDaemon {                          # this is me
  Name = ClientHost-fd
  FDAddress = 25.16.221.130 # 25.16.221.130 = ip client address
  FDport = 9102                  # where we listen for the director
  WorkingDirectory = /var/spool/bacula
  Pid Directory = /var/run
  Maximum Concurrent Jobs = 20
}
Messages {
  Name = Standard
  director =  BackupServer-dir = all, !skipped, !restored
  # nombre del servidor con -dir
}
```

Ejecutar el siguiente comando, y ver si tiene errores, si no escribe nada, es porque está bien.

```
sudo bacula-fd -tc /etc/bacula/bacula-fd.conf
```

Ejecutar los servicios.

```
sudo systemctl restart bacula-fd
sudo systemctl enable bacula-fd --now
sudo mkdir -p /bacula/restore
sudo chown -R bacula:bacula /bacula
sudo chmod -R 700 /bacula
```

CREACIÓN DEL SERVIDOR BACULA

```
#Instalar Bacula, MariaDB y otros componentes
sudo yum install -y bacula-director bacula-storage bacula-console
bacula-client mariadb-server

#Empezar el servicio de MariaDB
sudo systemctl start mariadb

/usr/libexec/bacula/grant_mysql_privileges
/usr/libexec/bacula/create_mysql_database -u root
/usr/libexec/bacula/make_mysql_tables -u bacula
sudo mysql_secure_installation

#Entrar a mysql
mysql -u root -p

#Se crea la password para la database
MariaDB [(none)]  UPDATE mysql.user SET
Password=PASSWORD('bacula_db_password') WHERE User='bacula';
MariaDB [(none)]  FLUSH PRIVILEGES;
MariaDB [(none)]  exit

#Habilitar mariadb
echo "Elegir la ruta mysql";
sudo systemctl enable mariadb
sudo alternatives --config libbaccats.so

#Instalar Bacula, MariaDB y otros componentes
echo "Creando directorios para los backups";
sudo mkdir -p /bacula/backup /bacula/restore
sudo chown -R bacula:bacula /bacula
sudo chmod -R 700 /bacula
```

Los siguientes cambios se hacen sobre el archivo que se encuentra dentro del directorio **/etc/bacula/bacula-dir.conf**

```
sudo nano /etc/bacula/bacula-dir.conf
```

```
#
# Default Bacula Director Configuration file
#
#  The only thing that MUST be changed is to add one or more
#  file or directory names in the Include directive of the
#  FileSet resource.
#
#  For Bacula release 5.2.6 (21 February 2013) -- debian squeeze/sid
#
#  You might also want to change the default email address
#  from root to your address.  See the "mail" and "operator"
#  directives in the Messages resource.
#
Director {                            # define myself
  Name = bacula-dir
  DIRport = 9101                # where we listen for UA connections
  QueryFile = "/etc/bacula/query.sql"
  WorkingDirectory = "/var/spool/bacula"
  PidDirectory = "/var/run"
  Maximum Concurrent Jobs = 1
  Password = "@@DIR_PASSWORD@@"         # Console password
  Messages = Daemon
  DirAddress = 127.0.0.1
}
#
Job {
  Name = "BackupLocalFiles"
  JobDefs = "DefaultJob"
}
#
Job {
  Name = "RestoreLocalFiles"
  Type = Restore
  Client=BackupServer-fd
  FileSet="Full Set"
  Storage = File
  Pool = Default
  Messages = Standard
  Where = /bacula/restore
}

#
FileSet {
  Name = "Full Set"
```

```
  Include {
    Options {
      signature = MD5
      compression = GZIP
    }
File = /
}
  Exclude {
    File = /var/lib/bacula
    File = /proc
    File = /tmp
    File = /.journal
    File = /.fsck
    File = /bacula
  }
}

================================================================
Storage {
  Name = File
# Do not use "localhost" here
  Address = backup_server_private_FQDN  # N.B. Use a fully qualified
name here
  SDPort = 9103
  Password = "@@SD_PASSWORD@@"
  Device = FileStorage
  Media Type = File
}

================================================================
# Generic catalog service
Catalog {
  Name = MyCatalog
# Uncomment the following line if you want the dbi driver
# dbdriver = "dbi:postgresql"; dbaddress = 127.0.0.1; dbport =
  dbname = "bacula"; dbuser = "bacula"; dbpassword =
"bacula_db_password"
}
```

```
=========================================================
# File Pool definition
Pool {
  Name = File
  Pool Type = Backup
  Label Format = Local-
  Recycle = yes                       # Bacula can automatically
recycle Volumes
  AutoPrune = yes                     # Prune expired volumes
  Volume Retention = 365 days         # one year
  Maximum Volume Bytes = 50G          # Limit Volume size to
something reasonable
  Maximum Volumes = 100               # Limit number of Volumes in
Pool
}
```

Los siguientes cambios se hacen sobre el archivo **/etc/bacula/conf.d/filesets.conf**, en caso no exista el archivo crearlo con el mismo nombre.

```
sudo nano /etc/bacula/conf.d/filesets.conf
```

```
FileSet {
  Name = "Completo"
  Include {
    Options {
      signature = MD5
      compression = GZIP
    }
    File = /home

  }
  Exclude {
```

```
    File = /home/archivo_a_ignorar
    File = /home/archivo_a_ignorar2
    File = /home/archivo_a_ignorar3
  }
}
```

Los siguientes cambios se generan sobre el archivo **/etc/bacula/conf.d/clients.conf**, en caso de no existir el archivo crearlo con el mismo nombre.

```
sudo nano /etc/bacula/conf.d/clients.conf
```

```
Client {
  Name = ClientHost-fd
  Address = client_private_FQDN
  FDPort = 9102
  Catalog = MyCatalog
  Password = "Y2Q5ODUyMWM0YTFhYjA3NTcwYmU5OTA4Y"          # password
for Remote FileDaemon
  File Retention = 30 days            # 30 days
  Job Retention = 6 months            # six months
  AutoPrune = yes                     # Prune expired Jobs/Files
}

Job {
  Name = "BackupClientHost"
  JobDefs = "DefaultJob"
  Client = ClientHost-fd
  Pool = RemoteFile
  FileSet="Home and Etc"
}
```

Los siguientes cambios se generan sobre el archivo **/etc/bacula/conf.d/pools.conf**, en caso de no existir el archivo crearlo con el mismo nombre.

```
sudo nano /etc/bacula/conf.d/pools.conf
```

```
Pool {
  Name = RemoteFile
  Pool Type = Backup
  Label Format = Remote-
  Recycle = yes
  AutoPrune = yes
  Volume Retention = 365 days
    Maximum volume Bytes = 50G
  Maximum Volumes = 100
}
```

Se desactiva SELinux:

```
setenforce 0
```

Se verifica si existen errores con el comando:

```
sudo bacula-dir -tc /etc/bacula/bacula-dir.conf
```

Los siguientes cambios se hacen sobre el archivo que se encuentra dentro del directorio **/etc/bacula/bacula-sd.conf**

```
sudo nano /etc/bacula/bacula-sd.conf
```

```
=====================================================================
Storage {                               # definition of myself
  Name = BackupServer-sd
  SDPort = 9103                  # Director's port
  WorkingDirectory = "/var/lib/bacula"
  Pid Directory = "/var/run/bacula"
  Maximum Concurrent Jobs = 20
  SDAddress = backup_server_private_FQDN
}
=====================================================================
Device {
  Name = FileStorage
  Media Type = File
  Archive Device = /bacula/backup
  LabelMedia = yes;                     # lets Bacula label unlabeled
media
  Random Access = Yes;
  AutomaticMount = yes;                 # when device opened, read it
  RemovableMedia = no;
  AlwaysOpen = no;
}
```

Se verifica si existen errores con el comando:

```
sudo bacula-sd -tc /etc/bacula/bacula-sd.conf
```

Se sigue con los comandos:

```
#Generando Passwords para la database
echo "Generando Passwords para la db...";
DIR_PASSWORD=`date +%s | sha256sum | base64 | head -c 33`
sudo sed -i "s/@@DIR_PASSWORD@@/${DIR_PASSWORD}/"
/etc/bacula/bacula-dir.conf
```

```
sudo sed -i "s/@@DIR_PASSWORD@@/${DIR_PASSWORD}/"
/etc/bacula/bconsole.conf
SD_PASSWORD=`date +%s | sha256sum | base64 | head -c 33`
sudo sed -i "s/@@SD_PASSWORD@@/${SD_PASSWORD}/"
/etc/bacula/bacula-sd.conf
sudo sed -i "s/@@SD_PASSWORD@@/${SD_PASSWORD}/"
/etc/bacula/bacula-dir.conf

echo "Iniciando Bacula";

sudo systemctl start bacula-dir
sudo systemctl start bacula-sd
sudo systemctl start bacula-fd

sudo systemctl enable bacula-dir
sudo systemctl enable bacula-sd
sudo systemctl enable bacula-fd
```

Una vez seguidos estos pasos se puede acceder a la terminal de bacula para empezar a probar cada una de sus funcionalidades.

```
bconsole
```

CONCLUSIONES

- Hamachi ayuda mucho para hacer una LAN pequeña, pero funcional, y así poder trabajar desde lugares “lejanos”, Hamachi es una gran herramienta para el desarrollo de este proyecto.
- Bacula puede ser una herramienta muy útil para el respaldo y la restauración de datos.
- WebMail, tiene muchas vertientes, y se puede utilizar con varios plugins, estos especializados en un Sistema operativo, utilizando RoundCube como plugin de Centos 7, y así utilizar WebMail.
- RoundCube permite la conexión a Ldap, esto en el archivo de configuración del mismo, permitiendo la verificación de usuarios ldap, desde el webmail, y así loguearse con usuarios de ldap.
- El conectarse desde webmail, a bacula se realiza de una manera sencilla, esto es por que el servidor de webmail, se comporta como cliente de bacula, el cual este le hará backup a los archivos cambiados de home, de cada cliente.
- Logmein Hamachi no permite administrar la IP de cada cliente conectado, por ende se deben de usar las IPs virtuales que genera hamachi.
- el parámetro URI del servidor ldap no es necesario para su uso, este solo es útil en interfaz gráfica, por lo tanto no fue necesario utilizarlo.
- Como parte del funcionamiento de OpenLDAP es muy importante hacer la activación de firewall para que esté funcionando correctamente.

RECOMENDACIONES

- Instalar hamachi en todas las máquinas, y conectarlas a la misma red, para así poder conectarse con cada uno de los servidores.
- Conectar solo 5 máquinas como máximo a la red hamachi.
- Crear un servidor más, específicamente para el almacenamiento de los backups, dejando al servidor tres con la única función de pasar los datos del servidor dos hacia el nuevo servidor.
- Utilizar los script tal y como aparecen, leerlos antes de ejecutarlos, ya que unos tienen configuraciones de archivos implícitos.
- Conectar el servidor webmail, a los otros dos servidores, esto utilizando las ips, que proporciona hamachi, y no la de la red local, en todas las configuraciones que pidan la ip.
- Instalar el cliente de bacula, antes de configurar la conexión de webmail y bacula
- Instalar el cliente de ldap, antes de configurar la conexión de webmail y ldap.
- Proporcionar a la máquina virtual de windows una ram conveniente y almacenamiento, para no tener ningún inconveniente, (2 ram mínimo, y 25 gb de memoria mínimo.)
- Utilizar la máquina del servidor webmail, como máquina para el cliente de CentOS, y así no sobrecargar la red de hamachi.
- Encriptar las contraseñas de los usuarios con el método o algoritmos hash es más útil ya que durante la autenticación de usuarios se optimiza el proceso entre cliente y servidor.

- Para la configuración del servidor LDAP se necesita la escritura de muchos archivos, dado esto es mejor generar archivos previo con la configuración y los scripts de escritura para estos mismo archivos.

BIBLIOGRAFÍA

- https://es.wikipedia.org/wiki/Protocolo_ligero_de_acceso_a_directorios
- https://webmail.gestiondecorreo.com/
- https://es.wikipedia.org/wiki/Webmail
- https://roundcube.net/
- https://web.archive.org/web/20121213153742/http://myroundcube.com/
- https://web.archive.org/web/20140416175914/https://identidad.usal.es/web/manualRC
- https://es.wikipedia.org/wiki/Webmail
- https://www.bacula.org/
- https://www.bacula.org/documentation/documentation/
- https://www.linux-kvm.org/page/Main_Page
- https://www.redhat.com/en
- https://es.wikipedia.org/wiki/Cliente-servidor
- https://www.logmein.com/es
- https://es.wikipedia.org/wiki/Hamachi
- https://www.digitalocean.com/community/tutorials/how-to-back-up-a-centos-7-server-with-bacula
- https://github.com/roundcube/roundcubemail/wiki/Configuration:-LDAP-Address-Books
- https://support.logmeininc.com/es/central/help/como-instalar-el-cliente-en-un-ordenador-local-central-t-hamachi-add-attached-local
- https://www.tecnocienciaperu.com/apuntes-informatica/como-instalar-roundcube-1-4-en-centos-7/

- https://www.digitalocean.com/community/tutorials/how-to-install-bacula-server-on-centos-7
- https://www.digitalocean.com/community/tutorials/how-to-back-up-a-centos-7-server-with-bacula
- https://www.bacula.org/5.2.x-manuals/en/main/main/Configuring_Director.html
- https://adsm.org/lists/html/Bacula-users/2008-07/msg00238.html
- https://www.itzgeek.com/how-tos/linux/centos-how-tos/step-step-openldap-server-configuration-centos-7-rhel-7.html