Comandos útiles para Ubuntu. Tutorial.

Eduardo C. Garrido Merchán

Sistemas Operativos. Práctica 1. Semana 1.

Motivación

- Cada cuál puede emplear las herramientas que desee, pero se recomienda (por experiencia propia) entrenarse y trabajar con la terminal.

Motivación

- Cada cuál puede emplear las herramientas que desee, pero se recomienda (por experiencia propia) entrenarse y trabajar con la terminal.
- Mejor aprovechamiento de los recursos del ordenador.

Motivación

- Cada cuál puede emplear las herramientas que desee, pero se recomienda (por experiencia propia) entrenarse y trabajar con la terminal.
- Mejor aprovechamiento de los recursos del ordenador.
- Disponibilidad de toda la potencia de comandos proporcionados por vuestra distribución de Linux preferida.

Motivación

- Cada cuál puede emplear las herramientas que desee, pero se recomienda (por experiencia propia) entrenarse y trabajar con la terminal.
- Mejor aprovechamiento de los recursos del ordenador.
- Disponibilidad de toda la potencia de comandos proporcionados por vuestra distribución de Linux preferida.
- Ejemplo: Programar simultaneamente en varios lenguajes de programación. Ejecutar todos sus IDEs consume muchos recursos...

Herramientas recomendadas

- Edición de texto: Vim. (Resaltado de términos, gran variedad de accesos directos, visualización de varios ficheros...).
  Novatos: vimtutor.

Herramientas recomendadas

- Edición de texto: Vim. (Resaltado de términos, gran variedad de accesos directos, visualización de varios ficheros...).
  Novatos: vimtutor.
- Depuración de programas: gdb. (Breakpoints en instrucciones, funciones, condicionales...)

Herramientas recomendadas

- Edición de texto: Vim. (Resaltado de términos, gran variedad de accesos directos, visualización de varios ficheros...).
  Novatos: vimtutor.
- Depuración de programas: gdb. (Breakpoints en instrucciones, funciones, condicionales...)
- Preparación de documentos: Latex.

Herramientas recomendadas

- Edición de texto: Vim. (Resaltado de términos, gran variedad de accesos directos, visualización de varios ficheros...).
  Novatos: vimtutor.
- Depuración de programas: gdb. (Breakpoints en instrucciones, funciones, condicionales...)
- Preparación de documentos: Latex.
- Sistema operativo: Familia Ubuntu. Otros: Debian, Arch Linux...

Herramientas recomendadas

- Edición de texto: Vim. (Resaltado de términos, gran variedad de accesos directos, visualización de varios ficheros...). Novatos: vimtutor.
- Depuración de programas: gdb. (Breakpoints en instrucciones, funciones, condicionales...)
- Preparación de documentos: Latex.
- Sistema operativo: Familia Ubuntu. Otros: Debian, Arch Linux...
- Hay muchísimas herramientas y distribuciones diferentes adecuadas para cada necesidad e individuo.

Gestión de Programas

- sudo apt install [Programa]: Instala un programa.

Gestión de Programas

- sudo apt installl [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.

Gestión de Programas

- sudo apt installl [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.

Gestión de Programas

- sudo apt installl [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.

Gestión de Programas

- sudo apt installl [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.
- whatis [Programa]: Descripción en una línea del programa.

Gestión de Programas

- sudo apt installl [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.
- whatis [Programa]: Descripción en una línea del programa.
- man -k [término]: Busca en todos los manuales el término.

Gestión de Programas

- sudo apt install [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.
- whatis [Programa]: Descripción en una línea del programa.
- man -k [término]: Busca en todos los manuales el término.
- apt search [término]: Busca en el repositorio programas con ese término.

Gestión de Programas

- sudo apt install [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.
- whatis [Programa]: Descripción en una línea del programa.
- man -k [término]: Busca en todos los manuales el término.
- apt search [término]: Busca en el repositorio programas con ese término.
- apt show [programa]: Descarga la información de ese programa.

Gestión de Programas

- sudo apt install [Programa]: Instala un programa.
- apt list –installed: Lista los programas instalados.
- sudo apt autoremove –purge [Programa]: Elimina el programa, sus dependencias y sus ficheros de configuración. OJO: No eliminar aquellos que dependan de [distro]-desktop*.
- man [Programa] : Ayuda del programa.
- whatis [Programa]: Descripción en una línea del programa.
- man -k [término]: Busca en todos los manuales el término.
- apt search [término]: Busca en el repositorio programas con ese término.
- apt show [programa]: Descarga la información de ese programa.
- ...

Navegación y exploración

- cd. Abre el directorio. cd $\sim$ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.

Navegación y exploración

- cd. Abre el directorio. cd ~ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).

Navegación y exploración

- cd. Abre el directorio. cd ~ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".

Navegación y exploración

- cd. Abre el directorio. cd ∼ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".
- locate. Busca ficheros por el nombre en todo el sistema.

Navegación y exploración

- cd. Abre el directorio. cd ∼ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".
- locate. Busca ficheros por el nombre en todo el sistema.
- whereis. Localiza el manual, binario y fuentes de un programa.

Navegación y exploración

- cd. Abre el directorio. cd ~ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".
- locate. Busca ficheros por el nombre en todo el sistema.
- whereis. Localiza el manual, binario y fuentes de un programa.
- mkdir. Crea un directorio.

Navegación y exploración

- cd. Abre el directorio. cd ~ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".
- locate. Busca ficheros por el nombre en todo el sistema.
- whereis. Localiza el manual, binario y fuentes de un programa.
- mkdir. Crea un directorio.
- rm. Borra un fichero. rmdir. Borra un directorio.

Navegación y exploración

- cd. Abre el directorio. cd ~ lleva al home. cd / al raíz. cd .. al directorio padre y cd . al actual.
- ls. Lista contenidos del directorio. -l: Detalles. -a: Ocultos (Directorios con . al principio).
- find. Busca ficheros a partir del directorio actual. Ejemplo: find . -name "*.txt".
- locate. Busca ficheros por el nombre en todo el sistema.
- whereis. Localiza el manual, binario y fuentes de un programa.
- mkdir. Crea un directorio.
- rm. Borra un fichero. rmdir. Borra un directorio.
- ...

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.
- |. Redirige la salida del proceso anterior al siguiente.

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.
- |. Redirige la salida del proceso anterior al siguiente.
- >. Redirecciona la salida al fichero deseado.

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.
- |. Redirige la salida del proceso anterior al siguiente.
- >. Redirecciona la salida al fichero deseado.
- &&. Permite ejecutar varios comandos a la vez, independientes.

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.
- |. Redirige la salida del proceso anterior al siguiente.
- >. Redirecciona la salida al fichero deseado.
- &&. Permite ejecutar varios comandos a la vez, independientes.
- >>. Escribe al final del fichero deseado la salida.

Manipulación de procesos

- &. Ordena que el proceso se ejecute en segundo plano.
- |. Redirige la salida del proceso anterior al siguiente.
- >. Redirecciona la salida al fichero deseado.
- &&. Permite ejecutar varios comandos a la vez, independientes.
- >>. Escribe al final del fichero deseado la salida.
- ...

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.
- head. Imprime las primeras n líneas de un fichero. head -n.

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.
- head. Imprime las primeras n líneas de un fichero. head -n.
- tail. Imprime las últimas n líneas de un fichero. tail -n. -f actualiza.

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.
- head. Imprime las primeras n líneas de un fichero. head -n.
- tail. Imprime las últimas n líneas de un fichero. tail -n. -f actualiza.
- less. Te permite navegar por el fichero.

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.
- head. Imprime las primeras n líneas de un fichero. head -n.
- tail. Imprime las últimas n líneas de un fichero. tail -n. -f actualiza.
- less. Te permite navegar por el fichero.
- nl. Imprime un contador de líneas. Ejemplo: cat ejemplo.txt | nl | less.

Visualización de ficheros

- cat. Imprime por pantalla el nombre de fichero.
- head. Imprime las primeras n líneas de un fichero. head -n.
- tail. Imprime las últimas n líneas de un fichero. tail -n. -f actualiza.
- less. Te permite navegar por el fichero.
- nl. Imprime un contador de líneas. Ejemplo: cat ejemplo.txt | nl | less.
- ...

Manipulación de ficheros

- grep. Busca un patrón de texto. -ri lo busca recursivo e ignorando mayúsculas y minúsculas.

Manipulación de ficheros

- grep. Busca un patrón de texto. -ri lo busca recursivo e ignorando mayúsculas y minúsculas.
- wc. Cuenta los caracteres. -l cuenta las líneas.

Manipulación de ficheros

- grep. Busca un patrón de texto. -ri lo busca recursivo e ignorando mayúsculas y minúsculas.
- wc. Cuenta los caracteres. -l cuenta las líneas.
- sort. Ordena las líneas de un fichero.

Manipulación de ficheros

- grep. Busca un patrón de texto. -ri lo busca recursivo e ignorando mayúsculas y minúsculas.
- wc. Cuenta los caracteres. -l cuenta las líneas.
- sort. Ordena las líneas de un fichero.
- uniq. Devuelve las líneas no repetidas de un fichero.

Manipulación de ficheros

- grep. Busca un patrón de texto. -ri lo busca recursivo e ignorando mayúsculas y minúsculas.
- wc. Cuenta los caracteres. -l cuenta las líneas.
- sort. Ordena las líneas de un fichero.
- uniq. Devuelve las líneas no repetidas de un fichero.
- ...

Scripting

- Declaración e inicialización de variables.

Scripting

- Declaración e inicialización de variables.
- Bucles: for i in $seq(inicio incremento fin); do comandos; done

Scripting

- Declaración e inicialización de variables.
- Bucles: for i in $seq(inicio incremento fin); do comandos; done
- Condicionales: if [$i -gt 0]; then echo "hola"; else echo "adios"; fi.

Scripting

- Declaración e inicialización de variables.
- Bucles: for i in $seq(inicio incremento fin); do comandos; done
- Condicionales: if [$i -gt 0]; then echo "hola"; else echo "adios"; fi.
- ‘comando‘. Recoge los resultados de comando en una lista para su procesamiento. for i in ‘find . -name "*.txt"‘; do echo $i; done

Scripting

- Declaración e inicialización de variables.
- Bucles: for i in $seq(inicio incremento fin); do comandos; done
- Condicionales: if [$i -gt 0]; then echo "hola"; else echo "adios"; fi.
- ‘comando‘. Recoge los resultados de comando en una lista para su procesamiento. for i in ‘find . -name "*.txt"‘; do echo $i; done
- ...

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.
- pstree Muestra el proceso como un árbol.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.
- pstree Muestra el proceso como un árbol.
- df. Muestra la cantidad de espacio libre en disco. -h (legible).

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.
- pstree Muestra el proceso como un árbol.
- df. Muestra la cantidad de espacio libre en disco. -h (legible).
- du. Muestra el espacio libre del directorio. -h (legible). –max-depth=x profundidad x en un directorio.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.
- pstree Muestra el proceso como un árbol.
- df. Muestra la cantidad de espacio libre en disco. -h (legible).
- du. Muestra el espacio libre del directorio. -h (legible). –max-depth=x profundidad x en un directorio.
- free. Muestra el espacio libre en memoria.

Gestión de procesos y sistema

- top. Muestra información de los procesos en ejecución.
- cat /proc/cpuinfo. Muestra información del equipo utilizado.
- ps. Muestra información de los procesos controlados por un terminal. -ea: Muestra todos los procesos.
- pstree Muestra el proceso como un árbol.
- df. Muestra la cantidad de espacio libre en disco. -h (legible).
- du. Muestra el espacio libre del directorio. -h (legible). –max-depth=x profundidad x en un directorio.
- free. Muestra el espacio libre en memoria.
- ...

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...
- Incluso podemos hacer operaciones como imprimir documentos... lpr.

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...
- Incluso podemos hacer operaciones como imprimir documentos... lpr.
- O usar el scripting para ganar tiempo: for j in $(seq 1 1 25); do for i in $(seq 1 1 200); do echo -5 >> exp_$j/human/error_test.txt; done; done

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...
- Incluso podemos hacer operaciones como imprimir documentos... lpr.
- O usar el scripting para ganar tiempo: for j in $(seq 1 1 25); do for i in $(seq 1 1 200); do echo -5 >> exp_$j/human/error_test.txt; done; done
- CTRL+R: cadena. Búsqueda del último comando que contenga a cadena.

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...
- Incluso podemos hacer operaciones como imprimir documentos... lpr.
- O usar el scripting para ganar tiempo: for j in $(seq 1 1 25); do for i in $(seq 1 1 200); do echo -5 >> exp_$j/human/error_test.txt; done; done
- CTRL+R: cadena. Búsqueda del último comando que contenga a cadena.
- ...

Trucos

- alias. Crea un alias. Si se guarda en .bash_aliases ya se tiene disponible. Ejemplo: alias "mytool=cd /dev/java/mytool/"
- watch -n[tiempo] [comando]. Ejecuta cada tiempo un comando.
- La terminal puede abrir cualquier tipo de fichero con su programa: evince, eog, acroread, vlc, firefox...
- Incluso podemos hacer operaciones como imprimir documentos... lpr.
- O usar el scripting para ganar tiempo: for j in $(seq 1 1 25); do for i in $(seq 1 1 200); do echo -5 >> exp_$j/human/error_test.txt; done; done
- CTRL+R: cadena. Búsqueda del último comando que contenga a cadena.
- ...
- Ganarás tiempo, no dudes en emplear la terminal.