Procedimiento para configurar jump server

¿Cómo configurar un jump server?

A continuación, desarrollaremos el script con los pasos a seguir para la configuración de un jump server para entrar directamente a los dispositivos y extraer sus configuraciones para el módulo opConfig:

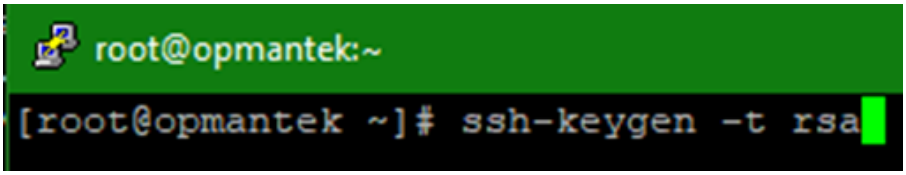
Primero, asegúrate que en el jump server, estén habilitadas las siguientes opciones en el archivo **/etc/ssh/sshd_config**

```
RSAAuthentication yes
PubkeyAuthentication yes
```

Así como el comando terminal legth 0. Debe de mostrarte algo como etsto

```
RP/0/RP0/CPU0:CUA-CORE-R01#terminal length 0
 Mon Mar
12 18:08:27.389 CST
```

Entrar al servidor de NMIS con las credenciales correspondientes. Una vez dentro, generar la llave para el IdentityFile con el comando ssh-keygen -t rsa



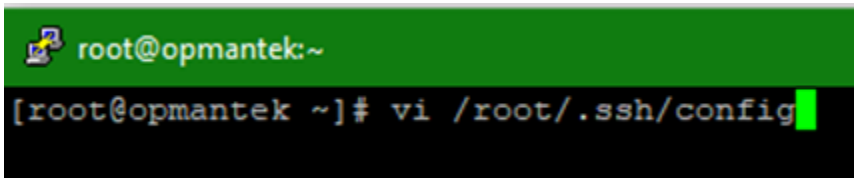
Identificar la ruta de la llave:

```
cd /root/.ssh/
```

Podemos ver el contenido con

```
cat ~/.ssh/id_rsa.pub
```

Ya una vez creada la llave, generamos el archivo config en /root/.ssh/



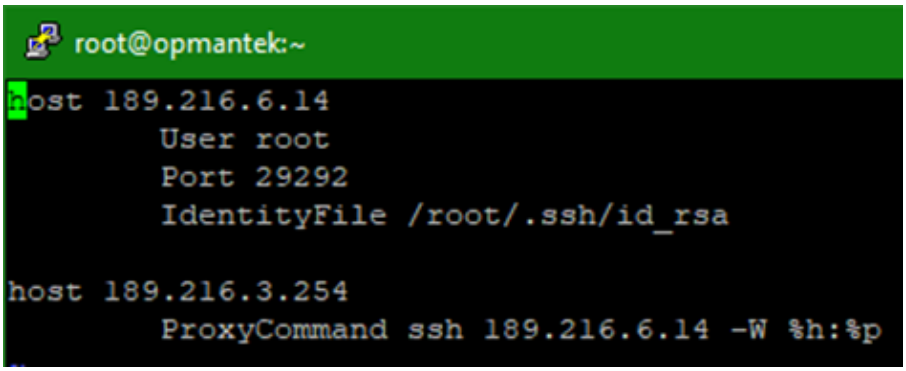
Ya que estemos en la edición de este, agregamos:

```
host <jump box IP address>
        User <jump box user name>
        Port #####
        IdentityFile /root/.ssh/key
host <node IP address>
        ProxyCommand ssh <jump box IP address> -W %h:%p
```

En el cual:

- **<jump box IP address>** es la dirección IP del servidor pivote
- **<jump box user name>** el usuario SSH para acceder al pivote
- **Port ####** es el número del puerto del servidor pivote
- **/root/.ssh/key** ruta de la llave generada más adelante
- **<node IP address>** dirección IP del nodo a probar

Ejemplo:



Ahora, copiamos la llave al servidor pivote con alguno de los siguientes comandos:

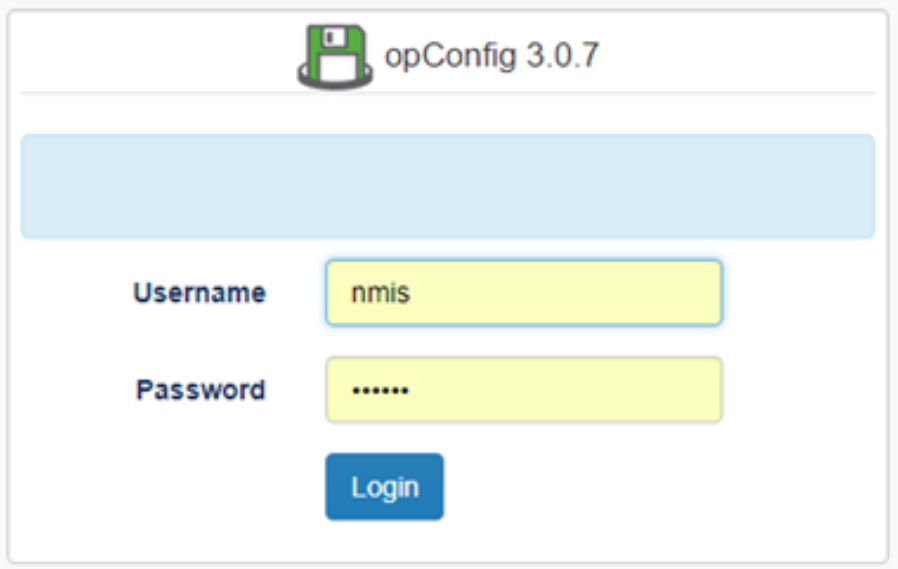
```
ssh-copy-id -i /home/user_name/.ssh/id_rsa.pub hostname

ssh-copy-id -i your_public_key user@host
```

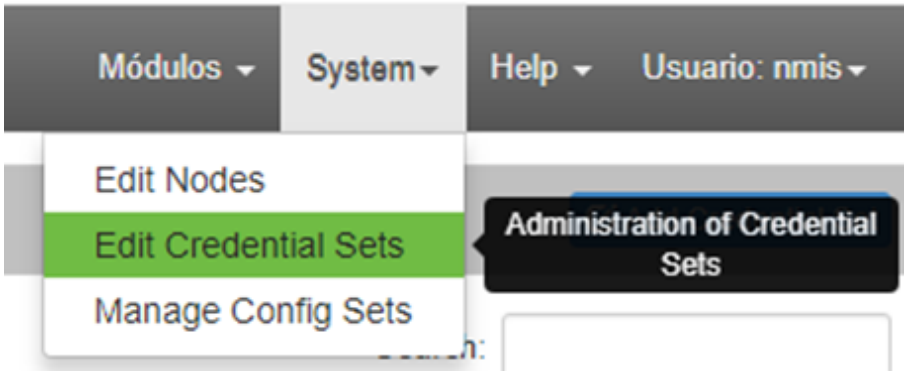
Podemos probar si funciona, entrando vía SSH a un nodo directamente y ahí nos va a pedir las credenciales de acceso **para el nodo**, no para el jump server.

```
ssh USUARIOSSH@IPDELNODO
```

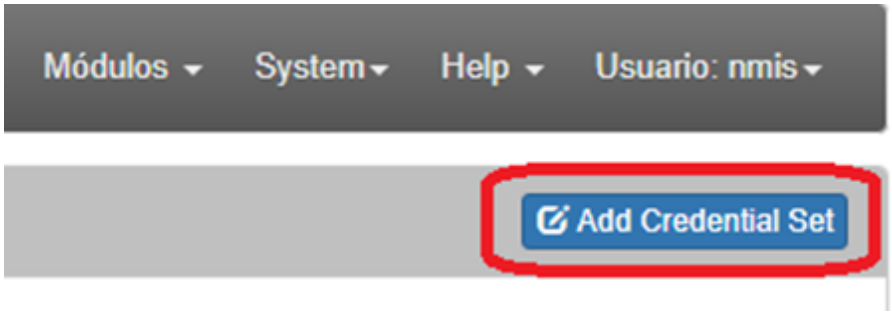
Ahora, en el servidor de opConfig, accedemos con las credenciales correspondientes.



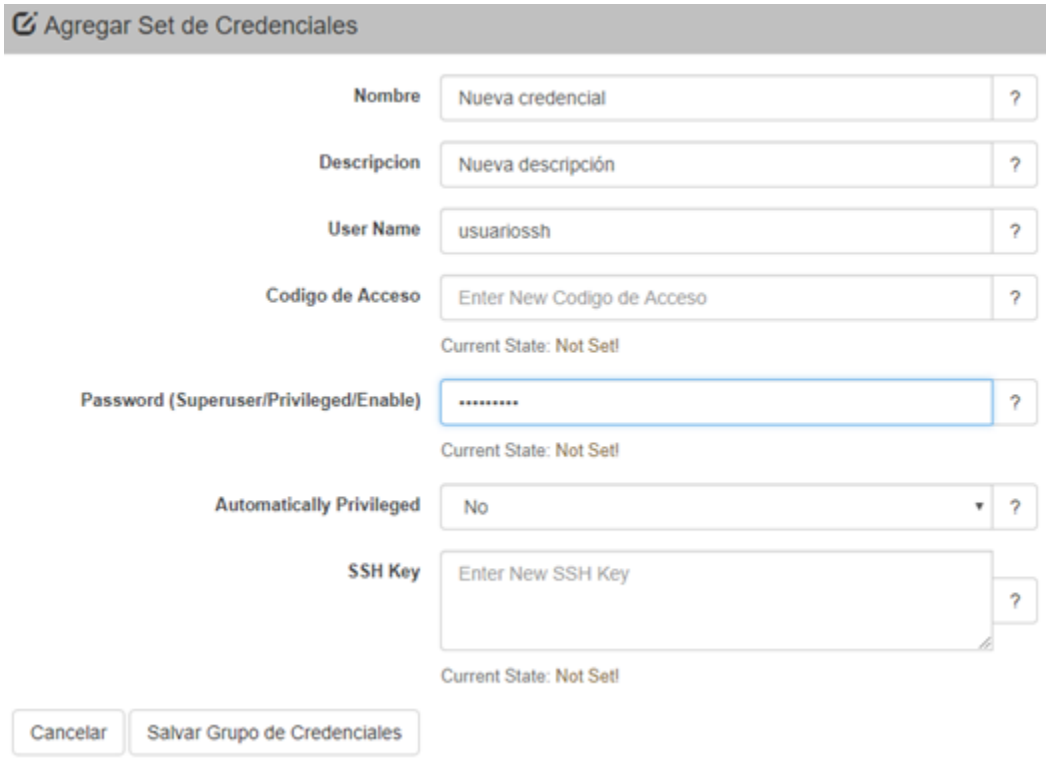
Vamos a **System > Edit Credential Sets**



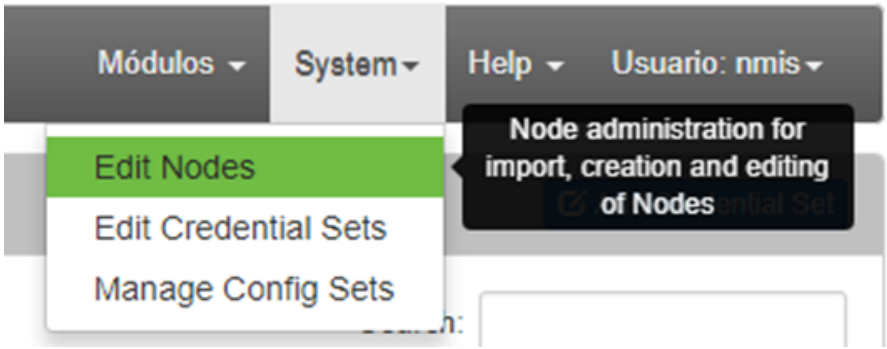
Damos clic en **Add Credential Set**



Agregamos la nueva credencial SSH para los dispositivos y damos clic en **Salvar Grupo de Credenciales**



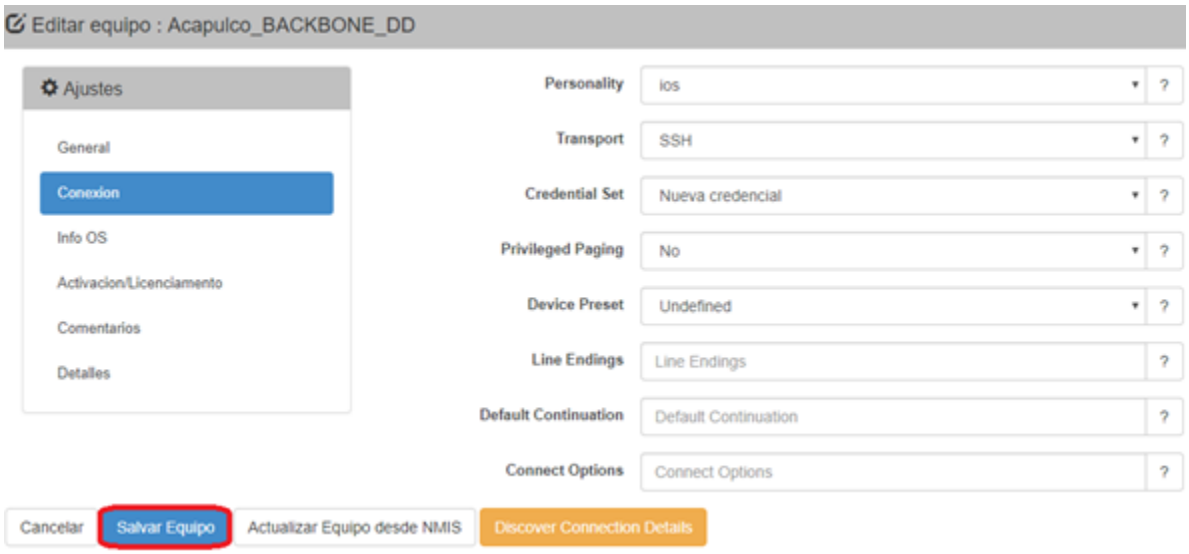
Ahora vamos a **System > Edit Nodes**



Seleccionamos el nodo al que le vamos a asignar la credencial que acabamos de crear.

En **Conexión**, asignamos la nueva credencial SSH como se ve en la imagen. Damos clic en **Salvar Equipo**.

Nota: Es recomendable hacer un update del nodo, haciendo clic en **Actualizar Equipo desde NMIS** antes de asignar la credencial

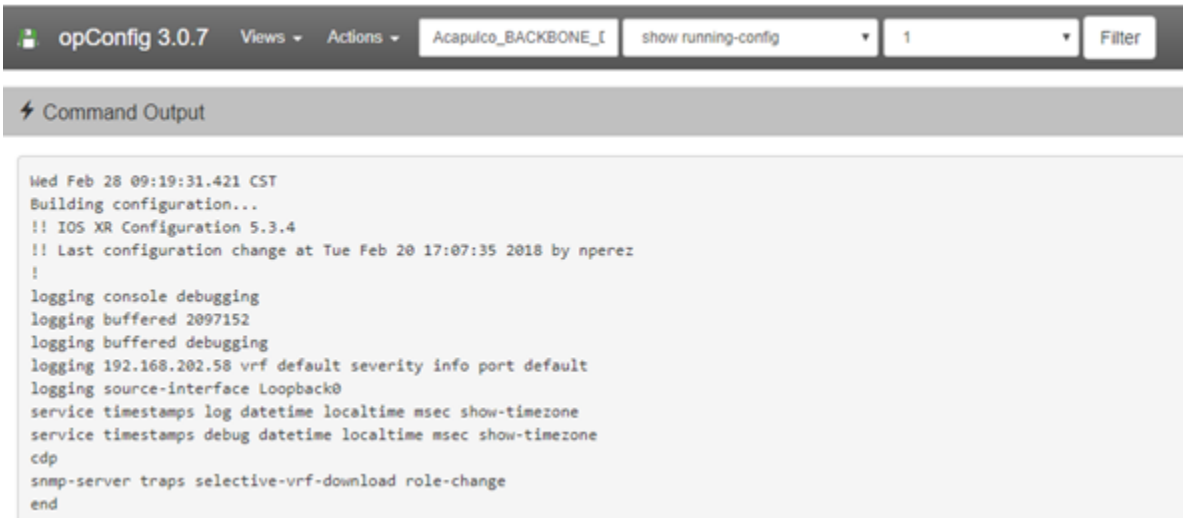


Ahora, en consola, ejecutamos los siguientes comandos:

```
[root@opmantek ~]# /usr/local/omk/bin/opconfig-cli.exe act=discover node=NOMBREDELNODO

[root@opmantek ~]# /usr/local/omk/bin/opconfig-cli.exe act=run_command_sets node=NOMBREDELNODO
```

Con esto, quedarán configurados los nodos con las credenciales SSH correspondientes y ya podremos ver outputs en opConfig.



https://www.digitalocean.com/community/tutorials/how-to-configure-ssh-key-based-authentication-on-a-linux-server