Platform limitations

pynput aims at providing a unified *API* for all supported platforms. In some cases, however, that is not entirely possible.

Linux

On *Linux*, *pynput* uses *X*, so the following must be true:

- An *X server* must be running.
- The environment variable `$DISPLAY` must be set.

The latter requirement means that running *pynput* over *SSH* generally will not work. To work around that, make sure to set `$DISPLAY`:

```
$ DISPLAY=:0 python -c 'import pynput'
```

Please note that the value `DISPLAY=:0` is just an example. To find the actual value, please launch a terminal application from your desktop environment and issue the command `echo $DISPLAY`.

macOS

Recent versions of *macOS* restrict monitoring of the keyboard for security reasons. For that reason, one of the following must be true:

- The process must run as root.
- Your application must be white listed under *Enable access for assistive devices*. Note that this might require that you package your application, since otherwise the entire *Python* installation must be white listed.
- On versions after *Mojave*, you may also need to whitelist your terminal application if running your script from a terminal.

Please note that this does not apply to monitoring of the mouse or trackpad.

Windows

On *Windows*, virtual events sent by *other* processes may not be received. This library takes precautions, however, to dispatch any virtual events generated to all currently running listeners of the current process.

Furthermore, sending key press events will properly propagate to the rest of the system, but the operating system does not consider the buttons to be truly *pressed*. This means that key press events will not be continuously emitted as when holding down a physical key, and certain key sequences, such as *shift*pressed while pressing arrow keys, do not work as expected.