

# LIMITES DE RECURSOS (ULIMIT)

## ULIMIT - LÍMITES DE LOS USUARIOS

Con `ulimit` y `ulimit -a` puedo revisar la cantidad de recursos a los que tienen acceso los usuarios, así como máximo de archivos abiertos, uso de memoria, etc.

Solo el usuario root puede determinar el número máximo de procesos (**límite hard**) que se pueden ejecutar por parte de los otros usuarios del sistema. Sin embargo, existe un límite que se puede establecer dependiendo de cada usuario (**límite soft**) y que este mismo puede modificar (**como máximo hasta el límite hard**). Por defecto los cambios que haga el usuario al **límite soft** solo tendran efecto en el proceso que este ejecutando o en algunos procesos subsecuentes de ese, porque al momento de cerrar sesión y volver a iniciar los valores del límite soft retornaran a su valor original por defecto, a menos que se ejecute una configuración especial como se muestra a continuación.

### MÁXIMO NUMERO DE ARCHIVOS ABIERTOS

Para mirar el número máximo de archivos abiertos permitidos por inicio de sesión para el usuario , lo podemos hacer a través del comando: `cat /proc/sys/fs/file-max`

### ULIMIT (USER LIMIT)

El comando `ulimit` lo podemos utilizar para ver y definir los límites de recursos para los usuarios que deseen iniciar sesión.

Con `ulimit -a` podemos desplegar toda la información correspondiente a los límites de recursos por usuario:

```

juanjtov@jjserver:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority     (-e) 0
file size                (blocks, -f) unlimited
pending signals         (-i) 3584
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files              (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues    (bytes, -q) 819200
real-time priority      (-r) 0
stack size              (kbytes, -s) 8192
cpu time                (seconds, -t) unlimited
max user processes      (-u) 3584
virtual memory          (kbytes, -v) unlimited
file locks              (-x) unlimited

```

A continuación se describe el significado de los recursos mas importantes mostrados en la tabla:

- `ulimit -n` → It will display number of open files limit
- `ulimit -c` → It display the size of core file
- `umilit -u` → It will display the maximum user process limit for the logged in user.
- `ulimit -f` → It will display the maximum file size that the user can have.
- `umilit -m` → It will display the maximum memory size for logged in user.
- `ulimit -v` → It will display the maximum memory size limit

## **ulimit -n**

Este comando nos permite ver el límite de archivos abiertos que puede tener el usuario por sesión, y como se mencionó anteriormente existen dos tipos de límites (**hard** y **soft**) y se pueden desplegar como se muestra a continuación:

```

juanjtov@jjserver:~$ ulimit -Hn
5000
juanjtov@jjserver:~$ ulimit -Sn
5000

```

En este caso los límites son iguales, pero igual se pueden modificar, hay que tener en cuenta que el **límite soft** nunca puede ser mayor al **hard**.

## NÚMERO MÁXIMO DE ARCHIVOS ABEIRTOS

### Ejemplo de como usar ulimit

Existen ocasiones donde el usuario podría necesitar el número máximo del límite (hard), como ususario adminisitrador puedo modificar ese número y hacerlo de manera temporal o permanente. Para poder hacerlo de forma permanente, de modo que así el usuario salga y entre de la sesión, y aún así la configuración persista, realizamos los siguientes pasos:

Si se quiere incrementar este número a 100000, en primer lugar editamos el siguiente archivo:

```
sudo vim /etc/sysctl.conf
```

Al final del archivo agregamos la siguiente línea:

```
fs.file-max = 100000
```

Ejecutamos el siguiente comando para actualizar la configuración:

```
sudo sysctl -p
```

Adicional a esto podemos configurar los límites listados anteriormente con **ulimit -a**, lo cual se puede realizar modificando el archivo limits.conf de la siguiente manera:

`sudo vim /etc/security/limits.conf` y modificando el archivo como se muestra a continuación, en este caso solo se hizo la modificación del máximo de archivos abiertos y el máximo numero de procesos para los usuarios raíz y \* (es decir para el resto de usuarios):

**NOTA:** Las líneas modificadas fueron las ultimas 8 (observar que no tienen el # que significa comentarios)

```

# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit (KB)
# - maxlogins - max number of logins for this user
# - maxsyslogins - max number of logins on the system
# - priority - the priority to run user process with
# - locks - max number of file locks the user can hold
# - sigpending - max number of pending signals
# - msgqueue - max memory used by POSIX message queues (bytes)
# - nice - max nice priority allowed to raise to values: [-20, 19]
# - rtprio - max realtime priority
# - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>          <value>
#
#*              soft    core            0
#root           hard    core            100000
#*              hard    rss             10000
#@student       hard    nproc           20
#@faculty       soft    nproc           20
#@faculty       hard    nproc           50
#ftp            hard    nproc           0
#ftp            -       chroot            /ftp
#@student       -       maxlogins        4
*              soft    nofile          3500
*              hard    nofile          7000
*              soft    nproc           5000
*              hard    nproc           5000
root           soft    nofile          100000
root           hard    nofile          100000
root           soft    nproc           100000
root           hard    nproc           100000

```

NOTA: Después de esto sin importar si el usuario hace logout y después vuelve a ingresar los valores ya quedan por defecto