Capturing your Desktop / Screen Recording

Here are a few solutions for capturing your desktop and recording a video of your screen with ffmpeg. (A Chinese version of this page is also available.)

For the sake of brevity, these commands do not specify any additional encoder settings. For more info about H.264 encoding, see the H.264 encoding guide.

By default, these commands will use the x264 encoder, which should be reasonably fast on modern machines. See Lossless Recording if you need to improve performance.

Linux

Use the x11grab device:

```
ffmpeg -video_size 1024x768 -framerate 25 -f x11grab -i :0.0+100,200 output.mp4
```

This will grab the image from desktop, starting with the upper-left corner at x=100, y=200 with a width and height of 1024×768.

If you need audio too, you can use ALSA (see Capture/ALSA for more info):

```
ffmpeg -video_size 1024x768 -framerate 25 -f x11grab -i :0.0+100,200 -f alsa -ac 2 -i hw:0 output.mkv
```

Or the pulse input device:

```
ffmpeg -video_size 1024x768 -framerate 25 -f x11grab -i :0.0+100,200 -f pulse -ac 2 -i default output.mkv
```

macOS

Use the avfoundation device:

```
ffmpeg -f avfoundation -list_devices true -i ""
```

This will enumerate all the available input devices including screens ready to be captured.

Once you've figured out the device index corresponding to the screen to be captured, use:

```
ffmpeg -f avfoundation -i "<screen device index>:<audio device index>" output.mkv
```

This will capture the screen from `<screen device index>` and audio from `<audio device index>` into the output file `output.mkv`.

Windows

Use DirectShow

Use a DirectShow device:

```
ffmpeg -f dshow -i video="screen-capture-recorder" output.mkv
```

This will grab the image from entire desktop. You can refer to a list of alternative devices.

If you need audio too:

```
ffmpeg -f dshow -i video="UScreenCapture":audio="Microphone" output.mkv
```

If you want to capture the audio that is playing from your speakers you may also need to configure so-called "Stereo Mix" device.

or

```
ffmpeg -f dshow -i video="UScreenCapture" -f dshow -i audio="Microphone" output.mkv
```

You can list your devices with:

```
ffmpeg -list_devices true -f dshow -i dummy
```

Use built-in GDI screengrabber

You can also use gdigrab as input device to grab video from the Windows screen.

To capture all your displays as one big contiguous display:

```
ffmpeg -f gdigrab -framerate 30 -i desktop output.mkv
```

If you want to limit to a region, and show the area being grabbed:

```
ffmpeg -f gdigrab -framerate 30 -offset_x 10 -offset_y 20 -video_size 640x480 -show_region 1 -i desktop output.mkv
```

To grab the contents of the window named "Calculator":

```
ffmpeg -f gdigrab -framerate 30 -i title=Calculator output.mkv
```

Hardware Encoding

You can use hardware acceleartion to speed up encoding and reduce the load on your CPU. For example, with NVIDIA hardware encoding:

```
ffmpeg -f gdigrab -framerate 30 -i desktop -c:v h264_nvenc -qp 0 output.mkv
```

Lossless Recording

If your CPU is not fast enough, the encoding process might take too long. To speed up the encoding process, you can use lossless encoding and disable advanced encoder options, e.g.:

```
ffmpeg -video_size 1920x1080 -framerate 30 -f x11grab -i :0.0 -c:v libx264 -crf 0 -preset ultrafast output.mkv
```

`-crf 0` tells x264 to encode in lossless mode; `-preset ultrafast` advises it to do so fast.

The encoder should be fast enough on most modern hardware to record without any framedrop, and even leave enough CPU headroom for other applications.

If you're going to archive the recording or are concerned about file size, re-encode it losslessly again, but with a slower preset. Note that since the initial recording was lossless, and the re-encode is lossless too, no quality loss is introduced in this process in any way.

```
ffmpeg -i output.mkv -c:v libx264 -crf 0 -preset veryslow output-smaller.mkv
```