

Trucos con Openssh en IBM i

+TRUCOS Y USO AVANZADO DE SSH
DESDE Y HACIA IBM I

diego@esselware.com.mx



Agenda

- + Por qué usar SSH en IBM i?
- + Cómo configuro OpenSSH en IBM i?
- + Cómo configuro OpenSSH en IBM i?
- + Qué es lo primero que debo hacer? (1)
- + Qué es lo primero que debo hacer? (2)
- + Generando llaves públicas y privadas
 - Conectándonos sin contraseña
- + Creando túneles - Generalidades
- + Túnel Local
- + Túnel Remoto
- + Túnel Dinámico
- + Combinando túneles
- + Montando un directorio del IFS con SSHFS
- + Usando un archivo de configuración
- + Enviando y comprimiendo archivos a la vez
- + Transfiriendo archivos por SFTP en modo batch y con usuario/password
- + Ejecutar comandos de forma remota
- + Norton Commander en IBM i
- + Sincronizar directorio del IFS con otro servidor
- + Saltando entre múltiples servidores
- + Modificar los parámetros de Port Forwarding mientras estamos conectados
- + Crear un túnel que se mantenga conectado de forma permanente
- + Ejecutar varias tareas a la vez en la misma sesión
- + Conectándonos con SSH para usar Access Client Solutions
- + Conocimientos básicos recomendados para sacar el máximo provecho al SSH

Por qué usar SSH en IBM i?

- +La comunicación es segura y podemos acceder a todo el equipo (o la red) desde un único puerto)
- +Nos permite crear túneles (Locales, Remotos, Dinámicos)
- +Nos permite intercambiar archivos (sftp/scp/ssh+cat)
- +Podemos sincronizar archivos (RSync)
- +Soporte compresión (parámetro -C)
- +Permite usar PASE en toda su potencia (QP2TERM es una pseudo-terminal)
- +Es un básico para el uso de herramientas de desarrollo como VSCode
- +Es casi INDISPENSABLE para usar herramientas Open Source portadas de Linux

Cómo configuro OpenSSH en IBM i?

- +Necesito instalar 5733SC1, que viene con la media de IBM i
- +Necesito ajustar algunos valores del archivo
/QOpenSys/QIBM/UserData/SC1/OpenSSH/etc/sshd_config
- +Agregar al final del archivo (podemos Filezilla para editarlo en nuestra PC)
 - +UseDNS no
 - +PermitRootLogin yes
 - +ibmpaseforienv PASE_USRGRP_LIMITED=N
 - +ibmpaseforishell=/QOpenSys/pkgs/bin/bash # Opcional #
- +Podemos cambiar el puerto (recomendable) con el parámetro Port.
 - +Ejemplo:
Port 8059

Qué es lo primero que debo hacer?

(1)

- +Hago que arranque en automático durante el IPL
 - +CHGTCPSVR SVRSPCVAL(*SSHD) AUTOSTART(*YES)
- +Creo el directorio de usuario en el IFS con los permisos sólo para mi perfil:
 - +CALL QP2TERM
 - +mkdir /home/USUARIO
 - +chown usuario /home/USUARIO
 - +chmod 700 /home/USUARIO
 - +ssh-keygen <Enter a todas la preguntas que hará>
 - +Salgo con F3

Qué es lo primero que debo hacer?

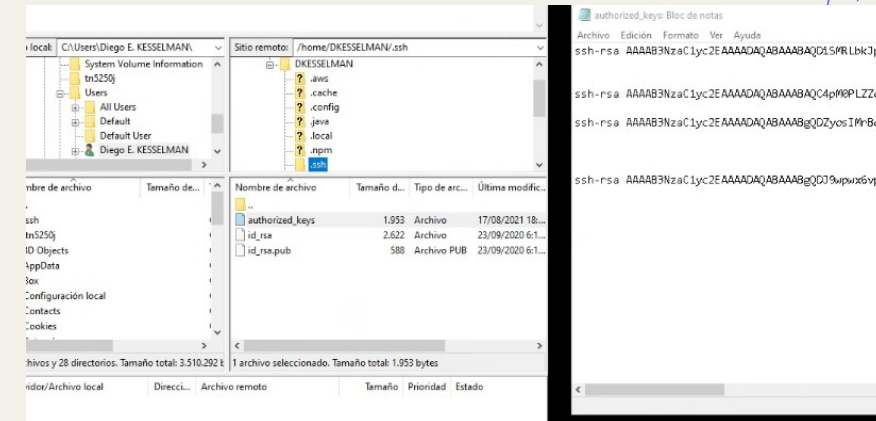
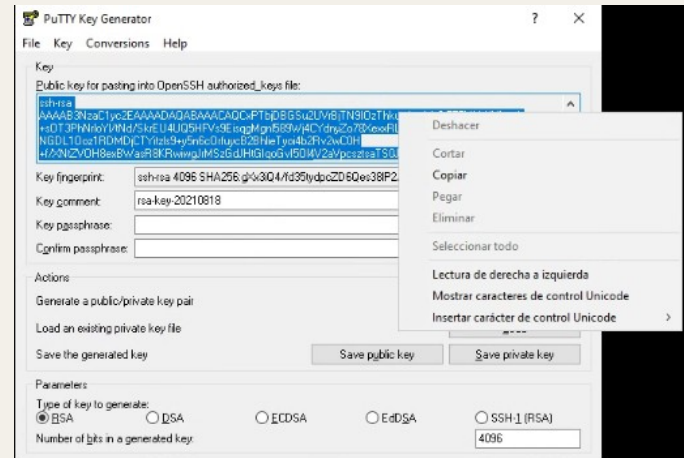
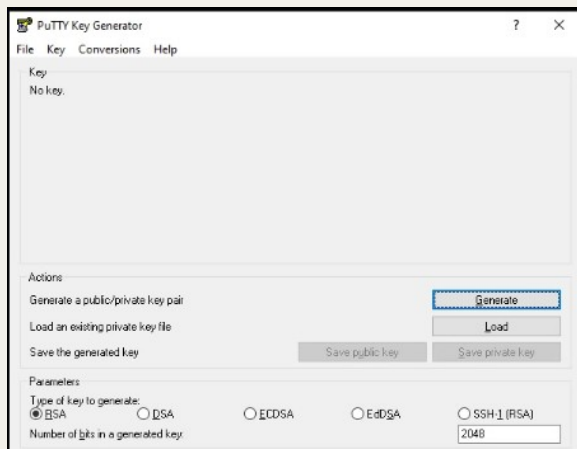
(2)

- +Creo mis llaves públicas y privadas usando PuTTYGen o ssh-keygen en mi PC y subo la llave privada al servidor para poder firmarme sin password. Puedo usar ssh-copy-id
- +Instalo las herramientas Open Source, preferentemente usando Access Client Solutions para que instale BASH (se instala por default). También se recomienda instalar TMUX, GZIP, PIGZ, Python3, Python3-pip, RSYNC, LFTP, MC, VIM, JOE, NANO, db2util
- +Creo mi archivo de perfil /home/USUARIO/.bash_profile (Puedo subirlo con Filezilla o usar alguno de los editores como "vi" o "nano")
 - +PATH=\$PATH:/QOpenSys/pkgs/bin
 - +TERM=aixterm
 - +export QIBM_MULTI_THREADED='Y'
 - +export LC_CTYPE=ES_MX
 - +export PS1="\u@\[\\e[32m\\]\H\[\\e[m\\]:\w>"
 - +export TERM
 - +export PATH
- +Reinicio el servicio de SSHD
 - +ENDTCPSVR *SSHD
 - +STRTCPSSVR *SSHD

Generando llaves públicas y privadas - Conectándonos sin contraseña

+ Para conectarnos desde Windows podemos usar el cliente de OpenSSH desde una sesión de CMD:
ssh-keygen <Enter a cada pregunta>

- + Generará un archivo id_rsa.pub en el directorio c:\Users\MIUSUARIO\.ssh\
- + Debo abrir con Notepad y copiar el contenido en el portapapeles.
- + Me conecto por Filezilla al servidor, entro en /home/UserIBMi/.ssh/ y edito el archivo authorized_keys. Debo agregar el contenido al final del archivo
- + Si tengo PuTTY debo usar PuTTYGen (viene en la versión completa del producto) y copiar la clave de OpenSSH que aparece en la ventana
- + En la sesión de Putty debemos indicar la clave privada y el usuario para conectarnos sin contraseña
- + [Configuring the PuTTY Secure Shell \(SSH\) Client to Use Public-Key Authentication](#)



Creando túneles - Generalidades

- +Un túnel me permite tomar un puerto remoto de algún servicio de TCP que sea accesible desde el servidor.
 - +Ejemplo: Puedo hacer que se vea el puerto 80 de un servidor remoto como si corriera en mi PC y ver la página haciendo `http://localhost`
- +Existen distintos tipos de túneles
 - +Locales: "Tomo" un puerto remoto del servidor a mi equipo (PC o Servidor)
 - +Remotos: "Envío" un puerto desde mi equipo al servidor remoto
 - +Dinámicos: Creo un proxy SOCKS5 que me permite ver múltiples puertos a la vez
- +Para crear túneles necesito los siguientes valores en `sshd_config` (pueden estar comentados con un `#`)
 - +`PermitTunnel` yes
 - +`AllowTcpForwarding` yes

Túnel Local

+El más común y fácil de usar:

+Ejemplo: `ssh -L 8080:127.0.0.1:80 usuario@IPSERVER`

+`-L` declara un túnel local

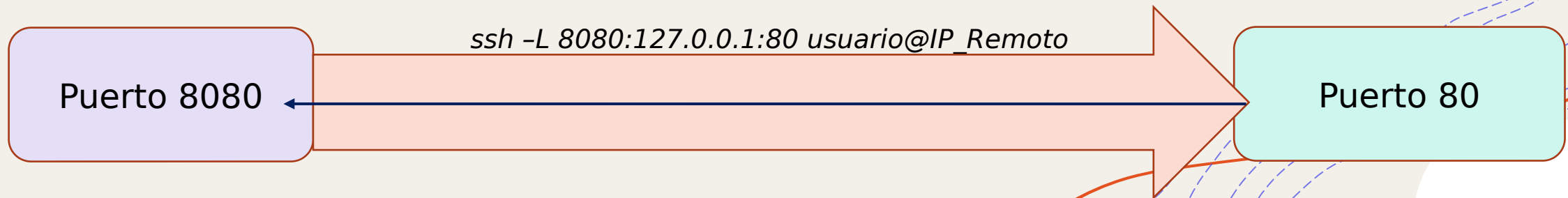
+8080 es el puerto en que veré el túnel

+12.0.0.1 es la IP vista desde el servidor remoto. En este caso es el propio servidor, pero podría ser otro en la misma red

+80 es el puerto del servidor remoto

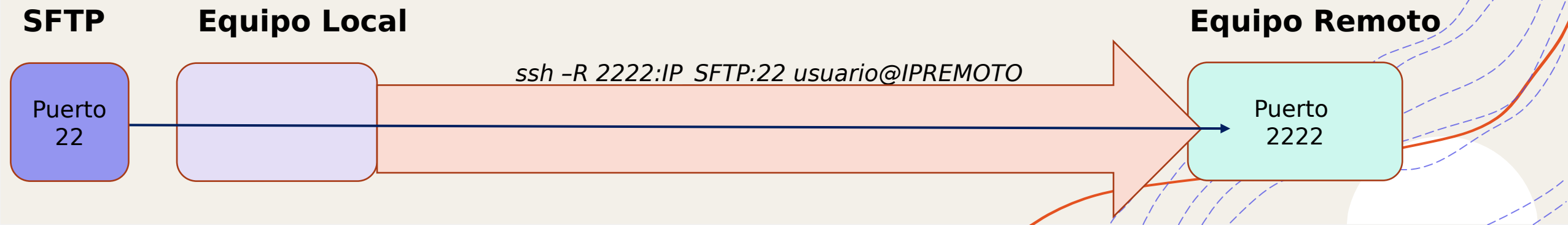
Equipo Local

Equipo Remoto



Túnel Remoto

- +A diferencia de un túnel local, entrego un puerto al servidor destino
 - +Ejemplo: `ssh -R 2222:IP_SFTP:22 usuario@IPREMOTO`
 - +`-R` declara un túnel Remoto
 - +2222 es el puerto en que veré el túnel
 - +`IP_SFTP` es la IP vista desde mi equipo. En este caso es un servidor de SFTP
 - +22 es el puerto del servidor de SFTP



Túnel Dinámico

- +Este tipo de túnel permite acceder a cualquier puerto visible por el equipo remoto
 - +Ejemplo: `ssh -D 1080 usuario@IPREMOTO`
 - +D declara un túnel Remoto
 - +1080es el puerto en que veré el túnel
- +Para hacer uso de este tipo de túnel necesito contar con aplicaciones que soporten proxy SOCKS5, como la mayoría de los navegadores, PuTTY y TN5250]
- +Para los programas que NO soporten proxy SOCKS5 puedo usar algunas aplicaciones que "proxifican" la comunicación, como Proxifier, WideCap, Proxychains, SSHUTTLE, etc
- +Desde con un túnel dinámico puedo ver TODA la red que ve el servidor remoto

Equipo Local

Puerto
1080

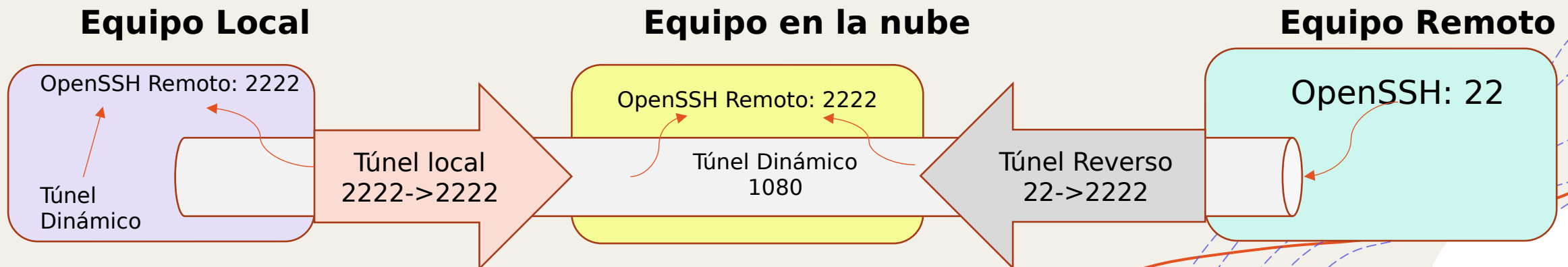
Ejemplo: `ssh -D 1080 usuario@IPREMOTO`

Equipo Remoto

Toda la red

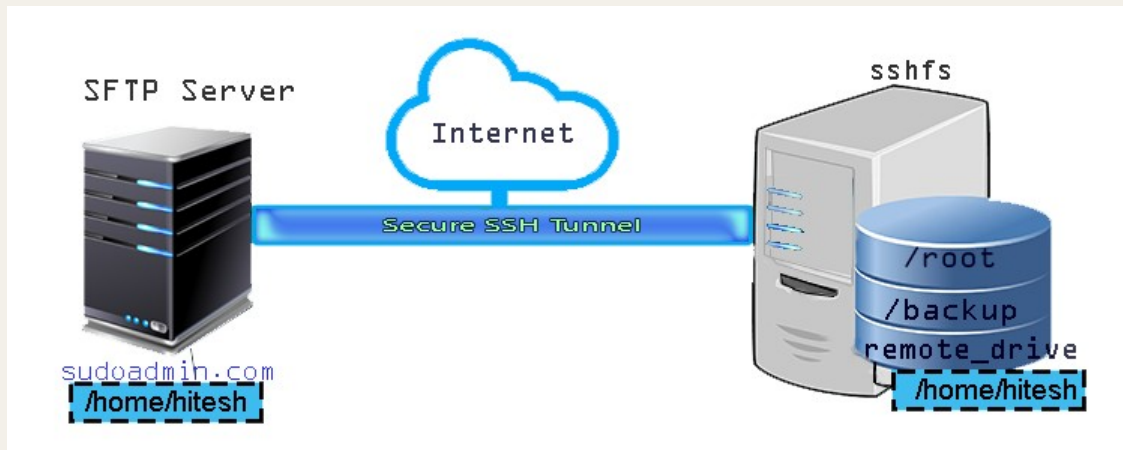
Combinando túneles

- +Podemos combinar diferentes tipos de túneles para cumplir con nuestros objetivos de conectividad
- +Podríamos usar un túnel reverso para conectar nuestro IBM i a un servidor en la nube y publicar el puerto SSH y luego desde nuestro equipo conectarnos al mismo equipo y desde ahí llegar al IBM i
- +El escenario sería así
 1. Equipo Remoto - Túnel Reverso: `ssh -R 2222:localhost:22 usuario_cloud@ip_cloud`
 2. Equipo Local - Túnel Local: `ssh -L 2222:localhost:2222 usuario_cloud@ip_cloud`
 3. Equipo Local - Túnel Dinámico: `ssh -D 1080 -p 2222 usuario_remoto@localhost`



Montando un directorio del IFS con SSHFS

- +El SSHFS es un file system que permite montar un directorio del IFS de forma similar al NFS, pero con encriptación y a mayor velocidad.
- +En Windows podemos usar SFTP Drive 2 de nSoftware (hay versión gratuita).



The 'Edit Drive' window in SFTP Drive 2 shows the following configuration:

- Drive Name: My Drive
- Drive Letter: Z:
- Remote Host: 10.0.1.63
- Remote Port: 22
- Authentication Type: Password
- Username: test
- Password: (masked with asterisks)
- Private Key (.ppk): (empty field with a 'Browse...' button)
- Host Key Fingerprint: (empty field)
- Remote Folder: /
- Root Folder on Server: ☒ Server root ☐ User's home folder ☐ Specified folder:
- ☐ Read-only Mode
- ☒ Open Remote Folder On Connect: (empty field)
- Test SSH Connection** (button highlighted with a red box)
- OK and Cancel buttons at the bottom right.

Usando un archivo de configuración

+En entornos Linux y en el mismo IBM i podemos usar un archivo de configuración para usar alias y simplificar la forma de conectarnos

+Ejemplo:

+Host ibmi01

+ HostName 192.168.50.225
 User dkesselman

+ Port 2222
 IdentityFile /home/diego/.ssh/id_rsa

+Lo usamos de la siguiente forma:

```
ssh ibmi01
```

Enviando y comprimiendo archivos a la vez

- + El comando SSH combinado con GZIP permite comprimir mientras enviamos. Esto puede ser muy útil para enviar un SAVF a un repositorio de respaldos, evitando comprimirlo antes y ocupar espacio adicional:

```
cat /QSYS.LIB/MYLIB.LIB/MYSAVF.FILE | gzip -c | ssh IPServerBKP "cat > MYSAVF.gzip"
```

Transfiriendo archivos por SFTP en modo batch y con usuario/password

- + Enviar archivos en modo bath con SFTP o SCP requiere del uso de claves públicas y privadas para evitar que pregunte por usuario y password.
- + La herramienta Open Source LFTP (se instala con YUM) permite usar usuario/password para ejecutar comandos de SFTP:

```
lftp sftp://usuario:password@IPServer -e "put  
archivo.ext; bye"
```

Ejecutar comandos de forma remota

- + Puedo ejecutar comandos en el equipo remoto y guardar el resultado en mi equipo local.
- + El comando "system" de PASE permite ejecutar comandos CL.

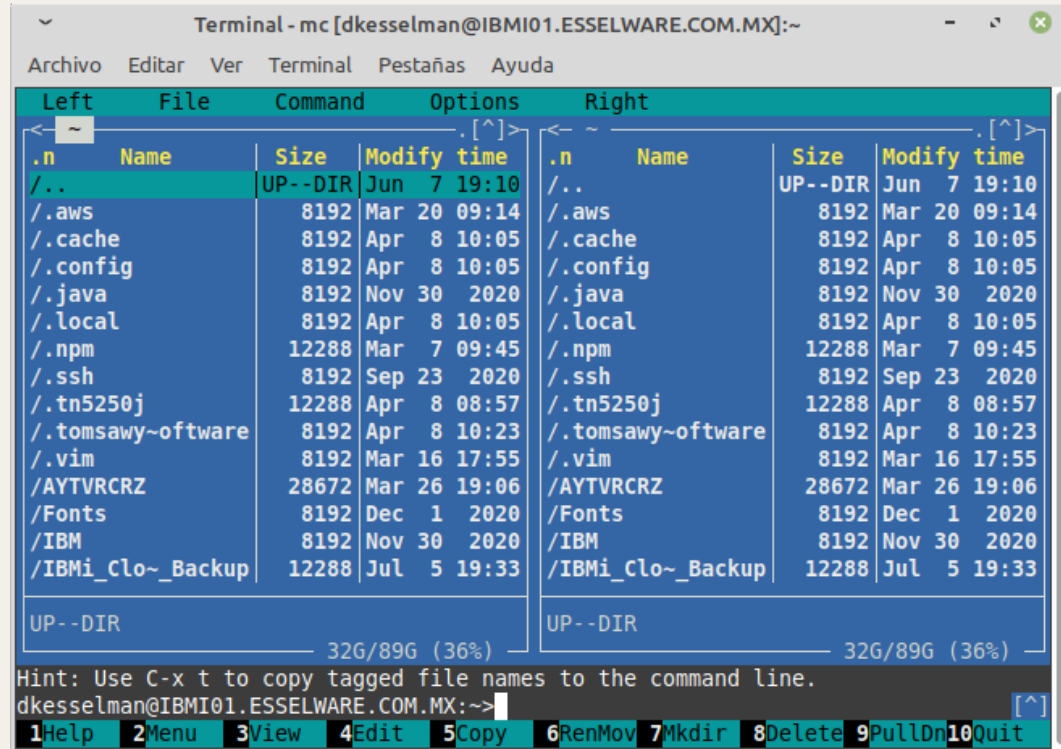
```
ssh ibmi01 -T "db2util 'SELECT JOB_NAME, AUTHORIZATION_NAME,  
ELAPSED_TOTAL_DISK_IO_COUNT, ELAPSED_CPU_PERCENTAGE FROM  
TABLE(QSYS2.ACTIVE_JOB_INFO()) X ORDER BY ELAPSED_TOTAL_DISK_IO_COUNT DESC FETCH  
FIRST 10 ROWS ONLY'" > C:\MyReports\WrkActJob_top10.txt
```

```
ssh ibmi01 -T "system WRKACTJOB"
```

- + **Nota:** db2util debe ser instalado usando YUM para poder ejecutar este ejemplo

Norton Commander en IBM i

- +Existe el programa Midnight Commander, que es similar al Norton Commander, pero versión IBM i.
- +Comando: mc



Sincronizar directorio del IFS con otro servidor

- +El comando **RSYNC** permite sincronizar directorios dentro de un mismo equipo o con un equipo diferente, pero sólo transfiere los archivos nuevos o con una fecha de actualización más reciente.
- +Esto puede ser muy útil si tenemos archivos que se acumulan en el IFS, como facturas electrónicas, archivos PDFs, etc.
- +Ejemplo:
`rsync -zha --max-size='20480k' /home/dkesselman/Documentos/*`
ServidorRemoto:/home/dkesselman/Documentos/



Saltando entre múltiples servidores

+Si necesito saltar entre varios servidores hasta llegar a mi destino puedo usar la función ProxyJump (-J) :

+Ejemplo:

```
ssh -J usuariosvr1@IPSvr1,usuariosvr2@IPSvr2 usuariosvr3@IPSvr3
```

Modificar los parámetros de Port Forwarding mientras estamos conectados

- + Presionando de forma rápida ~C podemos modificar la sesión en curso y agregar más túneles.
- + Con la opción -h podemos ver la ayuda:

```
ssh> -h
Commands:
  -L[bind_address:]port:host:hostport  Request local forward
  -R[bind_address:]port:host:hostport  Request remote forward
  -D[bind_address:]port                Request dynamic forward
  -KL[bind_address:]port               Cancel local forward
  -KR[bind_address:]port               Cancel remote forward
  -KD[bind_address:]port               Cancel dynamic forward
```

Crear un túnel que se mantenga conectado de forma permanente

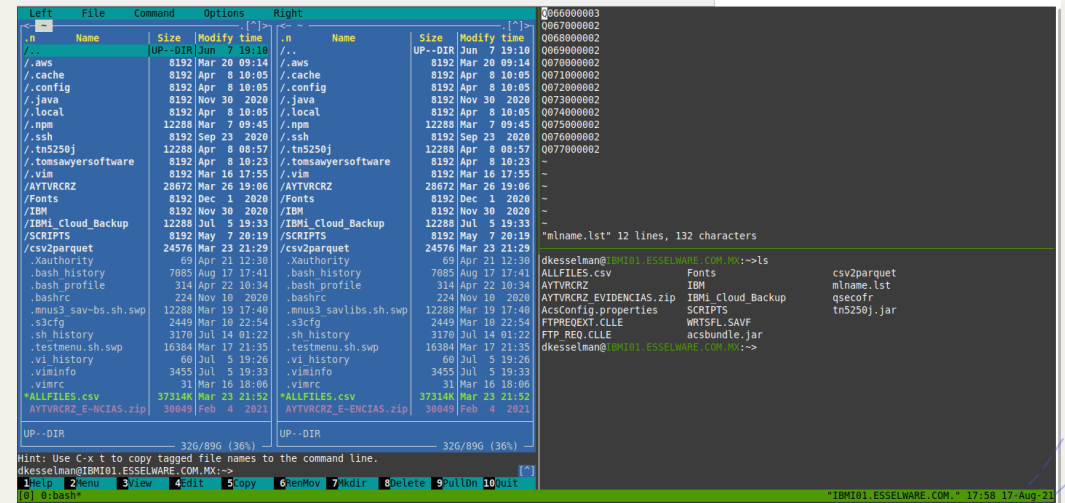
- +En Linux e IBM i podemos usar el comando "autossh" para establecer una conexión que no se desconecte.
- +Algunos servidores cortan la comunicación pero no dan aviso a nuestro equipo y sólo deja de pasar información. El comando autossh envía paquetes a intervalos regulares para evitar esto y reintenta en caso de caídas.
- +Es ideal para establecer conexiones de túneles persistentes usando llaves público/privadas
- +La opción -M indica el puerto de monitoreo
- +La opción -f hace que ejecute en 2do plano
- +Ejemplo:

```
autossh -M 10081 -o "ServerAliveInterval 30" -o "ServerAliveCountMax 3" -L  
2222:localhost:22 usuariocloud@servercloud -f
```

Ejecutar varias tareas a la vez en la misma sesión

- + El comando TMUX (se instala con YUM) permite multiplexar la terminal.
- + También permite compartir la sesión de SSH con otro usuario y trabajar en un archivo de forma simultánea.
- + Para usarlo sólo necesitamos ejecutar "tmux":
 - + CTRL+B % -> Divide la pantalla horizontalmente
 - + CTRL+B " -> Divide la pantalla de forma vertical
 - + CTRL+Flecha de cursor -> Cambia de panel según la flecha

+ Para más información pueden ir a la página:
<https://tmuxcheatsheet.com/>



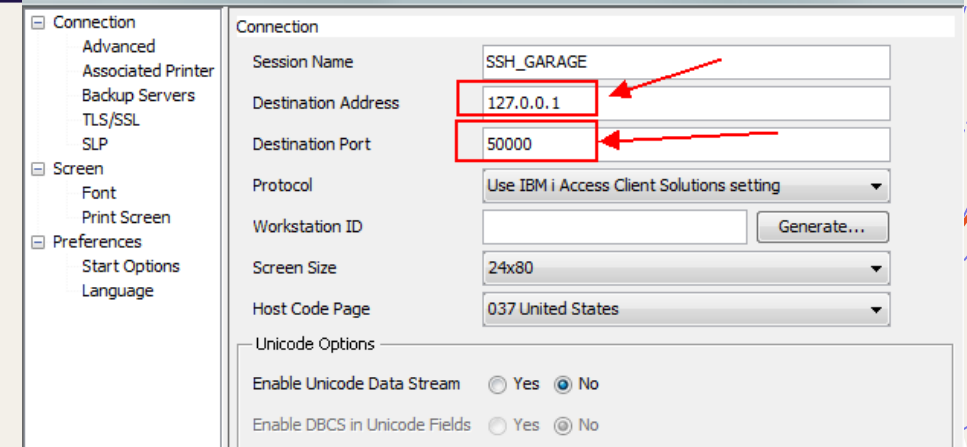
The screenshot displays a tmux terminal window with a split view. The left pane shows a file listing for the directory /IBM1_Cloud_Backup, including files like .aws, .cache, .config, .java, .local, .npm, .ssh, .tn5250j, .tomsawyersoftware, .vim, .ATTURCRZ, .Fonts, .IBM, .IBMi_Cloud_Backup, .SCRIPTS, .csv2parquet, .xauthority, .bash_history, .bash_profile, .bashrc, .mmus3_sav-bs.sh.swp, .s3cfg, .sh_history, .testmenu.sh.swp, .vi_history, .viminfo, .vimrc, .ALLFILES.csv, and .ATTURCRZ_E-NCIAS.zip. The right pane shows a command prompt where the user has executed 'ls' and is viewing the output of 'ls -la', which lists files and directories including ALLFILES.csv, FONT, csv2parquet, mname.lst, qsecofr, tn5250j.jar, and WRTSFL.SAVF. The terminal status bar at the bottom indicates the current directory is /IBM101.ESSLMWARE.COM.MX and the time is 17:58 on 17-Aug-21.

Conectándonos con SSH para usar Access Client Solutions

- + Si sólo tenemos acceso al IBM i a través de un servidor de salto o el propio IBM i pero el único puerto disponible es de SSH podemos hacer esto:
- + `ssh -L 50000:localhost:23 -L 2001:localhost:2001 -L 2005:localhost:2005 -L 449:localhost:449 -L 8470:localhost:8470 -L 8471:localhost:8471 -L 8472:localhost:8472 -L 8473:localhost:8473 -L 8474:localhost:8474 -L 8475:localhost:8475 -L 8476:localhost:8476 -o ExitOnForwardFailure=yes -o ServerAliveInterval=15 -o ServerAliveCountMax=3 <myuser>@<myIPaddress>`

Para más detalles podemos ir a:

SSH Tunnel configuration for use with IBM i Access Client Solutions



The screenshot shows the 'Connection' tab of the IBM i Access Client Solutions configuration window. The left sidebar lists categories: Connection, Screen, and Preferences. The main area is titled 'Connection' and contains the following fields:

- Session Name: SSH_GARAGE
- Destination Address: 127.0.0.1 (highlighted with a red box and an arrow pointing to it from the Session Name field)
- Destination Port: 50000 (highlighted with a red box and an arrow pointing to it from the Destination Address field)
- Protocol: Use IBM i Access Client Solutions setting (dropdown menu)
- Workstation ID: (empty field with a 'Generate...' button)
- Screen Size: 24x80 (dropdown menu)
- Host Code Page: 037 United States (dropdown menu)
- Unicode Options section:
 - Enable Unicode Data Stream: ☐ Yes ☒ No
 - Enable DBCS in Unicode Fields: ☐ Yes ☒ No

Conocimientos básicos recomendados para sacar el máximo provecho al SSH

- + Se recomienda tener conocimientos básicos de Linux:
 - + Uso de vi (recomendado porque viene de fábrica), vim o nano
 - + Uso de | y > para redireccionar salida standard
 - + Permisos de Unix y uso de links simbólicos
 - + Les recomiendo investigar sobre [SSHUTTLE](#). Permite crear una pseudo-VPN entre un Linux (o MAC) y nuestro IBM i