xclip X11 clipboard from shell

START HERE

`<cmd> | xclip -sel clip`
Copy output to clipboard

`xclip -o | xclip -sel clip`
Primary to clipboard

`xclip -sel clip <file>`
Copy a file's text

`xclip -o -sel clip > <file>`
Save clipboard to file

`xclip -o -sel clip`
Paste clipboard to stdout

`xclip -sel clip -t image/png -i <file>`
Copy an image

`xclip -o`
Print highlighted text

`xclip -o -t TARGETS -sel clip`
List formats on offer

stdin / file your data → `xclip -i` → **X selection** held by xclip → `app pastes` → **Application** requests a target

CONCEPTS

Without -sel, xclip uses PRIMARY (highlight, middle-click), not the Ctrl+C clipboard.

xclip stays in the background to serve pastes; it exits when another program takes the selection.

X11 only. On Wayland use wl-copy and wl-paste from wl-clipboard.

If a selection vanishes after the source exits, that is normal X behaviour: owner serves the data.

COPY

`<cmd> | xclip`
Copy output to PRIMARY

`<cmd> | xclip -sel clip`
Copy output to clipboard

`xclip -sel clip <file>`
Copy file contents to clipboard

`xclip -i`
Read stdin into selection (default mode)

`xclip -r -sel clip`
Drop the trailing newline
Handy for pwd or a path, so pasting into a shell does not run it.

`xclip -f -sel clip`
Copy and pass stdin through to stdout
Use mid-pipeline: cmd | xclip -f -sel clip | next

`xclip -sel clip -l 1`
Exit after one paste

`xclip -l <n> -verbose <file>`
Serve n pastes, log each request

COPY (cont.)

`xclip -sensitive -sel clip`
Clear the selection after one paste
Implemented as -wait 50; raise -wait if pasting then fails.

`xclip -wait <ms> -sel clip`
Clear selection <ms> after first paste

PASTE

`xclip -o`
Print PRIMARY

`xclip -o -sel clip`
Print clipboard

`xclip -o -sel clip > <file>`
Save clipboard to a file

`xclip -o -sel sec`
Print SECONDARY

`xclip -o | xclip -sel clip`
Copy PRIMARY to clipboard

`xclip -o -sel clip | xclip`
Copy clipboard to PRIMARY

`$(xclip -o -sel clip)`
Use clipboard in a command

`xclip -o -sel clip | wc -w`
Count words in clipboard

SELECTIONS & DISPLAY

`-sel primary / -sel p`
Highlighted text (default)

`-sel clipboard / -sel c`
Ctrl+C clipboard

`-sel secondary / -sel s`
Rarely used third selection

`-d <host:0> / -display <host:0>`
Pick the X display
Defaults to $DISPLAY; needed over ssh without -X.

SELECTIONS & DISPLAY (cont.)

`-quiet`
Stay in the foreground, show messages

`-silent`
Fork to background (default)

`-verbose`
Log activity and paste requests

`-noutf8`
Legacy non-UTF-8 text handling

`-version`
Show installed version

`-h / -help`
Short usage summary

TARGETS & IMAGES

`xclip -o -t TARGETS -sel clip`
List formats currently offered

`xclip -sel clip -t <type> -i <file>`
Copy as a given MIME type

`xclip -sel clip -t image/png -i <file>`
Copy an image to clipboard

`xclip -o -sel clip -t image/png > <file>`
Save a clipboard image

`xclip -sel clip -t text/html <file>`
Offer a file as HTML

`xclip -t text/html -alt-text "<text>"`
Plain-text fallback beside HTML

`xclip -o -sel clip -t text/uri-list`
Read copied file paths

TARGETS & IMAGES (cont.)

Without -t the target is STRING. Images and HTML need an explicit MIME type, and the receiving app must support it.

FILE TRANSFER

`xclip-copyfile <file>`
Copy a file or tree into the clipboard

`xclip-copyfile -p <file>`
Same, keeping the path

`xclip-cutfile <file>`
Mark a file for moving

`xclip-pastefile`
Paste copied files to current dir

These helper scripts need an X11 connection on both ends, so they work between machines too.

RECIPES

Copy a public SSH key

```
xclip -sel clip < ~/.ssh/id_ed25519.pub
# paste into the web form with Ctrl+V
```

Copy your working directory

```
pwd | xclip -r -sel clip
```

Screenshot into clipboard

```
import png:- | xclip -sel clip -t image/png
# then Ctrl+V into chat or editor
```

Fix clipboard text and put it back

```
xclip -o -sel clip | tr -d '\r' | xclip -sel clip
# transform: paste, filter, copy
```

Pick a line with fzf and copy it

```
history | fzf | cut -c8- | xclip -r -sel clip
```