Linux command usage

1. apt

apt (Advanced Packaging Tool) is a shell frontend package manager in Debian and Ubuntu.

apt command provides commands for finding, installing, upgrading, and deleting a single, a group, or even all software packages, and the commands are concise and easy to remember.

The execution of the apt command requires super-administrator privileges (root).

- ### Grammar explanation

```
apt [options] [command] [package ...]
```

  - **options:** Optional, options include - h (Help), - y (When prompted to select all as "yes" during installation), - q (Do not display installation process), and so on.
  - **command:** The operation to be performed.
  - **package**: The name of the installed package.

- ## Common commands

- Update the apt software source database：**sudo apt update**
- Update installed software packages：**sudo apt upgrade**

  List upgradable software packages and version information：**apt list --upgradeable**

  Upgrade software package, delete the package that needs to be updated before upgrading：**sudo apt full-upgrade**
- Install the specified software command：**sudo apt install <package_name>**

  Install multiple software packages：**sudo apt install <package_1> <package_2> <package_3>**
- Display specific information about the software package, such as version number, installation size, dependencies, etc：**sudo apt show <package_name>**
- Delete package command：**sudo apt remove <package_name>**
- Clean up dependencies and library files that are no longer in use: **sudo apt autoremove**
- Remove software packages and configuration files: **sudo apt purge <package_name>**
- Find software package command： **sudo apt search <package_name>**
- List all installed packages：**apt list --installed**
- List the version information of all installed packages：**apt list --all-versions**

2. dmesg

`The dmesg` command is used to view or control the kernel ring buffer.

The kernel will store the kernel startup log in the ring buffer. If you don't have time to view the information when turning on the computer, you can use dmesg to check.

- ## Grammar explanation

```
dmesg [options]

       dmesg -C / dmesg --clear
       dmesg -c / dmesg --read-clear [options]
```

  - Option description
  - -c, --read-clear　After displaying the information, clear the contents of the ring buffer.
  - -C, --clear　Clear the contents of the ring buffer.
- ## Common commands

  - Display the kernel log contents of all ring buffers

    ```
    dmesg
    ```

  - Save kernel logs to a file

    ```
    dmesg > kernel.log
    ```

  - Clearing cache logs can reduce log content during driver debugging

```
dmesg -C
```

3. dpkg-deb

dpkg deb command is a package management tool under Debian Linux, which can perform packaging and unpacking operations on software packages and provide package information.

- ## Grammar explanation

```
dpkg-deb [<option> ...] <command>
```

- ## Command description

  dpkg-deb command not only has options to set, but also requires commands to perform different functions.

  - -b：Create a debian software package；
  - -c：Display the file list in the software package；
  - -e：Decompress the main control information;
  - -f：Print the field content to standard output;
  - -x：Release the files in the software package to the specified directory;
  - -X：Release the files in the software package to the specified directory and display the detailed process of releasing the files;
  - -w：Display information about software packages;
  - -l：Display detailed information of the software package;
  - -R：Extract control information and archive inventory files;
  - -b: Create a debian package.
  - -c: Display the file list in the software package.
  - -e: Decompress the main control information.
  - -f: Print the field content to standard output.
  - -x: Release files from the software package to the specified directory.
  - -X: Release the files in the software package to the specified directory and display the detailed process of releasing the files.
  - -w: Display information about the software package.
  - -l: Display detailed information of the software package.
  - -R: Extract control information and archive inventory files.

- ## Option description

  - `-v, --verbose`：Enable detailed output.
  - `-D, --debug`：Enable debugging output.
    - `--showformat=<format>`：Use alternative formats for `--show`。
    - `--deb-format=<format>`：Select the archive format. The allowed values are 0.939000 and 2.0 (default values).
    - `--nocheck`：Prohibit controlling file checks (building malicious software packages).
    - `--root-owner-group`：Force the owner and group of the file to be root.
    - `--[no-]uniform-compression`：Use compression parameters on all members. If specified, uniform compression parameters will be used.
  - `-z#`：Set the compression level during construction.

- `-Z<type>`：Set the compression type used during construction. The allowed types are gzip, xz, zstd, and none.
- `-S<strategy>`：Set compression strategy during construction. The allowed values are none, extreme (xz), filtered, Huffman, rle, fixed (gzip).

- ## Common commands

  - Extract program files

    ```
    dpkg-deb -x hobot-configs_2.2.0-20231030133209_arm64.deb
    ```

  - Extract control file

    ```
    dpkg-deb -e hobot-configs_2.2.0-20231030133209_arm64.deb hobot-configs/DEBIAN
    ```

  - Query the file contents in the deb package

    ```
    dpkg-deb -c hobot-configs_2.2.0-20231030133209_arm64.deb
    ```

4. dpkg

Install, create, and manage software packages on a Debian Linux system.

dpkg command is a utility used by Debian Linux systems to install, create, and manage software packages.

- ## Grammar explanation

  ```
  dpkg [<option> ...] <command>
  ```

- ## Command description

  dpkg command not only has options to set, but also requires commands to perform different functions.

  - -i：Install software packages.
  - -r：Delete software packages.
  - -P：Delete software packages while deleting their configuration files.
  - -L：List files belonging to the specified software package.
  - -l：Briefly list the status of the software package.
  - -S：Search for software packages containing specified files.
  - --unpack: Unpack the software package.
  - -c：Display the file list within the software package.
  - --confiugre: Configure software packages.

- ## Option description

  - --admindir=Useinstead of/var/lib/dpkg.
  - --root=Install to another root directory.
  - -- instdir=Change the installation directory while keeping the management directory unchanged.

- -- path exclude=Do not install paths that match Shell expressions.
- --path include=After excluding a pattern, include another pattern.
- -O | -- selected only ignores software packages that have not been selected for installation or upgrade.
- -E | -- skip same version ignores packages with the same version as the installed software version.
- -G | -- reuse downgrade ignores packages with versions earlier than the installed software version.
- -B | -- Auto configure needs to be installed even if it affects other software packages.
- --[no-]triggers Skip or force trigger handling that occurs with it.
- -- verify format=Check the output format ('rpm 'is supported).
- --no-debsig does not attempt to verify the signature of the software package.
- -D | -- debug=Enable debugging (see - DHelp or -- debug=help).
- --status logger=sends status updates to the standard input of.
- --log=Update status and operation information to.
- --ignore pending= Ignore all dependencies related to.
- --force-... Ignore the problems encountered (see -- force help).
- --no-force-...|-- Pause when encountering problems.
- --abort-after encounteringerrors cumulatively.

Common commands

- Installation package

```
dpkg -i package.deb
```

- Delete package

```
dpkg -r package
```

- Delete packages (including configuration files)

```
dpkg -P package
```

- List the files associated with the package

```
dpkg -L package
```

- Display the version of the package

```
dpkg -l package
```

- Unzip the contents of the deb package

```
dpkg --unpack package.deb
```

- Search for the content of the package to which it belongs

```
dpkg -S keyword
```

- List the currently installed packages

```
dpkg -l
```

- List the contents of the deb package

```
dpkg -c package.deb
```

- Configuration package

```
dpkg --configure package
```

5. find

The find command is used to search for files and directories in a specified directory.

It can use different options to filter and restrict the search results. Any string before the parameter will be considered as the directory name to be searched for. If no parameters are set when using this command, the find command will search for subdirectories and files in the current directory. And display all the subdirectories and files found.

- ## Grammar explanation

```
find [-H] [-L] [-P] [-Olevel] [-D debugopts] [path...] [expression]
```

- ## Option description

path The directory path to be searched can be a single directory or file name, or multiple paths separated by spaces. If no path is specified, it defaults to the current directory.

expression It is an optional parameter used to specify the search criteria, which can include file name, file type, file size, and so on.

There are many options available in the expression, and the most commonly used ones are listed below.

- `-name pattern`: Search by file name, supporting the use of wildcard characters `*` and `?`.
- `-iname pattern`: The effect of this parameter is similar to specifying the `-name` parameter, but ignores the difference in character case;
- `-type type`: Search by file type, which can be `f` (regular file), `d` (directory), `l` (symbolic link), etc.
- `-size [+-]size[cwbkMG]`: Search by file size, supporting the use of `+` or `-` to indicate larger or smaller than a specified size. The units can be `c` (bytes), `w` (word count), `b` (block count), `k` (KB), `M` (MB), or `G` (GB).
- `-mtime days`: Search by modification time, supports using `+` or `-` to indicate before or after a specified number of days, days is an integer representing the number of days.
- `-user username`: Search by file owner.
- `-group groupname`: Search by file group.

The parameters used for time in the find command are as follows.

- `-amin n`: Search for files that have been accessed within n minutes.
- `-atime n`: Search for files that have been accessed within n*24 hours.
- `-cmin n`: Search for files with status changes (such as permissions) within n minutes.
- `-ctime n`: Search for files with status changes (such as permissions) within n*24 hours.
- `-mmin n`: Search for files that have been modified within n minutes.
- `-mtime n`: Search for files that have been modified within n*24 hours.

Among these parameters, n can be a positive, negative, or zero number.

Positive numbers indicate files that have been modified or accessed within a specified time.

Negative numbers indicate files that have been modified or accessed before the specified time.

Zero indicates files that have been modified or accessed at the current time point.

For example, **- mtime 0** indicates searching for files modified today, and **- mtime -7**indicates searching for files modified a week ago.

Explanation on the parameter of time n:

- **+n**: Search for files or directories that are older than n days ago.
- **-n**: Search for files or directories that have changed their properties within n days.
- **n**: Find files or directories whose properties have been changed n days ago (specify the day).

Common commands

List all files and folders in the current directory and subdirectories

```
find .
```

Search for a file named 'file. txt' in the current directory

```
find . -name file.txt
```

List all files with the suffix **.c** in the current directory and its subdirectories

```
find . -name "*.c"
```

Same as above, but ignoring capitalization

```
find . -iname "*.c"
```

List all files in the current directory and its subdirectories

```
find . -type f
```

Search for files larger than 1MB in the/home directory

```
find . -size +1M
```

Search for files smaller than 10KB

```
find . -type f -size -10k
```

Search for files equal to 10KB

```
find . -type f -size 10k
```

- File size unit:
  - **b** —— Block (512 bytes)
  - **c** —— Byte
  - **w** —— Word（2 bytes）
  - **k** —— kilobyte
  - **M** —— megabyte
  - **G** —— Gigabyte

Search for files in/var/log directory that were modified 7 days ago.

```
find /var/log -mtime +7
```

List all files in the current directory and its subdirectories that have been updated in the last 20 days.

```
find . -ctime  20
```

List all files updated 20 days ago or earlier in the previous directory and its subdirectories.

```
find . -ctime  +20
```

List all files updated within the last 20 days in the current directory and its subdirectories.

```
find . -ctime  20
```

Search for regular files in the/var/log directory that have been changed before the 7th, and ask them before deleting them.

```
find /var/log -type f -mtime +7 -ok rm {} \;
```

Find files in the current directory that belong to the owner with read and write permissions, and whose group users and other users have read permissions.

```
find . -type f -perm 644 -exec ls -l {} \;
```

Find all regular files with a length of 0 in the system and list their complete paths.

```
find / -type f -size 0 -exec ls -l {} \;
```

Search for all files ending in. txt and. pdf in the current directory and subdirectories.

```
find . \( -name "*.txt" -o -name "*.pdf" \)
Or
find . -name "*.txt" -o -name "*.pdf"
```

Match file path or file.

```
find /usr/ -path "*local*"
```

Matching file paths based on regular expressions.

```
find . -regex ".*\(\.txt\|\.pdf\)$"
```

Same as above, but ignoring capitalization.

```
find . -iregex ".*\(\.txt\|\.pdf\)$"
```

Negative parameter, find files under/home that do not end in. txt.

```
find /home ! -name "*.txt"
```

Search by file type.

```
find . -type Type parameter
```

- Type parameter list
  - **f** Regular file
  - **l** Symbol connection
  - **d** Directory
  - **c** Character device
  - **b** Block device
  - **s** Socket
  - **p** Fifo

Based on directory depth search, the maximum downward depth limit is 3

```
find . -maxdepth 3 -type f
```

Search for all files with a depth distance of at least 2 subdirectories from the current directory

```
find . -mindepth 2 -type f
```

Delete all `.log` files in the current directory.

```
find . -type f -name "*.log" -delete
```

Search for a file with permission 777 in the current directory

```
find . -type f -perm 777
```

Find the `. conf` file in the current directory with permissions other than 644

```
find . -type f -name "*.conf" ! -perm 644
```

Find all files owned by the current directory user `sunrise`

```
find . -type f -user sunrise
```

Find all files owned by the current directory user group `sunrise`

```
find . -type f -group sunrise
```

Find all `root` files in the current directory and change the ownership to user `sunrise`

```
find .-type f -user root -exec chown sunrise {} \;
```

In the example above, **{}** is used in combination with the **-exec** option to match all files, which are then replaced with the appropriate file name.

Find all the `.txt` files in the `home` directory and delete them.

```
find $HOME/. -name "*.txt" -ok rm {} \;
```

In the above example, the behavior of **- ok** and **- executive** is the same, but it will give a prompt whether to perform the corresponding operation.

Search for all `. txt` files in the current directory and concatenate them to write them into the `all. txt` file

```
find . -type f -name "*.txt" -exec cat {} \;> /all.txt
```

Search for all. txt files in the current directory or subdirectories, but skip subdirectories sk

```
find . -path "./sk" -prune -o -name "*.txt" -print
```

> ⚠ ./sk cannot be written as./sk/; otherwise, it has no effect.

Ignore two directories

```
find . \( -path ./sk -o  -path ./st \) -prune -o -name "*.txt" -print
```

> ⚠ If writing relative path must add `./`

To list all files of zero length.

```
find . -empty
```

Count lines of code

```
find . -name "*.c"|xargs cat|grep -v ^$|wc -l # Code line count, excluding
empty lines
```

6. grep command

Powerful text search tool.

Grep (global search regular expression (RE) and print out the line) is a powerful text search tool that can use regular expressions to search for text and print out matching lines. Specific characters used for filtering/searching. Regular expressions can be used in conjunction with various commands, making them very flexible to use.

Similar commands include egrep, fgrep, and rgrep.

- ## Grammar specification

```
grep [OPTION]... PATTERNS [FILE]...
grep [OPTION...] PATTERNS [FILE...]
grep [OPTION...] -e PATTERNS ... [FILE...]
grep [OPTION...] -f PATTERN_FILE ... [FILE...]
```

  - **PATTERNS** - Indicates the string or regular expression to look for.
  - **FILE** - Indicates the FILE name to be searched. Multiple files can be searched at the same time. If the 'file' parameter is omitted, the data is read from the standard input by default.

- ## Option description

Common options description:

  - `-i`: Ignore case for matching.
  - `-v`: Reverse lookup, printing only rows that do not match.
  - `-n`: Displays the line number of the matching row.
  - `-r`: Find files in subdirectories recursively.
  - `-l`: Only matching filenames are printed.
  - `-c`: Only the number of matching lines is printed.

More Parameter Description

  - **-a or -- text** : Do not ignore binary data.
  - **-Aor -- after context=** : In addition to displaying the column that conforms to the template style, display the content after that row.
  - **- b or -- byte off set** : Before displaying the line that conforms to the style, indicate the number of the first character of that line.
  - **-Bor -- before content=** : Display the content before the line, in addition to the line that matches the style.
  - **-c or -- count**: Calculate the number of columns that match the style.

- **-Cor -- context=or -**: In addition to displaying the line that conforms to the style, display the content before and after that line.
- **-dor -- directories=** : This parameter must be used when specifying that the search is for a directory rather than a file, otherwise the grep directive will report information and stop the action.
- **-e**