Administración remota en GNU/Linux

O obxectivo é probas distintas ferramentas para administración remota dunha distribución GNU/Linux dende clientes Linux e Windows. Nos exemplos empregaremos Ubuntu Desktop.

1 OpenSSH

1.1 Introdución

OpenSSH é un conxunto de aplicacións que permiten realizar comunicacións cifradas unha rede utilizando o protocolo SSH. O conxunto de aplicacións incluídas son:

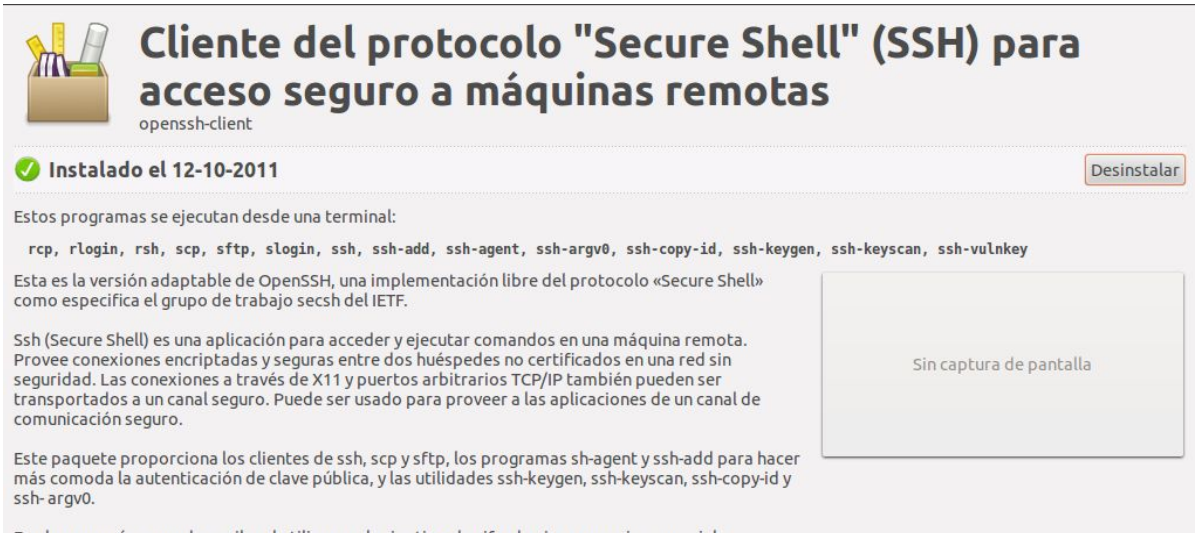
- ssh: acceso remoto á máquina.
- scp: permite copiar arquivos.
- sftp: permite copiar arquivos, pero como faríamos mediante ftp.
- sshd: o servidor de ssh.
- ssh-keygen: ferramenta para inspeccionar e xerar as claves que se utilizan para a autenticación.

1.2 Instalación

En Ubuntu as utilidades cliente e o servidor de OpenSSH instálanse por separado. Vamos a ver como instalalas

- Cliente: en Ubuntu Desktop o cliente ssh ven instalado por defecto. É necesario instalar o paquete **openssh-client.** A seguinte captura amosa unha consulta do para coñecer se o paquete está instalado no Centro de software de Ubuntu:

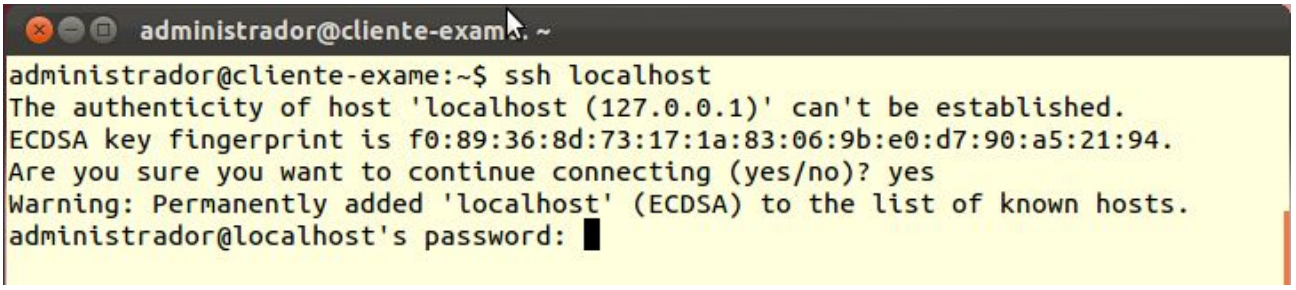
  O paquete instala varias utilidades para conexións e transferencia de ficheiros dende a liña de comandos:



- Servidor: para dispoñer do función de servidor e poder conectarnos ao equipo debemos instalar o paquete **openssh-server**, que por defecto non está instalado en Ubuntu.

1.3 Conexión dende cliente Linux (ssh) e comando relacionados (transferencia ficheiros: scp, sftp).

Para conectarnos dende o cliente utilizaremos o comando ssh (Security Shell). Na primeira conexión o servidor envía a clave pública para que o cliente poida cifrar a información que envía ao servidor e descifrar a que recibe. É necesario responder *yes* á pregunta para engadir esa clave ao lista de servidores coñecidos.



Para conectarnos só pide o contrasinal. Se non indicamos o nome de usuario entende que conectamos utilizando o mesmo nome de usuario, pero o normal é que a máquina cliente e o servidor non teña os mesmos usuarios creados. Para indicar o nome de usuario co que queremos iniciar sesión podemos poñer o nome do usuario utilizando a arroba ou coa opción -l.

ssh nomeusuario@nomeservidor

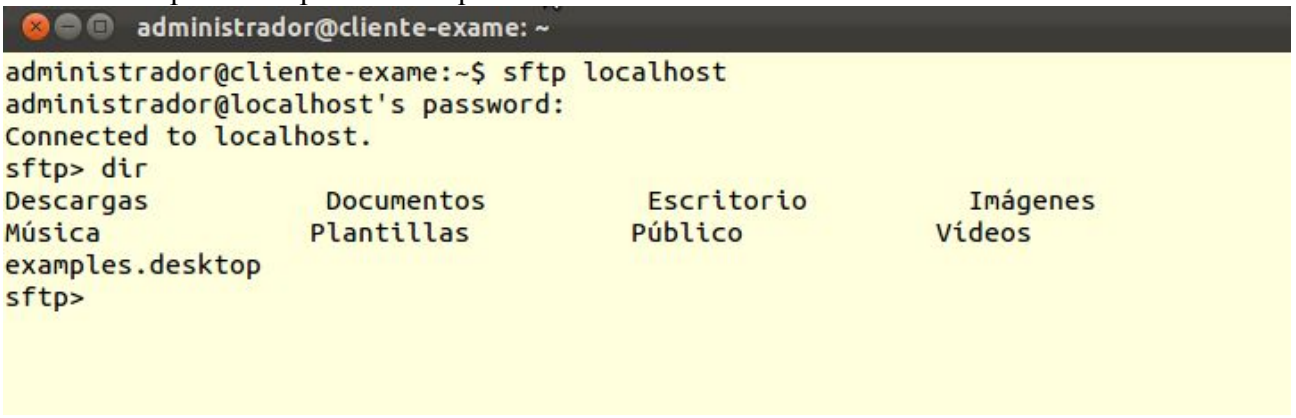
ssh -l usuario nomeservidor

Unha vez establecida a conexión traballamos como nunha sesión de terminal local.

O comando scp permite copiar arquivos entre a máquina cliente e o servidor. O comando abre unha conexión para copiar os arquivos e pecha esa conexión cando remata de copiar os arquivos.

Exemplo de funcionamento:
scp /ruta/local/arquivo.txt usuario**@**servidor_remoto**:**/ruta/servidor/remoto/novo_nome.txt

O comando sftp permite transferir arquivos na maneira de traballar dos clientes FTP, sen pechar a conexión despois de copiar cada arquivo.



1.4 Auto autoautenticarse dende un cliente SSH

http://magicmonster.com/kb/net/ssh/auto_login.html

https://debiantalk.wordpress.com/2015/04/27/debian-8-no-root-login-via-ssh/

Se queremos conectarnos como root mediante SSH é necesario facer algún cambios. Debemos iniciar sesión como root e editar o ficheiro sshd_config:

```
nano /etc/ssh/sshd_config
```

Debemos buscar a liña PermitRootLogin (Ctrl+W). Comentaremos a liña e engadirémolo de novo coa opción yes:

```
# PermitRootLogin prohibit-password
PermitRootLogin yes
```

Agora temos que reiniciar o servizo de ssh e xa poderemos conectarnos como root mediante ssh:

```
/etc/init.d/ssh restart
```

Para configurar a opción de iniciar sesión mediante ssh sen necesidade de poñer o contrasinal debemos conectámonos ao servidor en local ou dende o cliente mediante ssh.

```
cliente$ssh -l  usuario-no-servidor  nome-ou-ip-servidor
```

Xa no servidor, xeramos un par de claves:

```
server$ ssh-keygen -t rsa
```

Deixamos a passphrase en branco. Vai crear un par de arquivos no directorio ~/.ssh. Son as claves: **id_rsa** e **id_rsa.pub.**
Gardamos o contido do arquivo id_rsa.pub no arquivo authorized_keys

```
server$cd  ~/.ssh
server$cat id_rsa.pub >> authorized_keys
```

Cambiámoslle os permisos

```
server$chmod 600 authorized_keys
```

Pechamos a sesión SSH no servidor e creamos o directorio .ssh dentro do directorio fogar do usuario no cliente (pode que xa exista).

```
cliente$cd
cliente$mkdir .ssh
```

Copiamos o arquivo id_rsa do servidor ao cliente:

```
cliente$cd .ssh
cliente$scp nome-usuario@nome-ip-servidor:~/.ssh/id_rsa .
cliente$chmod 600 id_rsa
```

Despois de facer estes cambios, ao conectarnos xa non pedirá contrasinal.

1.5 Conexión dende cliente Windows (PuTTY).

Para conectarnos dende Windows podemos utilizar o programa gratuíto PuTTY (http://www.chiark.greenend.org.uk/~sgtatham/putty/) que faría ven a ser o equivalente ao comando ssh en Linux. Tamén é posible instalar outros programas, por exemplo, os programas PSCP e PSFTP son os equivalentes aos comandos scp e sftp.

Os programas son compatibles coa versións de 32 e 64 bits Windows e dispoñemos dun arquivo de instalación de Windows (.msi) para instalalos todos. Tamén podemos utilizalos descargando cada executable (.exe) por separado. A páxina de descarga do programa é:
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

A primeira vez que executamos un dos programas é necesario instalar a clave do servidor Linux.

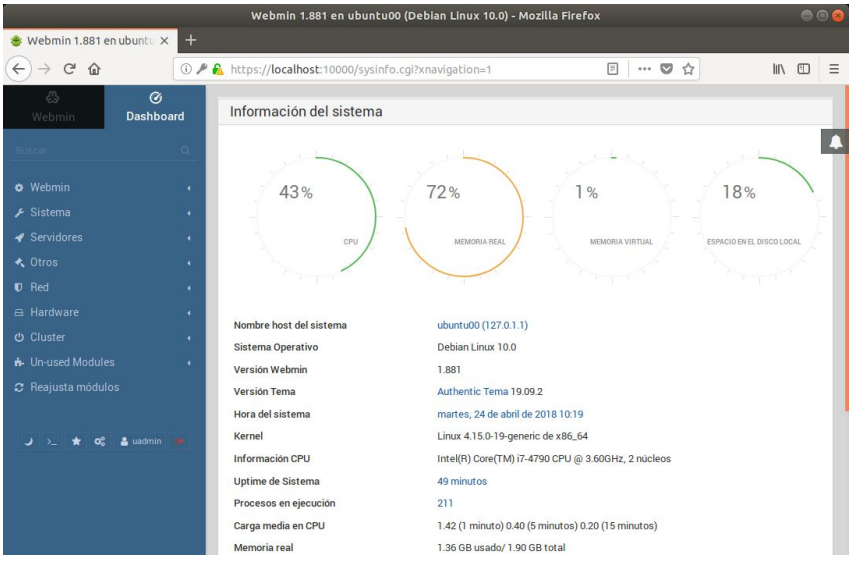
2 Webmin

2.1 Introdución.

Webmin non é un comando senón unha ferramenta que permite administrar un sistema Linux mediante un interfaz web, usando unha conexión segura co protocolo **https**. Pode utilizarse en moitas distribucións de Linux (Ver **distribucións soportadas**) incluíndo Ubuntu, aínda que as distribucións nas que mellor funciona é nas baseadas en Redhat (**Redhat Enterprise Linux**, **CentOS**, **Mandriva**, **Fedora**, etc.).

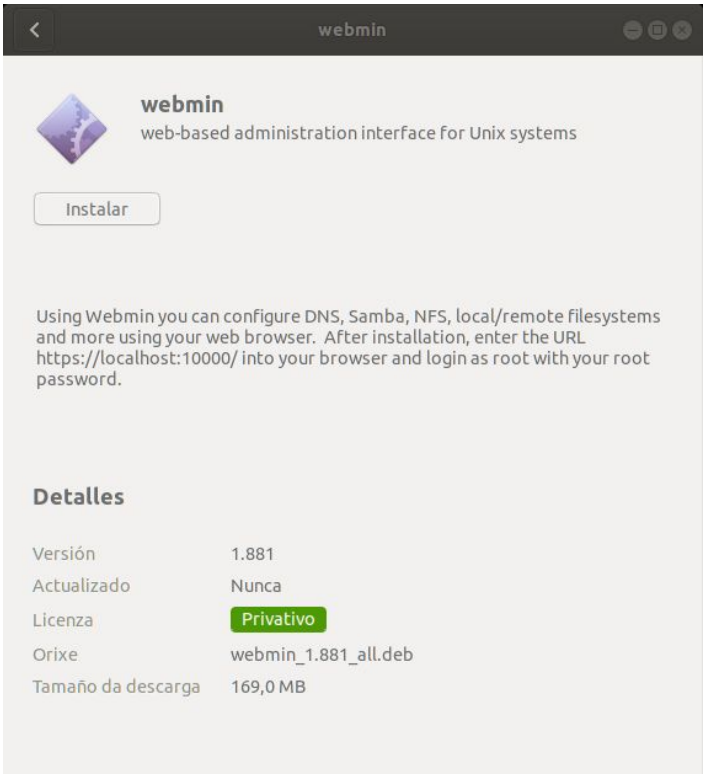
Con esta ferramenta podemos configurar moitos aspectos e servizos do sistema, como os usuarios e grupos, as particións de disco, o servidor apache, un servidor DNS, DHCP, etc. Webmin non está incluído nos repositorios de Ubuntu, así que para instalalo descargarémolo directamente da páxina oficial e instalarémolo co comando **dpkg**.

Unha vez instalado o paquete, poderemos acceder ao webmin dende outro equipo introducindo no navegador a dirección **https://enderezo_ip_servidor_ubuntu:10000**. Para iniciar a sesión en Webmin teremos que introducir o login e clave do usuario administrador creado na instalación do servidor. Tamén podemos iniciar sesión co usuario root pero é necesario que teña contrasinal:



2.2 Instalación.

Comezaremos por ir á páxina web do proxecto (http://www.webmin.com) e descargar a última versión do paquete .deb. Dende a interface gráfica chega con facer dobre clic sobre o arquivo descargado para que se abra coa utilidade de **Instalar Software**. Dende a liña de comandos podemos instalar o paquete con: ***sudo dpkg -i paquete.deb*** (É aconsellable executar **sudo apt update** antes de instalar Webmin para que se configuren ben as dependencias).



En ambos pode que que necesitemos introducir o contrasinal do usuario para confirmar a instalación e terá que ter permisos administrativos. Dende o entorno gráfico as dependencias instálanse automaticamente e dende a liña de comandos será necesario completar a seguinte liña para completar a instalación:

```
sudo apt-get install -f
```

Para conectarnos debemos utilizar o conexións seguras e conectarnos ao porto 10000 (https://ip_máquina_Linux:10000).

No casa das máquinas virtuais de VirtualBox, se a interface de rede funciona en modo NAT necesitaremos redirixir o porto 10000 para conectarnos dende a máquina real. En a interface funciona en modo ponte só necesitamos saber a IP da máquina xa que tódolos porto son directamente accesibles.

3 Escritorio remoto

3.1 Habilitar o escritorio remoto en Ubuntu.

Debemos comezar por compartir o escritorio dende Preferencias do sistema>Compartición ou buscando a palabra clave escritorio dende o botón de actividades:

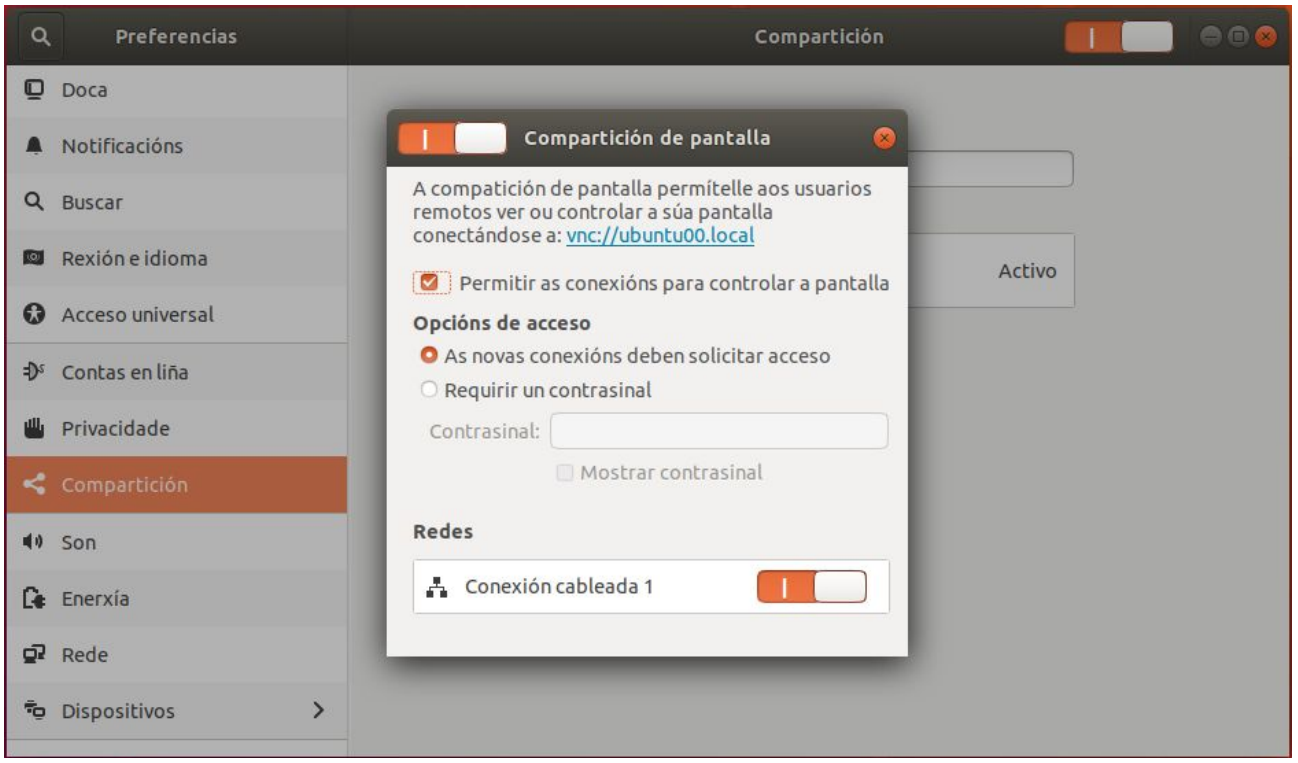
Comezamos por configurar o nome co que se verá o equipo no rede:

Despois de desbloquear a Compartición dende a barra de título podemos entrar na configuración. As opción dispoñibles son:

- Permitir que as conexións poidan controlar a pantalla, en caso de activala só poderán ver o que realiza o usuario local.
- Establecer un contrasinal para poder conectarse remotamente ou indicar que se solicite acceso. Se marcamos a opción “As novas conexións deben solicitar acceso” o que sucede é

que pedirá unha confirmación cada vez que se conecta, polo que alguén debe estar diante do ordenador que funcione como servidor para confirmar cada conexión (non é práctico).

- Tamén podemos indicar mediante que conexión de rede é posible conectarse ao escritorio remoto.



Para compartir o escritorio utiliza un servidor VNC (protocolo RFB).

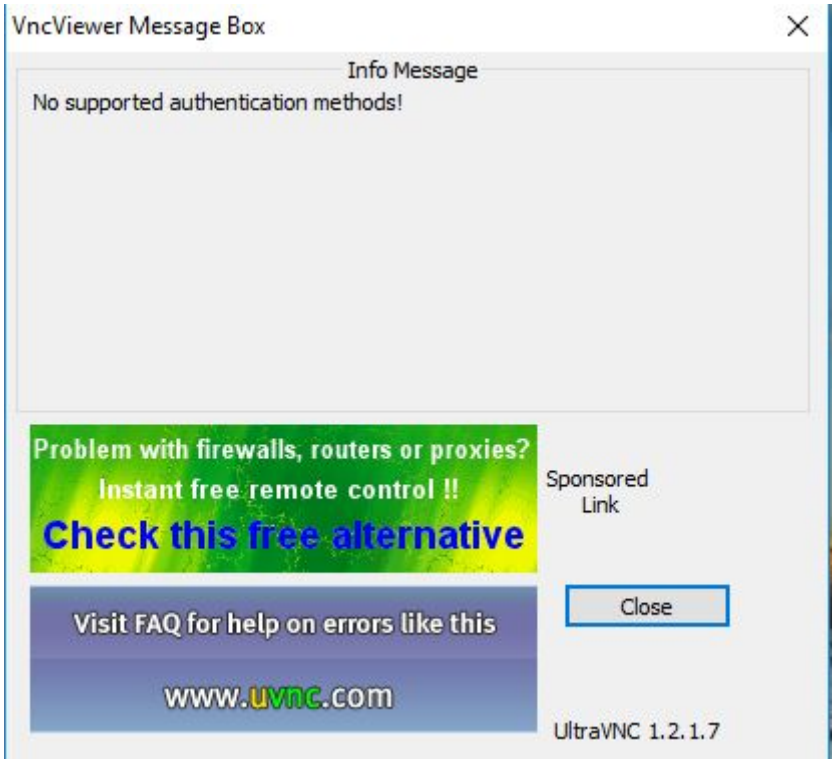
3.2 Conexión dende un cliente Linux.

Para conectarnos dende un cliente Linux podes utilizar o cliente que ven instalado en Ubuntu (Remmina) ou calquera outro (por exemplo, o visualizador de escritorios remoto Vinagre) que permita establecer conexións mediante VNC.

3.3 Conexión dende un cliente Windows. UltraVNC

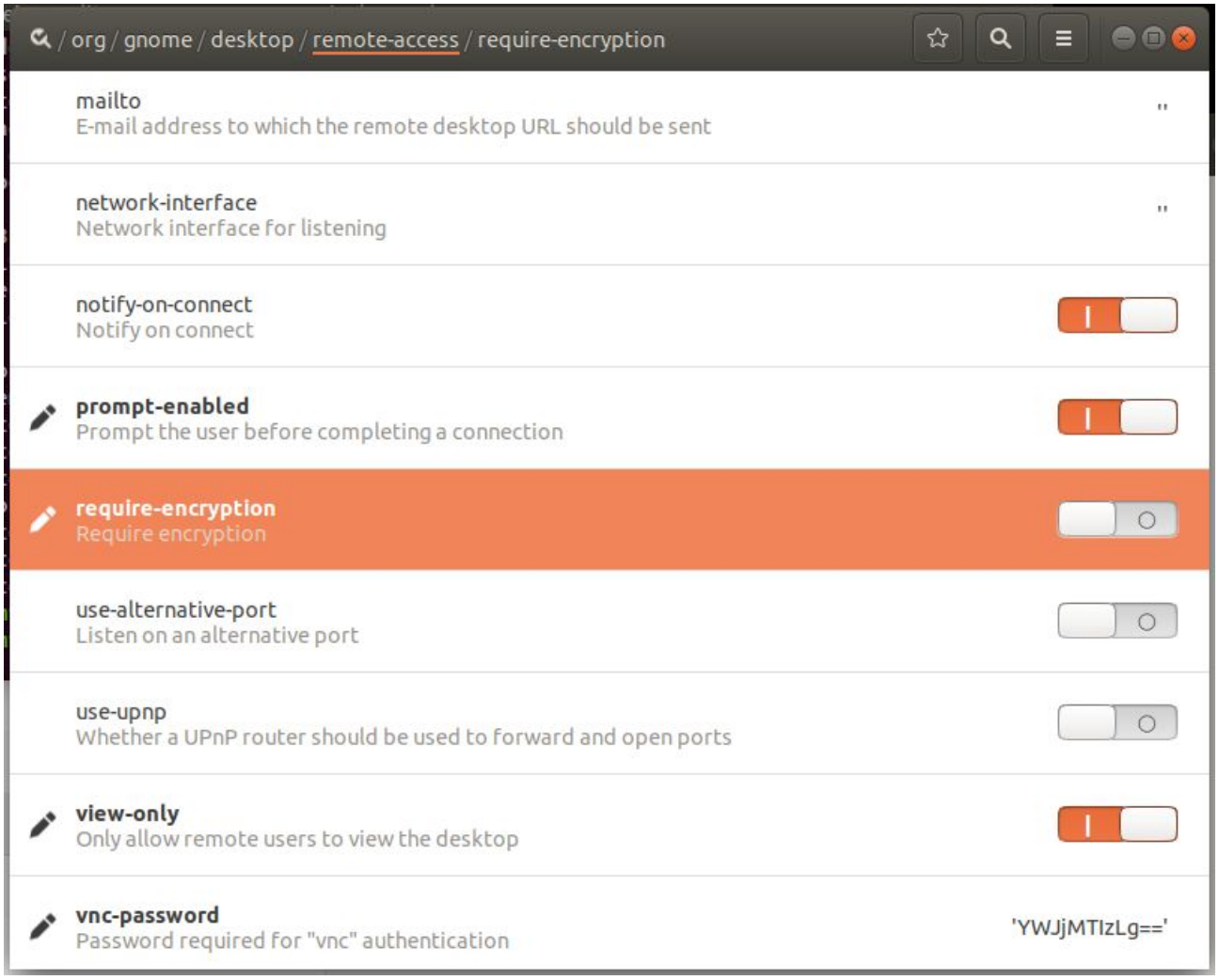
Dende Windows podemos utilizar clientes que admitan o protocolo RFB, máis coñecido por VNC (Virtual Network Computing), por exemplo ultraVNC.

Se ao conectarnos aparece o erro No supported authentication methos! debemos facer cambios na configuración de Ubuntu:



Seguimos as instrucións deste artigo para cambiar a configuración. Instalamos a utilidade dconf-editor (**sudo apt install dconf-editor**) e despois de abrila. O **editor dconf** debe executarse como o usuario que comparte o escritorio polo que non debemos poñer **sudo** diante do comando **dconf-editor**. Tamén serve buscar a aplicación dende botón **Actividades.** Ao abrir o editor veremos unha mensaxe ben clariña

Imos a **org>gnome>desktop>remote-access** en desactivamos a opción **require-encryption**.



Agora xa poderemos acceder dende Windows con UltraVnc Viewer.

Ligazóns

http://informatica.iessanclemente.net/manuais/index.php/Ferramentas_de_administraci%C3%B3n_remota
http://es.wikipedia.org/wiki/Openssh