Project Report — Keylogger with Encrypted Data Exfiltration

Disclaimer

This project was created strictly for educational and research purposes. It was developed and tested only in a controlled virtual machine environment. Deploying keyloggers on machines without explicit consent is illegal and unethical.

Objective

Build a proof-of-concept keylogger that:
1. Captures keystrokes using pynput.
2. Encrypts logs using cryptography.fernet.
3. Stores logs locally with timestamps.
4. Simulates exfiltration by sending encrypted logs to a local server.
5. Implements persistence and a kill switch (for controlled PoC demonstration).

Tools & Libraries

- Python 3.x
- pynput
- cryptography (Fernet)
- base64
- socket
- pathlib
- time/os libraries

Key Features

- Captures keystrokes (letters, spaces, special keys).
- Encrypts data with a persistent Fernet key stored in key.txt.
- Saves encrypted logs as log-YYYYMMDD-HHMMSS.enc.
- Simulates exfiltration to a local TCP server.
- Hotkey Kill Switch (Ctrl+Shift+Q) to stop the logger safely.
- Persistent key ensures consistent decryption across runs.

Implementation Overview

1. **Keystroke Capture**: Using pynput to detect keypresses and append to a log string.
2. **Encryption**: Symmetric encryption via Fernet ensures confidentiality.
3. **Storage**: Logs saved locally as encrypted binary files with timestamps.
4. **Network Simulation**: Encrypted logs base64-encoded and sent to a local server.
5. **Persistence**: Implemented through a persistent key (key.txt). Optional documentation for startup persistence methods.
6. **Kill Switch**: Hotkey combo (Ctrl+Shift+Q) stops the logger and saves logs.

Testing Procedure

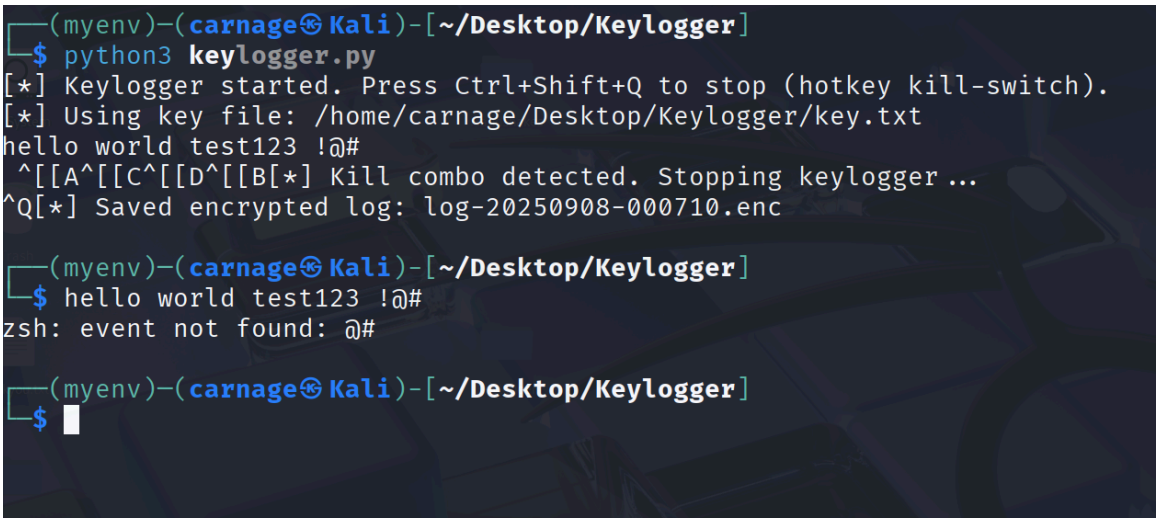
1. Start the receiver server (server.py) on localhost.
2. Run the keylogger (keylogger.py) inside the VM.
3. Type dummy test data including spaces, enter, and arrow keys.
4. Stop using the Ctrl+Shift+Q kill switch.
5. Verify encrypted .enc log file creation.
6. Decrypt using decrypt_file.py with persistent key.txt.
7. Confirm decrypted file matches dummy input.
8. Optionally verify base64 log received by server.

Results

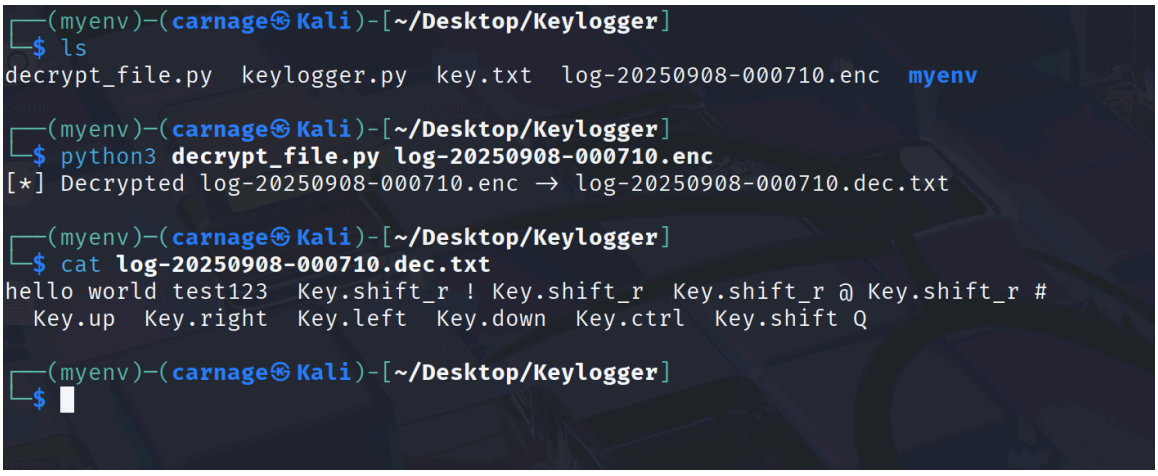
- ✅ Logs captured accurately
- ✅ Logs encrypted and saved with timestamps
- ✅ Persistent key enabled successful decryption
- ✅ Kill switch stopped logger as expected
- ✅ Network exfiltration simulation successful (localhost only)

Screenshots

Running the Keylogger



Decrypting the saved logs



Cleanup

After testing, encrypted and decrypted logs were removed using:

rm log-*.enc *.dec.txt

Conclusion

The proof-of-concept keylogger successfully demonstrates keystroke capture, encryption with a persistent key, local log storage, network exfiltration simulation, and safe termination via a kill switch. The project was implemented ethically and tested only in a controlled VM environment.