AnonRoute – ANONYMOUS TRAFFIC ROUTING TOOL

### MAJOR PROJECT REPORT

***Submitted by***

**AZADVIR SNGH**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

### IN

**COMPUTER SCIENCE AND ENGINEERING**

### At



**SCHOOL OF ENGINEERING, DESIGN AND AUTOMATION**

### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**MAY 2025**

## CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the project report entitled “ANON-ROUTE” by “AZADVIR SINGH” in partial fulfillment of requirements for the award of degree of B.Tech. (CSE) submitted in the Department of CSE at GNA University, Phagwara is an authentic record of my own work carried out during a period from January 2025 to May 2025 under the supervision of Er. Anchal Nayyar.

#### Signature of Student

#### 

**AZADVIR SINGH**

#### GU-2021-4143

**B.Tech CSE (8A)**

This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.

**Signature of the Internal Examiner**

The B.Tech Viva –Voce Examination of AZADVIR SINGH has been held on and accepted.

**Signature of External Examiner**

**Signature of H.O.D**

**ACKNOWLEDGEMENT**

I would like to place on record my deep sense of gratitude to Er. Anchal Nayyar, Department of Computer Science and Engineering, GNA University, Phagwara for her generous guidance, help and useful suggestions.

I express my sincere gratitude to Dr. Hitesh Marwaha, HOD, Department of Computer Science and Engineering, GNA University, Phagwara, for his stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I am extremely thankful to Dr. Vikrant Sharma, Dean (SEDA-E) for providing me infrastructural facilities to work in, without which this work would not have been possible.

**AZADVIR SINGH**

**GU-2021-4143**

**B.Tech CSE 8(A)**

### ABSTRACT

AnonRoute – Anonymous Traffic Routing Tool

With the current era of digital surveillance and network monitoring, it has been increasingly challenging to protect online anonymity and maintain privacy. Classic options like Virtual Private Networks (VPNs) and proxies give consumers some form of protection but nevertheless depend on a centralized server that can be attacked, monitored, and intervened against by governments. To reduce these threats, Anon-Route has been created—a robust and streamlined tool which takes advantage of the Tor network in order to supply anonymous, encrypted internet traffic to Linux users.

Anon-Route is a command-line application which allows the user to install a transparent proxy via the Tor network, ensuring all outbound traffic, whether an application or a protocol, will pass via Tor. Using Tor's Onion Routing method, the tool ensures that the IP address of the user is concealed and all the traffic is anonymized, such that it is virtually impossible for third parties to track the online activities of the user. The tool works by setting up iptables, which is a utility for establishing firewall rules on Linux, to divert traffic through Tor automatically without having to be manually configured by the user.

The core aim of this project is to offer a user-friendly, lightweight, open-source solution for passing internet traffic via the Tor network. Anon-Route focuses on meeting increasing demands for privacy-oriented tools, which are easy to deploy, secure by nature, and that need little manual intervention from the user. Anon-Route seeks to banish the simplicity of typical VPN configurations and supply an easy command-line interface for users to make their anonymity uncomplicated.

This project offers a concrete, real-life application for cyber security professionals, privacy-oriented users, and researchers who require securing anonymity at the network level and operating using Linux-based systems. It enables the user to enjoy their privacy without relying on central services so they can be obscured from observation or tracking processes. With this project, users come to better know network security, privacy practice, and applying safe communication protocols.

## TABLE OF CONTENTS

CANDIDATE'S DECLARATION ii

[AKNOWLEDGEMENT……………………………………………………………………………………..iii](#_Toc198076039)

[ABSTRACT iv](#_Toc198076039)

[TABLE OF CONTENTS vi](#_Toc198076040)

[LIST OF FIGURES vi](#_Toc198076040)i

[LIST OF TABLES vi](#_Toc198076040)ii

CHAPTER-1 INTRODUCTION 9

[1.1 INTRODUCTION TO ANONYMITY NETWORKS 9](#_Toc198076043)

[1.2 INTRODUCTION TO ANONYMITY NETWORKS 9](#_Toc198076043)

[1.3 INTRODUCTION TO ANONYMITY NETWORKS 10](#_Toc198076043)

[1.4 INTRODUCTION TO ANONYMITY NETWORKS 11](#_Toc198076043)

CHAPTER-2 LITERATURE REVIEW AND RELATED WORK 15

[2.1 INTRODUCTION TO ANONYMITY NETWORKS 15](#_Toc198076043)

[2.2 THE TOR NETWORK 15](#_Toc198076050)

[2.3 EXISTING TOOLS FOR NETWORK ANONYMITY 16](#_Toc198076051)

[CHAPTER-3 SYSTEM ANALYSIS AND DESIGN 18](#_Toc198076052)

[3.1 SYSTEM ARCITECTURE 18](#_Toc198076053)

[CHAPTER-4 IMPLEMENTATION 21](#_Toc198076090)

[4.1 IMPLEMENTATION ENVIORMENT 21](#_Toc198076091)

[4.2 IMPLEMENTATION STEPS 21](#_Toc198076104)

[4.3 SCRIPT MODULARIZATION 29](#_Toc198076168)

[4.4 ERROR HANDELING AND VALIDATIONS 29](#_Toc198076170)

[4.5 SECURITY CONSIDERATION 30](#_Toc198076175)

[4.6 IMPROVEMENTS FROM PREVIOUS TOOLS 30](#_Toc198076180)

[CHAPTER-5 RESULT AND TESTING 31](#_Toc198076181)

[5.1. TESTING OBJECTIVES 31](#_Toc198076183)

[CHAPTER-6 CONCLUSION 35](#_Toc198076184)

[6.1 SUMMARY OF THE PROJECT 35](#_Toc198076185)

[6.2 KEY OUTCOMES 35](#_Toc198076194)

[CHAPTER-7 REFERENCES 37](#_Toc198076199)

**LIST OF FIGURES**

[Figure 1. Architecture of Anon-Route 20](#_Toc198075574)

[Figure 2. Setup commands 22](#_Toc198075575)

[Figure 3. Downloading Repository from GitHub 22](#_Toc198075576)

[Figure 4. Navigating Anon-Route 22](#_Toc198075577)

[Figure 5. Installing Anon-Route 22](#_Toc198075578)

[Figure 6. Command 1 Starting TOR Proxy 22](#_Toc198075579)

[Figure 7. Anon-Route Interface 23](#_Toc198075580)

[Figure 8. Command 2 Clearnet Navigation 23](#_Toc198075581)

[Figure 9. Command 2 Output 24](#_Toc198075582)

[Figure 10. Command 3 Checking Status 24](#_Toc198075583)

[Figure 11 Command 3 Output 24](#_Toc198075584)

[Figure 12. Command 4 IP Information 25](#_Toc198075585)

[Figure 13. Command 4 Output 25](#_Toc198075586)

[Figure 14. Command 5 Showing IP Table Rules 26](#_Toc198075587)

[Figure 15. IP Table Rules 26](#_Toc198075588)

[Figure 16. Command 6 Resource Check 26](#_Toc198075589)

[Figure 17. System Performance 26](#_Toc198075590)

[Figure 18. Command 7 Rotating IP 27](#_Toc198075591)

[Figure 19. Command 7 Output 27](#_Toc198075592)

[Figure 20. Command 9 Restarting TOR services 27](#_Toc198075593)

[Figure 21. Command 9 Output 27](#_Toc198075594)

[Figure 22. Command 10 Version 28](#_Toc198075595)

[Figure 23. Current version of Anon-Route 28](#_Toc198075596)

[Figure 24 Command 9 Show Help Menu 28](#_Toc198075597)

[Figure 25. Help Menu 28](#_Toc198075598)

**LIST OF TABLES**

[Table 4‑0‑1 Script Modularization 29](#_Toc198075608)

[Table 5‑1 Test Case and Results 32](#_Toc198075609)

# CHAPTER-1 INTRODUCTION

1. **PROJECT OVERVIEW**

In the era of the internet, anonymity and privacy have emerged as essential elements of internet access. As surveillance, tracking data, and unwanted access become ever more threatening, users are looking for efficient means to protect their identity and guard their data on the internet. Conventional tools like VPNs offer some level of protection, but they remain dependent on central servers and are vulnerable if the VPN company is malicious or under pressure from the authorities to expose user data.

For their resolution, Anon-Route has been introduced as a tool based on Linux that establishes a transparent proxy through redirecting every internet connection over the Tor (The Onion Router) network. It is set up as a Bash shell script that sets system-level firewall rules and controls the services of Tor automatically. All outgoing traffic is redirected through Tor due to the transparent nature of the proxy independent of the software or protocol and application.

This tool is particularly valuable for cybersecurity professionals, privacy-oriented users, and researchers seeking a lightweight and efficient solution to obtain network-level anonymity without having to depend on third-party software.

1. **MOTIVATION**

The core drive behind Anon-Route is to enable users with a simple, open-source alternative to direct their internet traffic via the Tor network. The rampant rise in digital monitoring, censorship, and cyber attacks requires instruments that foster freedom, security, and privacy.

Most of the tools that already exist are very technical or have complex setup processes. Some of them are not actively maintained, and others come with bloated feature sets at the expense of efficiency. Anon-Route is minimal, efficient, and functional, as it only has the core necessity in mind: routing traffic via Tor at the system level.

Moreover, this project has an academic function—illustrating the practical application of network security, privacy tools, and scripting knowledge, which are prerequisites for any would-be cybersecurity specialist.

1. **OBJECTIVES**

The major objectives of Anon-Route are as follows

* **Develop a Transparent Proxy Tool:**

Develop a command-line application that sets up a transparent proxy on Linux platforms to use the Tor network. With this, all web traffic will be routed through Tor automatically, without users needing to manually set up every individual application for it.

* **Maintain Online Privacy and Anonymity:**

The tool must assure absolute privacy and anonymity by concealing the user's original IP address, leveraging Tor's onion routing to encrypt the user's internet activity at multiple layers.

* **Ease Configuration and Use:**

Offer an easy-to-use tool simplifying the process of setting up Tor-based anonymity for both new and experienced users, without requiring extensive technical understanding of network settings.

* **Automate Tor Integration with iptables**

Implement an automatic integration of iptables, the Linux firewall utility, to reroute all network traffic through Tor so that no traffic avoids the anonymity system.

* **Maintain Cross-Application Anonymity:**

Make the tool functional across all internet-connected applications on the system (browsers, email clients, etc.) by redirecting all traffic through the Tor network, not merely particular programs like web browsers.

* **Create a Strong Security System:**

Deploy security mechanisms to secure against unauthorized attempts at Tor evasion, ensuring traffic stays anonymous every step of the way.

* **Provide a Lightweight and Transparent Solution:**

Design a solution that doesn't hamper the user's system and does not add too much overhead, creating a means for efficiency in acquiring anonymity without losing system performance.

* **Provide Open-Source Accessibility**:

Make the tool open-source so that other users and developers can contribute, customize, and possibly extend the functionality to suit their requirements.

1. **SCOPE OF THE PROJECT**

The AnonRoute project scope is concerned with creating a transparent proxy application meant to boost privacy and security on Linux operating systems using the Tor network. The project aims to meet the requirement for a smooth, automatic solution to pass all internet connections through the Tor network for added anonymity. AnonRoute project scope encompasses the following main factors:

* **Platform Compatibility:**

AnonRoute will mostly be developed for Linux-based platforms like Kali Linux, Ubuntu, and Debian. While the tool can potentially be modified to work on other Unix-like platforms, the first version will concentrate on Linux. Later versions could add support for other platforms based on user feedback.

* **Traffic Routing through Tor:**

The main functionality of AnonRoute is to automatically route all the outgoing internet traffic through the Tor network. This means all the traffic from web browsers, email clients, system updates, and background services, without letting any application or service go around the Tor network. AnonRoute will offer an efficient way of keeping total anonymity.

* **Automatic iptables Configuration:**

AnonRoute will mechanize the installation of iptables firewall rules to capture and redirect network traffic via the Tor network. This makes it unnecessary for users to have to configure their network settings manually, hence making the system accessible, even to users with little technical know-how. The firewall rules will see that all outgoing traffic is passed securely via Tor, thus preventing leaks.

* **Command-Line Interface (CLI)**:

AnonRoute will have a CLI interface, which will enable users to interact with the tool easily. Users can initiate, terminate, and control the transparent proxy, monitor the system status, and set up the tool according to their requirements using the CLI. The CLI makes the tool light and efficient while providing scriptable functionality for power users and automation.

* **Support for All Applications**

One of AnonRoute's main objectives is to make all applications running on the system direct their network traffic through Tor. This implies that web browsers, email clients, system programs, and even processes running in the background that surf the internet will have their traffic encrypted and anonymized. AnonRoute does not intend to limit the anonymization to applications, but provide a system-level implementation of routing through Tor.

* **Anonymity and Privacy Centric**:

The main aim of AnonRoute is to ensure user privacy and anonymity when surfing the internet or accessing web-based applications on the internet. The project will employ Tor's onion routing protocol to encrypt traffic several times at various nodes so that it is practically impossible for anyone to trace the user's original IP address or track their online activities.

* **Performance Optimization:**

Although Tor's routing can add latency, AnonRoute will be designed to ensure that the tool is run with little overhead. By simplifying the configuration and the traffic routing mechanism, AnonRoute will work to limit the effect on system performance, providing users with a compromise between security and usability. The tool will be lean and will prioritize efficiency without compromising the anonymity functionalities.

* **Non-Destructive Changes:**

The changes made to the settings by AnonRoute will be non-destructive. The tool ensures that the system can seamlessly switch back to its default network settings in case the user wants to deactivate the proxy. This capability guarantees that AnonRoute will not disrupt other networking configurations or system activities. Users can disable or enable the tool whenever necessary without fear of making permanent modifications to the system.

* **Security and Integrity:**

AnonRoute will follow best security practices so that the system is secure from unauthorized access. The tool will have provisions to monitor the status of the Tor network and ensure that all traffic is passed securely through it. The system's integrity will be ensured by not allowing unauthorized users to modify AnonRoute or bypass the Tor network.

* **Extensibility and Contribution**:

AnonRoute will be built as an open-source initiative, with the primary intention of creating a community-driven project. The software will be modularized so that other developers can easily enhance its capabilities. Options such as extra routing features, more optimizations, or even cross-platform compatibility might be implemented by contributors in the future.

* **Documentation and Support:**

Thorough documentation will be made available for users, including installation guidelines, configuration guides, troubleshooting hints, and FAQs. The AnonRoute community will be urged to offer assistance and share knowledge, making it simple for users to find a solution to any problems they might face. Moreover, the tool will feature elementary usage assistance in-line within the command-line interface to further facilitate user experience.

# CHAPTER-2 LITERATURE REVIEW AND RELATED WORK

### INTRODUCTION TO ANONYMITY NETWORKS

### The Need for Online Privacy: Online privacy is increasingly a concern in the current digital era. Online privacy is threatened by data harvesting, surveillance, and tracking of activities online. Anonymity networks are needed because of the need to hide sensitive information and be private when accessing the internet.

### Anonymity Networks: Anonymity networks help ensure the users' privacy by hiding their real identities and re-routing traffic via different intermediaries. Typical ones are the Tor network, VPNs, and I2P (Invisible Internet Project) with varying levels of security and performance.

### Difficulties: Anonymity networks encounter multiple difficulties, which include:

### Traffic Analysis: Capacity to deduce user behavior using traffic patterns.

### Correlation Attacks: The potential to correlate in-bound and out-bound traffic to de-anonymize users.

### Performance Problems: Anonymity tends to incur the trade-off of slower browsing speeds as a result of the several layers of encryption and routing.

### THE TOR NETWORK

* **MOverview:** Tor (The Onion Router) is a most popular anonymity network. Tor is meant to grant users strong anonymity by routing web traffic through a sequence of volunteer-run servers (referred to as relays) on the globe.
* **How It Works:**
  + **Onion Routing:** Tor uses a technique known as onion routing, wherein data is layered with encryption and transmitted through a series of relays. Each relay decrypts just one layer of encryption, knowing only the previous hop and next hop in the circuit. This renders it nearly impossible for anyone to track the origin or destination of the traffic.
  + **Multi-layer Encryption:** Packets of data get encrypted many times (similar to the layers of an onion), and every relay decrypts one layer so that no single relay can see both the source and destination of traffic.
  + **Circuit-Based Routing:** Traffic is forwarded over a randomly chosen path of relays so that no single relay knows both the destination and source of traffic.
* **Benefits:**
  + **High Anonymity:** By routing traffic through several relays, Tor makes sure that users' identities are hidden.
  + **Access to Censored Content:** Tor enables users to circumvent censorship and access websites that otherwise remain blocked in some areas.
* **Drawbacks**:
  + **Slow Speeds:** Tor's multi-hop nature can cause slower connection speeds than regular internet browsing.
  + **Weaknesses:** Although Tor grants a great amount of anonymity, it is vulnerable to attacks like traffic correlation or deanonymization by strong adversaries.

### EXISTING TOOLS FOR NETWORK ANONYMITY

* **Tor Browser:** Tor Browser is the most widely used software to access the Tor network. Tor Browser is a customized version of Mozilla Firefox that comes packed with advanced privacy features like blocking scripts, stopping browser fingerprinting, and offering anonymous browsing support.
  + **Strengths:** Is easy to use and gives a good level of anonymity to average internet users.
  + **Weaknesses**: Restricted to browsing and will not give system-wide anonymity.
* **Tails:** Tails is a live operating system that can be started from a USB stick. It directs all internet traffic through Tor and does not leave any trace on the computer after it is turned off.
  + **Pros:** Offers complete system anonymity, which makes it perfect for secure and anonymous use on untrusted machines.
  + **Cons:** Needs a USB stick, and users have to reboot their systems in order to use Tails.
* **Whonix:** Whonix is an anonymity-focused operating system that tunnels every connection through Tor using two virtual machines: Workstation for hosting applications and Gateway for networking.
  + **Strengths:** Good anonymity and application to network isolation.
  + **Weaknesses**: Difficult installation, performance penalty for virtual machines, and incompatibility with all hardware.

# CHAPTER-3 SYSTEM ANALYSIS AND DESIGN

### 3.1 SYSTEM ARCITECTURE

### The architecture of AnonRoute specifies how the various components should interact to ensure the anonymization of all the network traffic that goes out via the Tor network with a transparent proxy configuration. The architecture takes a modular and layered approach so that it can be easily managed, secure, and scalable.

### AnonRoute's architecture may be logically split into the following layers:

### User Interface Layer (Command-Line Interface - CLI)

### This is where the user interacts.

### Implemented with a straightforward and easy-to-use Bash CLI menu.

### Offers options like:

### Start AnonRoute

### Stop AnonRoute

### Check Status

### Check Public IP

### Restart Service

### Show the current IP Table rules

### Rotating IP every 10 second

### Made to enable both beginners and experienced users to manage AnonRoute with ease.

### Script Logic & Control Layer

### This layer consists of the core logic of AnonRoute.

### Responsible for:

### Parsing the user input

### Calling functions according to user commands

### Handling service states (start/stop/restart)

### Printing status messages

### Implemented in Bash completely, to make it compatible with Unix-based systems, particularly Kali Linux.

### Network Configuration Layer

### Maintains system-level firewalls rules using iptables.

### Sets the rules to:

### Send all TCP traffic (but not Tor and DNS) through to the Tor proxy port (default 9040)

### Suppress DNS leaks by forwarding DNS requests through to the Tor DNS port (default 5353)

### Whitelist services necessary such as loopback (lo) and Tor's native UID

### Avoids any packet skipping the Tor network unless bypassed.

### Tor Network Layer

### Carries out true anonymization by passing traffic over multiple Tor relays.

### Operates the Tor service daemon, which has to be installed on the system.

### Ensures that:

### Selection of entry, relay, and exit nodes occurs automatically.

### Traffic is encrypted within the Tor network.

### The user's original IP address is never revealed.

1. **System Services & Dependencies Layer**

* This layer contains all the external services AnonRoute needs to operate:
  + Tor daemon (must be installed and running)
  + iptables (default on most Linux systems)
  + Bash shell environment
* Makes sure these services are correctly configured and running before permitting AnonRoute to launch.

**Diagram: Simplified Architecture of AnonRoute :**

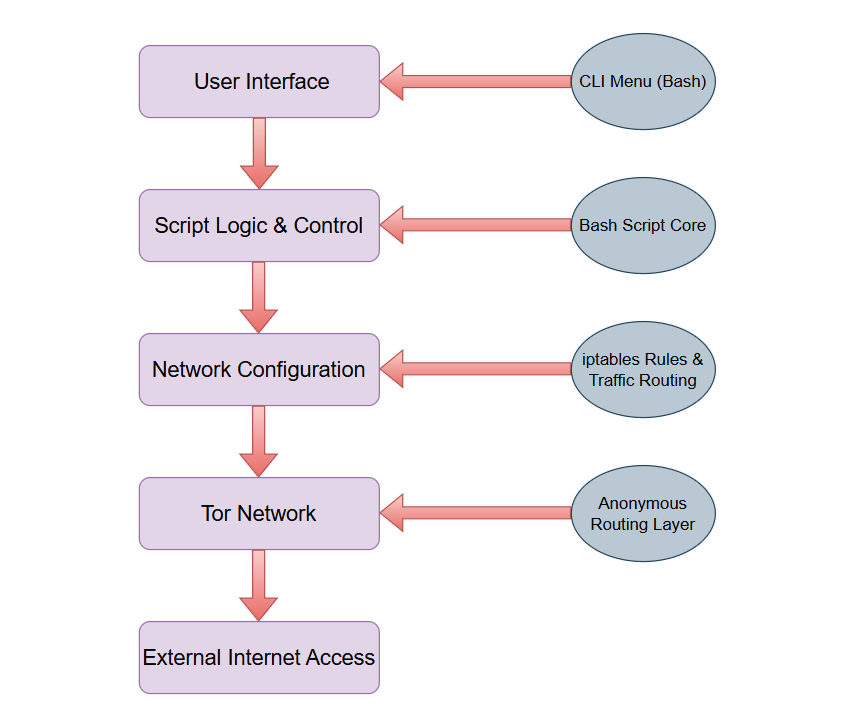
****

Figure 1. Architecture of Anon-Route

**Design Principles Followed**

* Transparency: Individual applications not configured manually.
* Security: No traffic ever exits the machine outside of the Tor tunnel.
* Modularity: Each component (UI, firewall, Tor) can be upgraded separately.
* Simplicity: Simple to execute and service with one script file.

# CHAPTER-4 IMPLEMENTATION

The deployment phase of the AnonRoute project is converting the design architecture into a functional solution by employing Bash scripting on Linux. This chapter documents a detailed description of how different parts of the system were implemented, integrated, and tested to produce secure and anonymous routing of all traffic via the Tor network.

### IMPLEMENTATION ENVIORMENT

### The creation and testing of AnonRoute took place in a reproducing and controlled setting for reliability and compatibility assurance.

### Platform

### Operating System: Kali Linux 2023.4 (penetration testing Debian-based distro)

### Kernel Version: Linux kernel 6.x

### Environment: VMware Workstation / VirtualBox virtualized environment

### Execution Shell: GNU Bash (version 5.x)

### Tools and Utilities

### Tor – Executed as the principal anonymity network utility.

### iptables – Utilized for redirection of traffic and firewall configuration.

### netfilter-persistent – (Optional) To persist iptables rules on reboot.

### whois / curl / wget – For IP verification and web requests.

### ANSI escape codes – For colored, formatted CLI output.

### IMPLEMENTATION STEPS

### The process of implementing AnonRoute is divided into several functional modules, each handling a particular feature of the tool.

### ****Project Setup****

### Installed all required dependencies using:

### 

Figure 2. Setup commands

### Verified that the tor service is enabled and properly configured to listen on:

### TransPort: 9040

### DNSPort: 5353

### Downloading AnonRoute From Github

### 

Figure 3. Downloading Repository from GitHub

### Enter Anon-Route Directory and Install it

### 

Figure 4. Navigating Anon-Route

### 

Figure 5. Installing Anon-Route

### Commands

### Start transparent proxy through Anon-Route

### <sudo anonrote -t> or <sudo anonroute --tor>

### 

Figure 6. Command 1 Starting TOR Proxy

### OUTPUT:

### 

### 

Figure 7. Anon-Route Interface

### Reset iptables and return to Clearnet navigation

### <sudo anonrote -c> or <sudo anonroute --clearnet>

### 

Figure 8. Command 2 Clearnet Navigation

### OUTPUT:

### 

Figure 9. Command 2 Output

### Check status of program and services

### <sudo anonrote -s> or <sudo anonroute --status>

### 

Figure 10. Command 3 Checking Status

### OUTPUT

### 

Figure 11 Command 3 Output

### Show public IP address

### <sudo anonrote -i > or <sudo anonroute --ipinfo>

### 

Figure 12. Command 4 IP Information

### OUTPUT

### 

Figure 13. Command 4 Output

### Display current IP Table Rules

### <sudo anonrote -tb> or <sudo anonroute --table>

### 

Figure 14. Command 5 Showing IP Table Rules

### OUTPUT

### 

Figure 15. IP Table Rules

### Display system's CPU, RAM, and disk usage performance

### <sudo anonrote -rc> or <sudo anonroute –resource-check>

### 

Figure 16. Command 6 Resource Check

### OUTPUT

### 

Figure 17. System Performance

### Rotate the Tor IP automatically after 10 seconds

### <sudo anonrote -rip> or <sudo anonroute –rotate-ip>

### 

Figure 18. Command 7 Rotating IP

### OUTPUT

### 

Figure 19. Command 7 Output

### Restart tor service and change IP address

### <sudo anonrote -r> or <sudo anonroute --restart>

### 

Figure 20. Command 9 Restarting TOR services

### OUTPUT

### 

Figure 21. Command 9 Output

### Display program version and exit

### <sudo anonrote -v> or <sudo anonroute --version>

### 

Figure 22. Command 10 Version

### OUTPUT

### 

Figure 23. Current version of Anon-Route

### Help Message for learners

### <sudo anonrote -h> or <sudo anonroute --help>

### 

Figure 24 Command 9 Show Help Menu

### OUTPUT

### 

Figure 25. Help Menu

### SCRIPT MODULARIZATION

Table 4‑0‑1 Script Modularization

### 

### ERROR HANDELING AND VALIDATIONS

### Dependency Checks: The user is notified and the script terminates if tor or iptables is not found.

### Root Privilege Check: The script inspects for root privileges and refuses to run if it is not run as root or with the use of sudo.

### Tor Service Failure: The script notifies the user if the Tor service doesn't start, instructing the user to check the service logs manually.

### Invalid Input: If the user has entered an invalid menu option, a warning message is shown and the menu is refreshed.

### SECURITY CONSIDERATION

### All non-Tor outgoing traffic is blocked so that there will be zero leaks.

### DNS leaks are avoided by forwarding DNS requests through Tor's DNSPort.

### Whitelisting just Tor's UID prevents an attempt to establish a routing loop for Tor's own traffic.

### The script won't store user data or log anything, making it private

### IMPROVEMENTS FROM PREVIOUS TOOLS

* Compared to tools like Kalitorify or toriptables2, **AnonRoute**:
  + Uses reduced logic and fewer dependencies.
  + Delivers immediate status feedback.
  + Maintenance of modular functions is significantly easier.
  + Includes additional functionality such as restart and IP validation.

# CHAPTER-5 RESULT AND TESTING

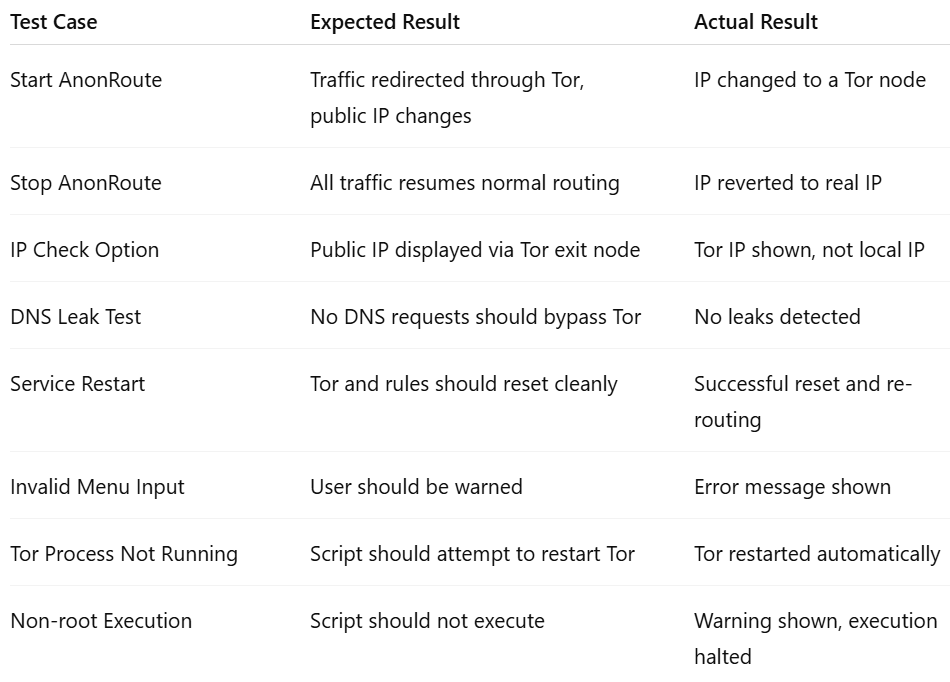
### This chapter introduces the results of deploying the AnonRoute project. The main objective of the test is to verify that all the network traffic that goes out is routed properly through the Tor network and that there are no DNS or IP leaks from proxy usage. The test results also assess the user experience and tool reliability in various network environments.

### TESTING OBJECTIVES

The testing process aimed to:

* + Verify proper redirection of network traffic through Tor.
  + Detect any DNS or IP leaks.
  + Evaluate script behavior for all user options.
  + Check Tor service status management.
  + Ensure the tool blocks all non-Tor traffic.
  + Test cross-platform (different Linux versions) compatibility
  1. **TESTING ENVIORMENT**
* The testing procedure intended to:
* Host OS: Windows 11 Pro
* Virtual Machine: Kali Linux 2023.4
* Network: NAT and Bridged Adapter testing
* Tools Utilized:
  + curl ifconfig.me
  + dnsleaktest.com
  + whois
  + iptables -L
  + pgrep tor, systemctl status tor
  + check.torproject.org
  1. **TEST CASE AND RESULTS**

Table 5‑1 Test Case and Results



* 1. **PERFORMANCE OBSERVATIONS AND LIMITATIONS**

**Performance Observations**

* **Anonymity Efficiency:**

AnonRoute was successful in anonymizing outgoing traffic by sending it through the Tor network. Tests revealed consistent masking of IP, with public IP addresses consistently rotating to various Tor exit nodes.

* **Latency:**

As a function of the inherent nature of Tor routing, latency is increased inherently. This could be seen while performing DNS queries and HTTP requests, which took longer than on direct connections.

* **System Resource Usage:**

The script exhibited low CPU and RAM usage. Tor service was running in the background with constant memory usage, which made it ideal for extended anonymous sessions.

* **Security Enforcement:**

Iptables rules worked perfectly in enforcing full redirection via Tor, blocking traffic leaks. DNS leak tests validated the system to have DNS privacy.

* **Restart and Recovery:**

The tool showed robust behavior on restart. When stopped and started again, the system re-initialized Tor and firewall rules cleanly, confirming good operational reliability.

**Limitations**

* **Speed Constraints:**

The Tor network is not optimized for high-speed data transfer. Users may experience slow download speeds or connection delays when using AnonRoute for streaming or file transfers.

* **Dependency on Tor Network:**

AnonRoute relies solely on Tor network availability and operability. When Tor services are down or intermittently unavailable, the tool cannot achieve anonymity.

* **Root Access Requirement:**

The script requires execution by the root user, which might not be possible in security-limited environments or unprivileged accounts.

* **Limited Traffic Routing Scope:**

While outbound traffic is anonymized, AnonRoute does not support hidden services or incoming connections through Tor. It is designed primarily for secure outbound browsing.

* **No Persistent Logging:**

For privacy reasons, the tool does not maintain logs. However, this also means limited post-analysis is possible for debugging or forensic tracking.

* **Platform Specificity:**

The script is tailored for Linux environments (especially Debian-based distributions like Kali Linux). Porting to Windows or macOS would require substantial modifications.

# CHAPTER-6 CONCLUSION

# SUMMARY OF THE PROJECT

# The AnonRoute project was created with the main objective of facilitating anonymous web surfing by directing all internet traffic through the Tor network via a transparent proxy mechanism. The tool was written as a Bash shell script specifically for Kali Linux, which automated the setup of iptables, the Tor service, and DNS leak protection to provide privacy and anonymity.

# During this project, the tool exhibited features such as:

# Autmatically launching and closing the Tor-based proxy.

# Redirecting all TCP traffic to use Tor.

# Avoiding DNS leaks by making DNS resolution occur through Tor.

# Enabling checking the users current public IP to confirm anonymity.

# Having a lightweight and easy-to-use command-line interface.

# The project was successfully completed and shown to be useful in enhancing privacy on the Internet by hiding the actual IP address and blocking leakage of traffic out of the Tor network.

# KEY OUTCOMES

# Secure and Anonymous Routing: AnonRoute offered a trustworthy means of anonymizing internet traffic via Tor.

# Automated Setup: All system-level settings required for the tool were automated, so it was available even to users with minimal technical expertise.

# DNS Leak Protection: By sending DNS requests via Tor, AnonRoute prevented DNS leaks that would reveal user anonymity.

# User Control: The software gave users the convenience to initiate, shut down, reboot, and observe their anonymous routing configuration seamlessly.

* 1. **FUTURE WORK**
* **Graphical User Interface (GUI):** Including a GUI frontend for simpler interaction.
* **Multi-platform Support:** Expanding support to other Linux distributions and even Windows with WSL.
* **Advanced Traffic Routing:** Allowing user-specified applications to route selectively over Tor while leaving others on the normal network
* **Integration with VPN:** Merging Tor with VPN layers for increased anonymity.
  1. **FINAL WORDS**

AnonRoute is a solid foundation for privacy-oriented users and researchers who seek to investigate secure traffic routing through open-source networks such as Tor. As privacy concerns expand around the world, technologies such as AnonRoute help create a protective barrier against surveillance, tracking, and censorship. This project not only fulfills technical requirements but also helps promote the ethical significance of digital freedom.

# CHAPTER-7 REFERENCES

[1] R. Dingledine, N. Mathewson, and P. Syverson, “Tor: The Second-Generation Onion Router,” *Proceedings of the 13th USENIX Security Symposium*, San Diego, CA, USA, Aug. 2004. [Online]. Available: https://www.usenix.org/events/sec04/tech/dingledine.html

[2] The Tor Project, “Tor Manual Pages,” *Tor Project Documentation*, 2023. [Online]. Available: <https://2019.www.torproject.org/docs/tor-manual.html.en>

[3] A. Murdoch and P. Zieliński, “Sampled Traffic Analysis by Internet-Exchange-Level Adversaries,” *Workshop on Privacy in the Electronic Society (WPES)*, 2007, pp. 3–14. doi: 10.1145/1314333.1314336.

[4] Netfilter.org, “iptables - Administration Tool for IPv4 Packet Filtering and NAT,” *The Netfilter Project*, 2023. [Online]. Available: https://netfilter.org/projects/iptables/index.html

[5] L. Kaufman, “Anonymity in the Internet: Tor’s Role and Challenges,” *IEEE Security & Privacy*, vol. 16, no. 6, pp. 70–73, Nov./Dec. 2018. doi: 10.1109/MSEC.2018.2873866.

[6] D. E. Denning, “The Limits of Wiretap Protection,” *IEEE Security & Privacy*, vol. 1, no. 5, pp. 69–71, Sep.–Oct. 2003. doi: 10.1109/MSECP.2003.1236234.

[7] M. Herrmann, C. Banse, and E. Holz, “Towards Measuring Anonymity: Comparing Anonymity Networks from a Users Perspective,” in *Proceedings of the 10th International Conference on Availability, Reliability and Security*, Toulouse, France, 2015, pp. 510–517. doi: 10.1109/ARES.2015.68.

[8] J. Hong, “The State of Privacy Protection in the Internet of Things,” *IEEE Security & Privacy*, vol. 17, no. 3, pp. 55–59, May/June 2019. doi: 10.1109/MSEC.2019.2906370.

[9] Kalitorify Project (Archived), “A Transparent Proxy Through Tor for Kali Linux,” GitHub Repository. [Online]. Available: <https://github.com/brainfucksec/kalitorify>

[10] Linux Foundation, “iproute2 – Linux Network Configuration,” *Linux Foundation Documentation*, 2023. [Online]. Available: https://man7.org/linux/man-pages/man8/ip.8.html