Administración y Despliegue de Ejabberd.

Nota sobre la obra:

Respecto al autor

La obra está realizada por Kamal Majaiti y queda atribuida a Kamal Majaiti con el cual se puede contactar mediante twitter mediante el siguiente Nick @kamalsenior o a través de la url https://twitter.com/kamalsenior. Ya sea para resolver una duda o cualquier otro interés.

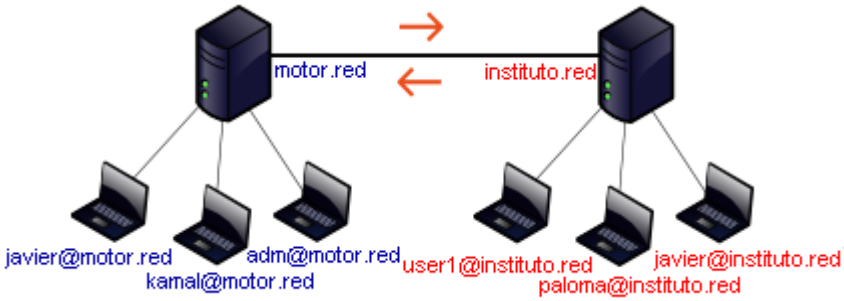
1. 0 Protocolo XMPP/jabber.

Sus siglas significan *Protocolo extensible de mensajería y comunicación de presencia. El protocolo XMPP es un protocolo Abierto y extensible ya que los mensajes se intercambian en formato XML. XMPP es capaz de comunicarse con otros servicios de Mensajería Instantánea a través de pasarelas XMPP. Y cabe destacar que XMPP no está controlado por ninguna empresa u organización. Todos los mensajes enviados a través de este protocolo serán fragmentos de XML. Las cuales forman parte del documento que sería la conversación entera.*

```
<message to=kamal@xxmp.red'>
 <subject>Qué tal estás?</subject>
 <body>Dime si puedes quedar esta noche...</body>
</message>
```

Gracias a la simplicidad del mensaje XML se aprecia perfectamente a quien va dirigido. El titulo del mensaje y el contenido así como que la compatibilidad entre diferentes sistemas está asegurada gracias a la estandarización de la w3c y las restricciones de XML.

El protocolo XMPP sigue la arquitectura cliente-servidor. Es decir Los clientes XMPP únicamente se comunican con el servidor XMPP de su dominio, que se encarga de ofrecerles el servicio.



Y en caso de que un usuario se intente comunicar con otro usuario de otro dominio el servidor XMPP de su red será el encargado de conectarse al otro servidor e enviarle dicho mensaje al usuario indicado. Cabe decir que toda la comunicación de un dominio se controla en el servidor pues este deberá manejar todos los mensajes de entrada y salida de los usuarios de su dominio.

XMPP también tiene otra funcionalidad importante que es la distinción de un usuario y una conexión de éste. Es decir es capaz de tener conectados varios dispositivos con una misma cuenta. De forma que se pueda comunicar desde cualquiera de los dispositivos.

Antes de continuar con la lectura del documento cabe mencionar un pequeño detalle estoy hablando del JID o Identificador jabber es sumamente importante para no confundirse más adelante. A partir de ahora el JID es y está formado por:

Diversas características de XMPP

Mensajería

Si el usuario de un dominio no está conectado pero se le envía una mensaje el servidor XMPP de ese dominio lo almacenara hasta que se conecta conecte. Esto quiere decir que soporta mensajería offline. En cambio si un usuario dispone de muchas conexiones activas desde diferentes dispositivos este se encargara de enviárselo a la mas priorizada (Se le indica en el servidor).

Estados

Soporta estados en texto es decir aparte del conectado/no conectado se puede escribir uno diciendo del tipo "Salí a Comer" o "estoy en la piscina".

Listas de contactos

La lista de contactos permite crear listas de contactos estas listas son almacenadas en el servidor del dominio del cliente.

Seguridad XMPP

Cada vez que se añade un servicio a una empresa se deben modificar las políticas de seguridad de la empresa en este caso se deben realizar las siguientes acciones.

- Establecer políticas de uso.
- Restringir los privilegios del usuario en el PC que se comunica.
- Actualizar el Sistema.
- Utilizar productos antivirus y antispyware en el caso de Windows.
- No confiar en servidores IM externos para el uso interno de IM.
- Crear rutas de comunicación seguras.
- Configurar debidamente los cortafuegos y los IDS de la red.
- Filtrar todo el tráfico través de un servidor proxy para proporcionar capacidades adicionales de filtrado y monitoreo de tráfico IM.
- Bloquear el acceso a conocidos servidores públicos IM que no han sido autorizados explícitamente.
- Bloquear los puertos IM conocidos.
- Monitorear con un sistema los túneles VPN, SSH y SSL.

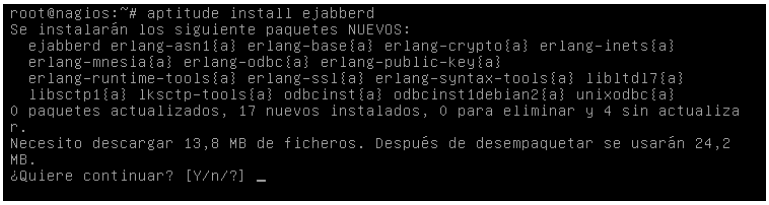
2.0 Despliegue de un Servidor XMPP.

A continuación se va a desplegar un servidor de mensajería XMPP sobre un servidor Linux. Pero los pasos descritos igual que en el protocolo IRC se pueden aplicar a un sistema Windows.

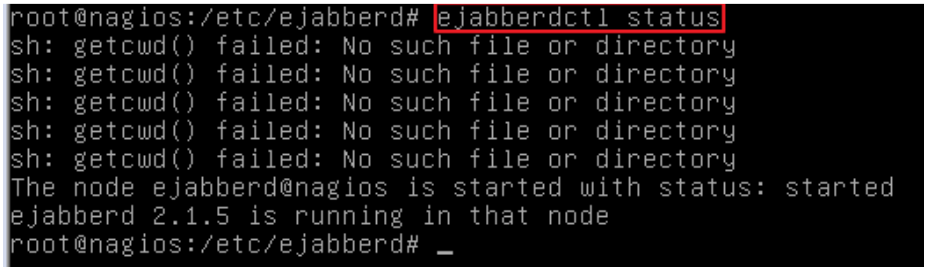
Comenzamos instalando un servidor XMPP en Linux más conocido como jabber por lo cual buscamos el demonio para este protocolo.

```
root@nagios:~# aptitude search jabberd
p   ejabberd                        - distributed, fault-tolerant Jabber/XMPP se
p   jabberd14                       - Instant messaging server using the Jabber/
p   libjabberd2                     - Runtime library for the Jabber/XMPP instan
p   libjabberd2-dev                 - Development files for the Jabber/XMPP inst
root@nagios:~# _
```

Y como se aprecia disponemos de diferentes opciones a instalar se recomienda usar ejabberd ya que tiene administración por web y por consola entre otras diversas opciones como que también está disponible para Windows. Para instalarlo usamos el comando de la imagen. Para instalar el demonio usamos el comando ***aptitude install ejabberd***



Una vez instalado podemos comprobar si todo funciona correctamente. Para ello usamos el comando ***ejabberdctl status***

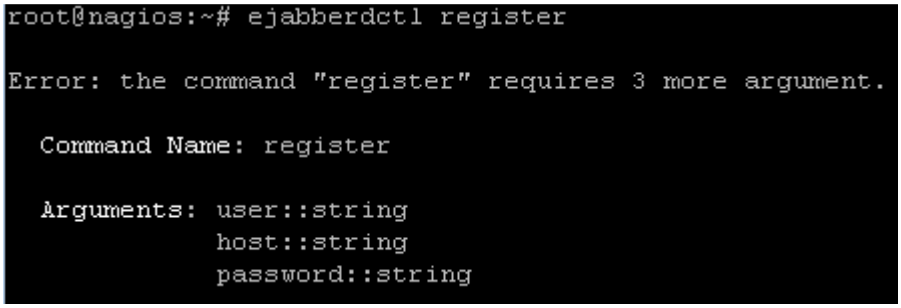


Después de apreciar el correcto funcionamiento conviene crear un primer usuario para administrar el nodo XMPP.

Registro de un usuario para la administración.

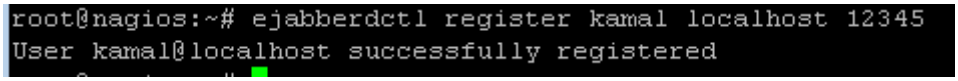
Para registrar un usuario ejecutamos el siguiente comando.

ejabberdctl register usuario servidor_Virtual contraseña



- Donde el servidor virtual es el nodo jabber al que va a acceder el usuario por lo cual debe estar en el DNS.

  Por ejemplo:



Para comprobar si se registro podemos usar el comando ***ejabberdctl registered_users localhost***

```
root@nagios:~# ejabberdctl registered_users localhost
kamal
```

Después de registrarlo queda darle privilegios de administración.

Delegación de privilegios de administración a un usuario.

Para dar permisos a un usuario en el demonio ejabberd debemos indicárselo en su fichero de configuración concretamente este fichero. ***/etc/ejabberd/ejabberd.cfg*** en el cual buscamos la línea "Admin user" que es una acl que da privilegios de administración a los usuarios.

Concretamente esta.

```
%% Admin user
{acl, admin, {user, "", "localhost"}}.
```

La modificamos introduciendo los datos usados para el registro del usuario en el punto anterior. En mi caso queda así.

```
%% Admin user
{acl, admin, {user, "kamal", "localhost"}}.
```

Después si miramos un poco el fichero de configuración apreciamos que la administración se realiza desde el puerto 5280.

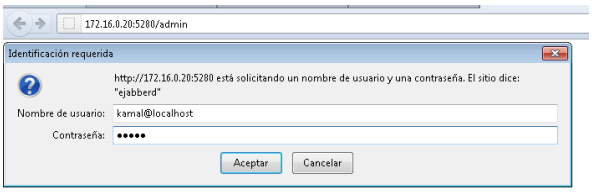
```
{5280, ejabberd_http, [
                       %%{request_handlers,
                       %% [
                       %%  {["pub", "archive"], mod_http_fileserver}
                       %% ]},
                       %%captcha,
                       http_bind,
                       http_poll,
                       web_admin
                      ]}
```

Pero antes debemos reiniciar el demonio para que recoja los nuevos parámetros modificados.

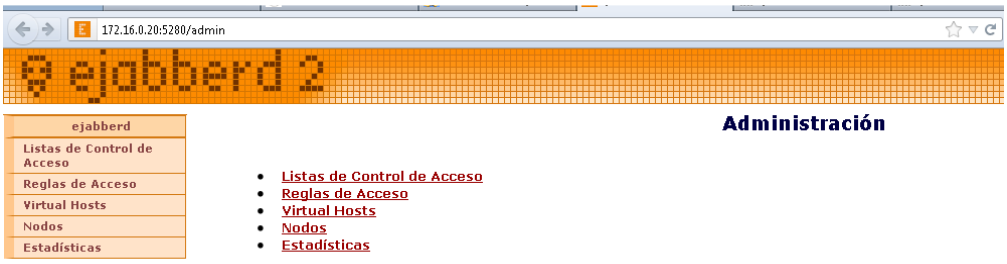
```
root@nagios:~# /etc/init.d/ejabberd restart
Restarting jabber server: ejabberd
```

Una vez hecho entramos por interface web a ese puerto y entramos con el formato usuario@dominio

Ejemplo:



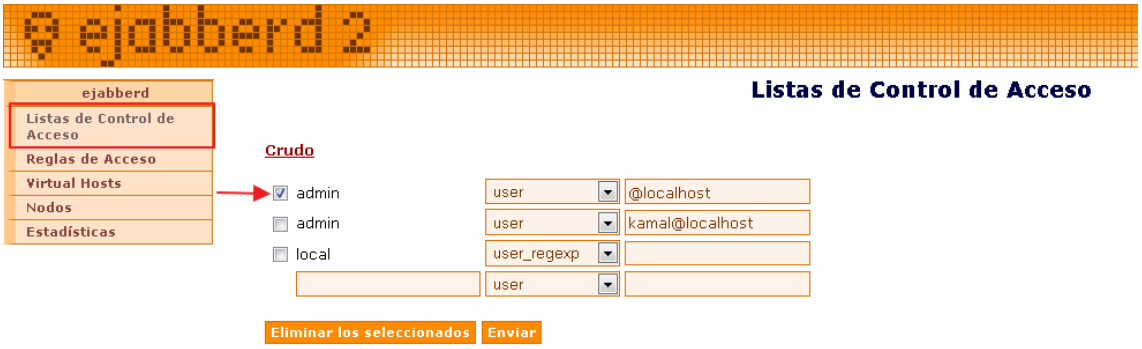
Le damos a aceptar y estamos dentro.



Lo primero que se debe hacer ahora es crear un backup del fichero de configuración antes de proceder con la edicion.

```
root@nagios:~# cd /etc/ejabberd/
root@nagios:/etc/ejabberd# cp ejabberd.cfg ejabberd.cfg_cop
```

Una vez hecho nos vamos a la opción ACL y borramos los usuarios por defecto para evitar posibles intrusiones en un futuro.



Y pulsamos el botón eliminar seleccionados.

También borramos este:

Una vez hecho procedemos con la configuración.

Añadir nombres de dominio a ejabberd.

Para añadir nombres de dominio a ejabberd debemos editar el fichero de configuración de forma manual. Buscamos la palabra Hostname en el fichero de configuración /etc/ejabberd/ejabberd.cfg y editamos añadiendo nuestro host.

```
%% Hostname
{hosts, ["localhost","nagios.local.red"]}.
```

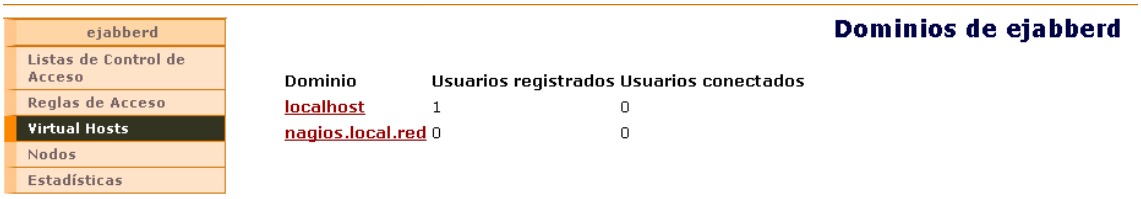
En caso de no disponer dns conviene añadirlo al fichero Hosts del S.O.

```
root@nagios:/etc/ejabberd# cat /etc/hosts
127.0.0.1       nagios.local.red
127.0.0.1       localhost
127.0.1.1       nagios
```

Una vez hecho el cambio reiniciamos el servidor jabber.

```
root@nagios:/etc/ejabberd# /etc/init.d/ejabberd restart
Restarting jabber server: ejabberd.
root@nagios:/etc/ejabberd#
```

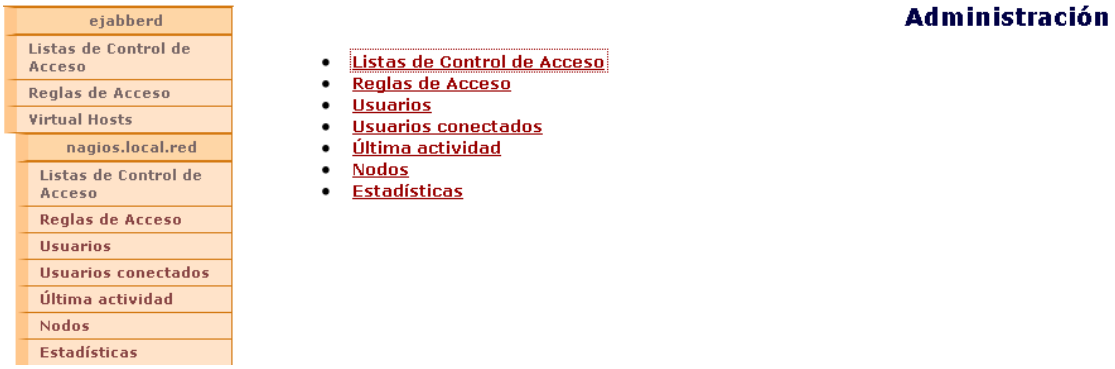
Entonces ahora si vamos a VirtualHost en la página web se aprecia que ya podemos registrar usuarios de nuestra empresa.



2.1.2 Administración básica de un dominio Jabber.

2.1.2.1 Controles de comunicación del dominio Jabber

Si pulsamos sobre el host nagios.local.red (En mi caso) podemos ver diversas opciones.



En este apartado trataremos las ACL del dominio que se encargan de controlar la comunicación y los tipos que hay son varios los más destacables son.

Globales Afectan a toda la comunicación del servidor.

Tipo ***user***: Bloquea la comunicación de un usuario a un usuario a afecta a todas las comunicaciones que pasen por el servidor.

Tipo ***server***: Bloquea la comunicación de servidor a servidor que pasen a través del servidor o salientes/entrantes de este.

Locales Afectan a los hosts virtuales del servidor.

Tipo ***user_rexexp***: Solo afecta a los usuarios de un host Virtual del servidor.

Tipo ***server_rexexp***: Solo afecta a la comunicación entre servidores virtuales.

Tipo ***node_rexexp***: Solo afecta a un nodo jabber es decir

ACL Especiales.

Estas ACL son especiales porque facilitan el uso de comodines.

Por ejemplo bloquear usuarios que empiecen por adm

Tipo ***user_glob*** : Filtra por usuarios.

[{acl, 'adm*', {user_glob, "usr*"}}].

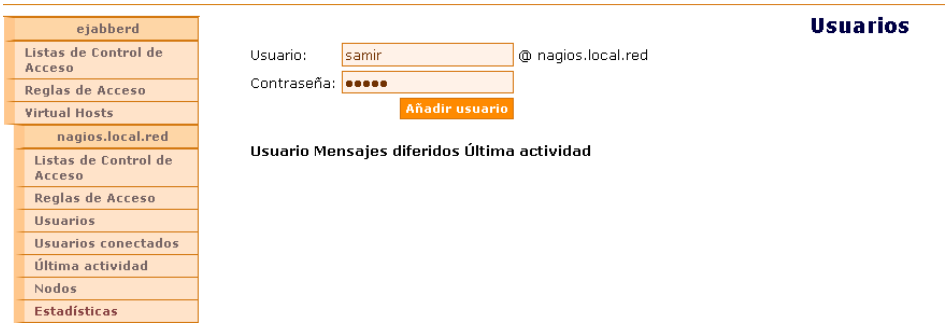
Tipo ***server_glob***: Filtra por nombre de servidor.

Tipo ***node_glob***: Filtra por el nodo del JID.

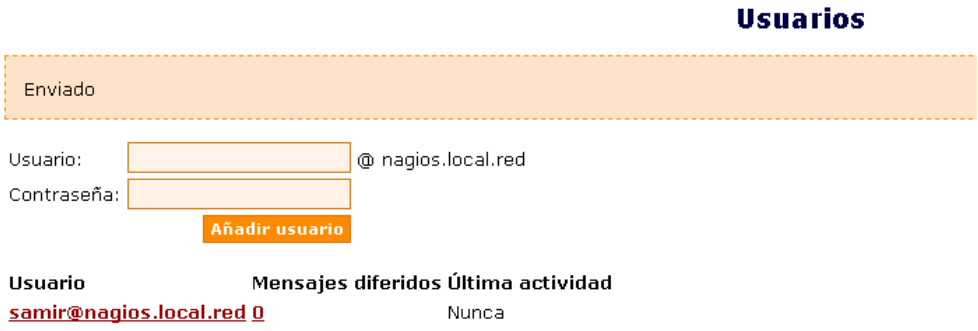
Tipo ***All***: Afecta a todos los JID que contengan ese carácter.

2.1.2.2 Añadir usuarios al servidor jabber.

Nos situamos en el virtualHost donde queremos añadir usuarios y una vez entro añadimos uno. Por ejemplo



Luego le damos a añadir.



Otra forma de hacerlo es desde la línea de comandos con el comando.

```
ejabberdctl register <username> <server> <password>
```

```
root@nagios:~# ejabberdctl register pepe nagios.local.red pepe
User pepe@nagios.local.red successfully registered
```

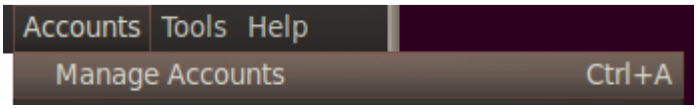
```
ejabberdctl unregister <username> <server>
```

```
root@nagios:~# ejabberdctl register pepe nagios.local.red pepe
User pepe@nagios.local.red successfully registered
root@nagios:~# ejabberdctl unregister pepe nagios.local.red
root@nagios:~# ejabberdctl registered_users nagios.local.red
kamal
samir
root@nagios:~#
```

Y queda registrado en la página inferior.

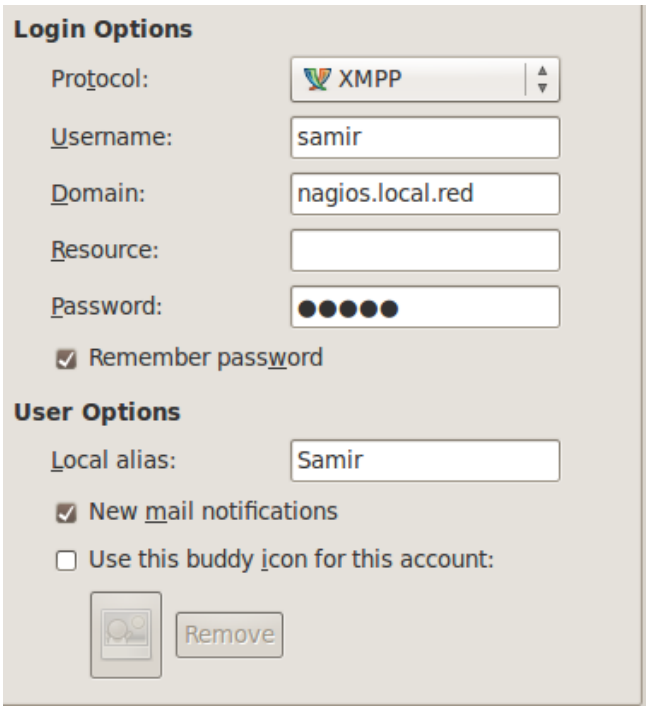
Una vez hecho ya podemos Iniciar sesión con pidgin o cualquier programa que soporte Jabber.

Le damos a cuentas

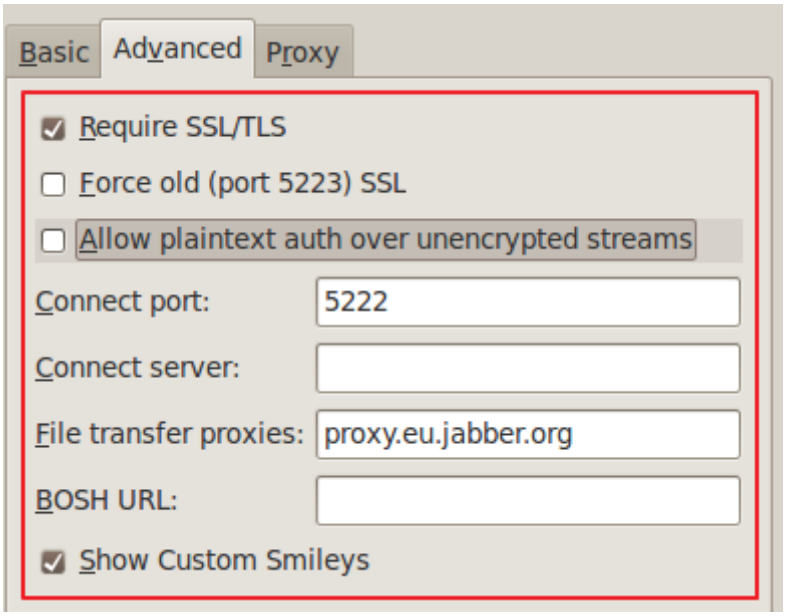


y luego en administración de cuentas pulsamos añadir.

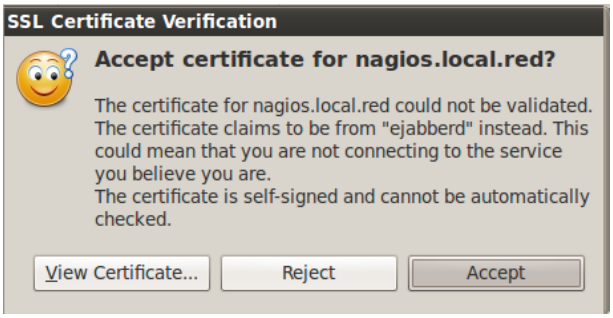
Después rellenamos las opciones de la cuenta Seleccionando el protocolo y sus datos de acceso.



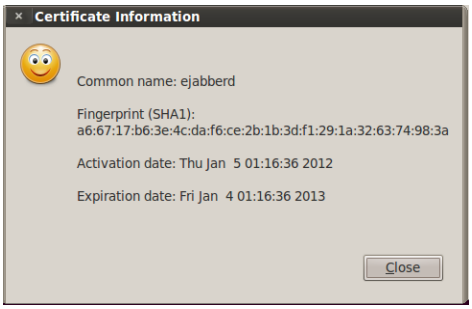
En la siguiente ventana configuramos el acceso SSL.



Una vez hecho le damos a conectar y nos pide confirmación sobre el certificado SSL.



Podemos pulsar View certifícate para ver la firma digital del certificado así como el Nombre Común y las fechas de expedición y expiración.



Una vez aceptado estamos dentro de nuestra red privada XMPP.

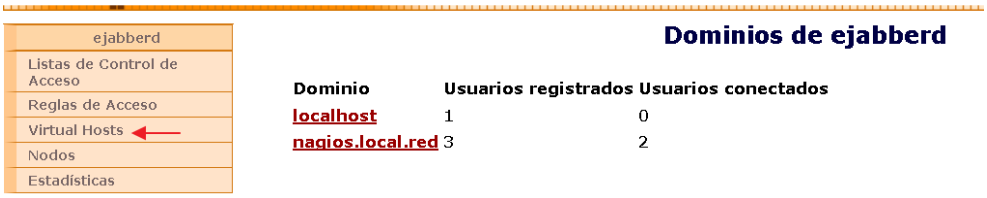
2.1.2.3 Evitar el registro de usuarios automático.

Mediante la directiva Access podemos evitar que se registren usuarios.

`{access, register, [{deny, all}]}.`

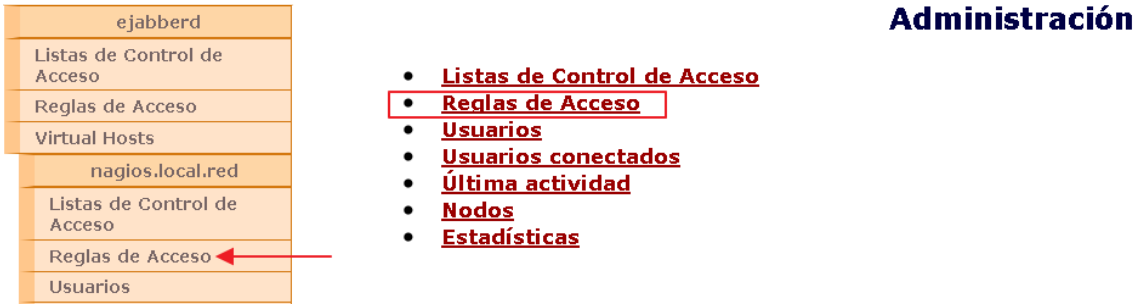
Para introducirla por interface web hacemos esto.

1° Nos vamos a VirtualHosts.



2° Nos colocamos sobre el dominio que deseamos modificar. Y hacemos clic sobre él. En mi caso debo hacer clic sobre nagios.local.red.

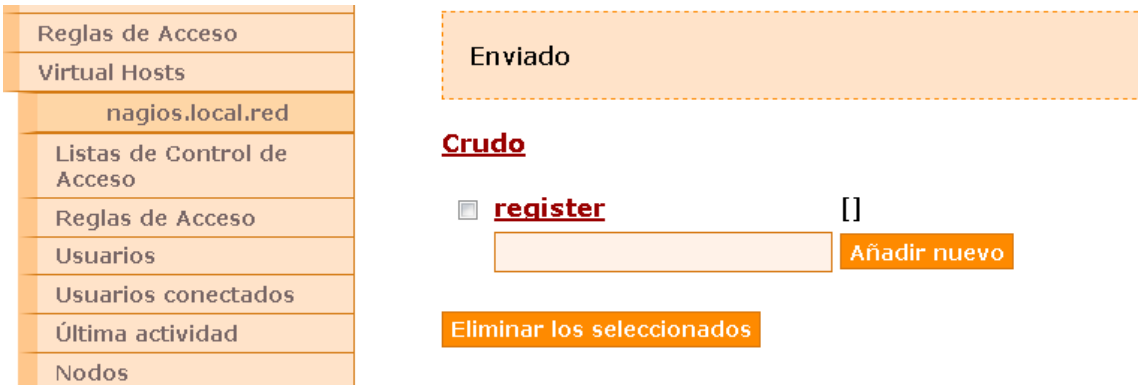
Una vez dentro seleccionamos Reglas de acceso.



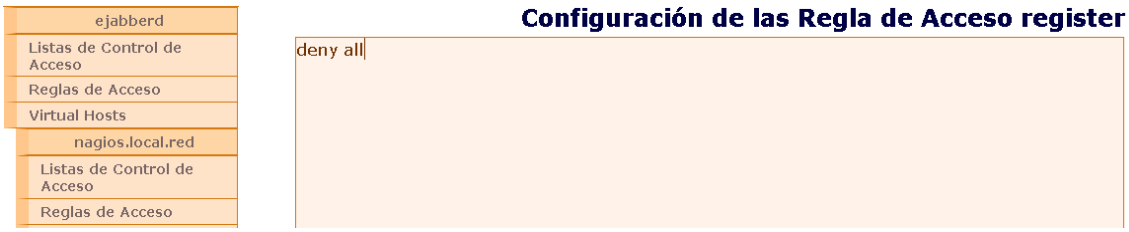
Luego escribimos la directiva Register.

Y Pulsamos el botón añadir.

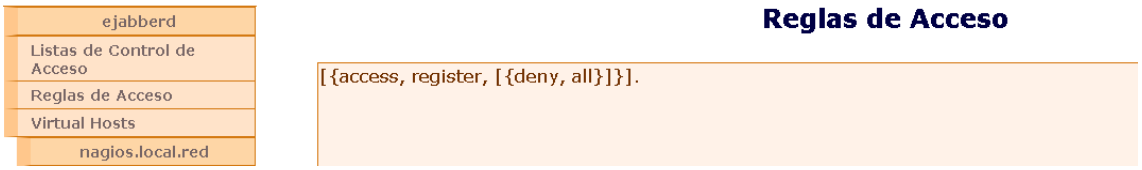
Nos quedara asi:



Ahora pulsamos sobre register.

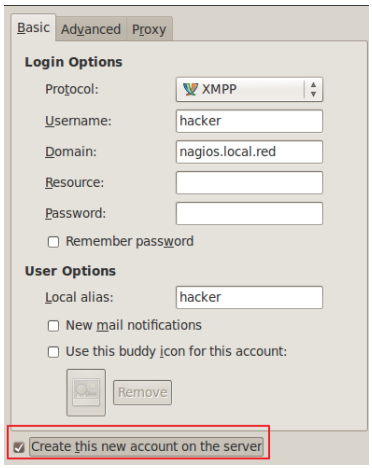


Y dentro escribimos deny all. Entonces en la cabecera aparecerá un mensaje que pone enviado. Si ahora le damos a reglas de acceso y seleccionamos RAW o crudo. Veremos la ACL con formato RAW.



Para ello nos vamos al host donde queremos evitar esto y una vez dentro le damos a reglas de acceso y luego seleccionamos.

Una vez hecha la faena si alguien intenta registrarse pasara esto.



En pidgin crear un usuario ficticio y marcar la opción register. Una vez hecho cuando intente registrarse recibirá este bonito mensaje.

Denegándole el acceso ha dicho servicio de registro[9].

2.1.2.4 Gestión de Salas XXMP.

En jabber igual que en el IRC existen salas publicas que posibilitan la comunicación en grupo.

2.1.2.4.1 Permisos generales sobre las Salas.

- **access_create ◊ Controla la** creación de nuevas Salas
- **default_room_options ◊ Define las opciones por defecto de las salas creadas.**
- **max_users ◊ Define el máximo de usuarios por sala.**
- **history_size ◊ Define el tamaño máximo del historial de la sala.**

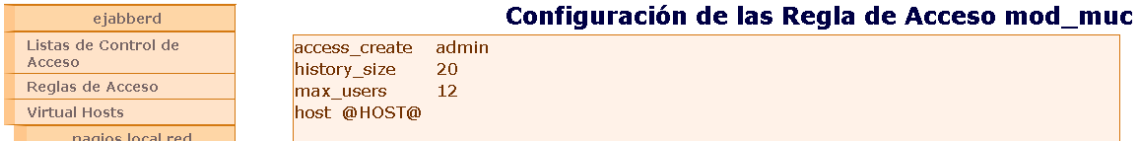
Ejemplo:

```
{mod_muc, [{access_create, kamal}, #Puede tomar valor All
           {history_size, 20},     #Es un valor Decimal.
           {max_users, 12},        #Es un valor Decimal.
           {host, @HOST@}]      #@HOST@ Es una variable que apunta al
                                  #servidor Virtual Jabber de ese
                                  #host.
```

Ahora se puede introducir en ejabber por vía web en la misma zona donde colocamos register.

Una vez realizado le damos a añadir y luego pulsamos sobre ella como la anterior vez.

Y dentro colocamos las directivas.



2.1.2.4.2 Características Por defecto de las Salas.

La directiva default_room_options establece una configuración por defecto de las salas creadas.

Sus opciones importantes son los siguientes.

Nota: Tipo F/T indica que el valor solo puede ser True o False.

allow_change_subj ◊ Indican si los usuarios de la sala pueden cambiar el tema de esta. F/T

allow_private_messages ◊Indica si se permite la mensajería privada. Tipo F/T

anonymous false ◊ Indica si se permite el acceso anónimo. Tipo F/T

allow_user_invites ◊ Indica si los usuarios pueden acceder a la sala. Tipo F/T

allow_visitor_nickchange ◊ Indica si el usuario puede cambiar de Nick en la sala. Tipo F/T

allow_visitor_status ◊ Indica si pueden ver el estado del usuario en la sala. Tipo F/T

logging ◊ Indica Si los mensajes se guardan en un registro. Tipo F/T

members_only ◊ Indica si Solo pueden entrar miembros de la sala. Tipo F/T

password ◊ Indica la Contraseña de acceso. String entre comillas dobles ej: "123"

password_protected ◊ Indica si Pedir contraseña de acceso. Tipo F/T

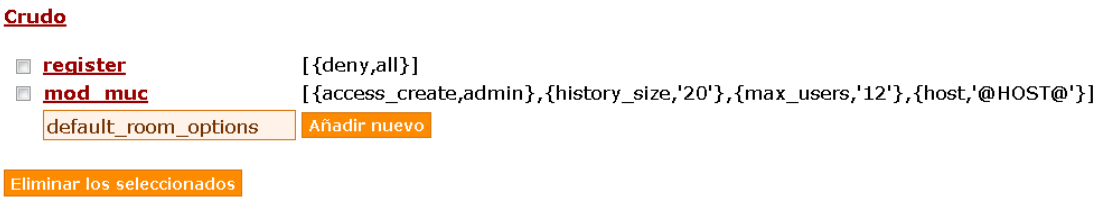
persistent ◊ Indica que Se guardan cambios persistentes de la sala. Tipo F/T

public ◊ Indica si aparece en la lista de salas. Tipo F/T

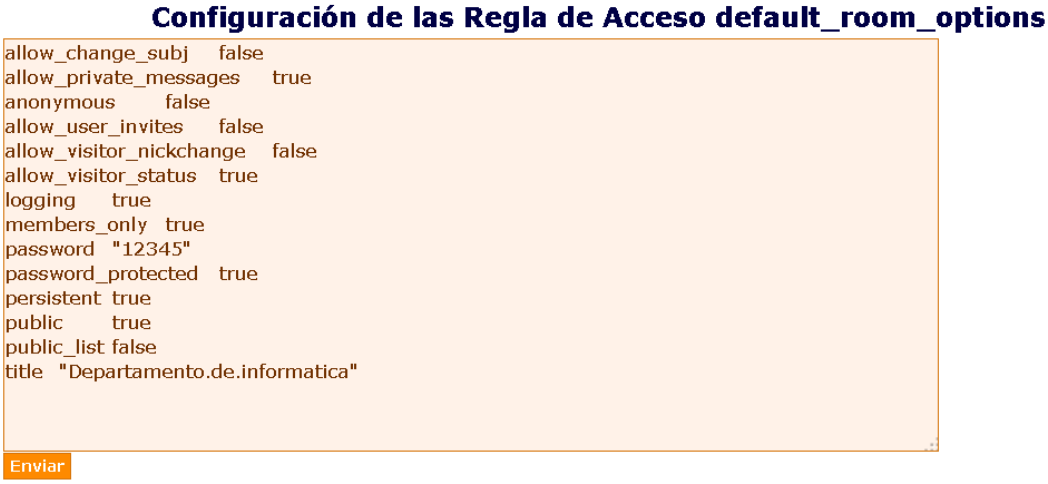
public_list ◊ Indica si No permite ver la lista de usuarios de la sala. Tipo F/T

title ◊ Indica Nombre de la sala. String entre comillas dobles ej: "Dep.Informatica" ¡Sin espacios!

Para introducirla creamos una nueva regla con el nombre de la directiva.



Después hacemos clic sobre ella e introducimos las opciones de la regla.



Una vez hecho la directiva se aplica directamente.

3. 0 Bibliografía.

Estándar XMPP
www.jabberes.org/files/download/Jabber-XMPP.pdf

Instalación y Administración de Ejabberd.
www.process-one.net/docs/ejabberd/guide_en.pdf