Hindawi
Security and Communication Networks
Volume 2022, Article ID 5832124, 10 pages
https://doi.org/10.1155/2022/5832124

Research Article

LUNAR: A Practical Anonymous Network Simulation Platform

Xianchun Zheng,[1] **Haonan Yan**,[1] **Rui Wang**,[1] **Ziwei Zhang**,[2] **and Hui Li**[1]

[1]*State Key Laboratory of Integrated Service Network, School of Cyber Engineering, Xidian University, Xi'an, China*
[2]*School of Cyber Science and Engineering, Wuhan University, Wuhan, China*

Correspondence should be addressed to Hui Li; lihui@mail.xidian.edu.cn

Received 25 February 2022; Revised 1 April 2022; Accepted 17 April 2022; Published 4 May 2022

Academic Editor: Jinbo Xiong

With the expansion of cyberspace and the increasing importance of users for privacy protection, anonymous network research has been further developed, especially for the numerous internet of things (IoT) devices. However, when we repeat the existing experiment in an anonymous network, there are many problems such as too old version, low realism, poor operability, and so on. In this paper, we analyzed the design requirements and topology of the new experimental platform. Two topologies with different levels of complexity are designed. We also set up a practical anonymous network simulation platform called LUNAR with virtualization, software-defined networking (SDN), and other technologies to solve those problems. The platform we proposed supports multiprotocol and reproducible complex networks with centralized management. Finally, we implement our simulation platform and reproduce two typical attacks, that is, time-linked Tor node reset attack and website fingerprint attacks on The Onion Router (Tor) network, to evaluate the platform. Experiments results indicate the practicality and superiority of our simulation platform in terms of anonymous network simulation.

1. Introduction

Anonymous network is a privacy protection technology that can protect the private information of the users and the service providers [1, 2]. It can hide the true identity and location of one or both parties in the communication so that the attacker cannot know the sender and receiver of the data, decrypt the plaintext information or associate the transmission of information between the sender and the receiver. The global anonymous network represented by Tor [3] and I2P [4] is widely used in the fields of anonymous access, anonymous communication, and hidden services [5, 6]. These networks have millions of daily users and thousands of relay nodes. Users use anonymous network such as Tor to conduct web pages, browse, online communication, and virtual trade. Due to the characteristics of anonymity, the dark web [7] has two sides. On the one hand, it can be used to protect the privacy of Internet users; on the other hand, it can also be concealed criminal traces or other malicious behavior [8].

1.1. Related Work. Since the birth of the anonymous network, the research on its security about the simulation platform has never stopped [9–15]. For example, [16] proposes a method for connecting IoT devices in a client-server configuration to utilize the Tor network for addressing and secure communication between IoT and CPS devices. Reference [17] introduces a new approach to exploiting Tor's anonymous communication to handle distributed attacks against smart devices on the Internet and demonstrates the effectiveness of Tor in IoT devices. Reference [18] designs a higher bandwidth Onion IoT gateway to provide robust security protection for vulnerable IoT devices hidden behind an IoT gateway. Reference [19] develops, investigates, and evaluates the performance of machine-learning-based darknet traffic detection systems (DTDS) in IoT networks. Reference [20] presents deep learning recurrent LSTM-based technique to classify the traffic over IoT-cloud platforms. Reference [21] realizes resource-conserving access control and end-to-end security for IoT devices and deploys onion routing for the IoT within

the well-established Tor network enabling IoT devices to leverage resources to achieve the same grade of anonymity as readily available to traditional devices. SDN [22] can decouple the control plane from the data plane of forwarding devices, and [23] proposes an SDN-based solution to mitigate the privacy threats by anonymizing both MAC and IP addresses. Reference [24] studies ARP spoofing attacks based on SDN technology. However, this experiment is only limited to the ARP protocol and does not discuss other protocols or networks. Reference [25] simulates the dark network scene on the OpenStack platform to detect dark network resources and obtain dark network information. Reference [26] proposes a novel methodology for constructing a real private tor network by editing and controlling Raspberry Pis. However, there is little research on anonymous network simulation platforms design, mostly focusing on improving existing platforms. Reference [27] shows an anonymous network named Crowds. They use Peersim [28] for Crowds anonymous network simulation. In this kind of network, researchers can conduct anonymous tracking. But the author did not carry out a comprehensive design for different forms of networks. Reference [29] designs and implements a new Tor network simulation model based on Shadow [30]. The current simulation platforms have more or less different shortcomings. Some existing platforms are no longer updated, or they are currently no version available to use. Almost all the platform can only simulate or emulate part of establishing a network connection, which is not suitable for various simulation requirements. At the same time, the modification cost is high and has an impact on the simulation results. Table 1 are statistics for Tor network security research.

1.2. Our Work. In this paper, we design and set up a practical anonymous network simulation platform called LUNAR. The overall design principle of LUNAR is to establish an anonymous network simulation platform with convenient operation, large scale, and a high degree of reality under feasible resource constraints. Besides, the platform can ensure the correctness of the Tor network simulation operation, ensure that the virtual nodes can be linearly expanded, and form a large-scale simulation platform.

Our main contributions are as follows:

(1) We summarize all experiments conducted on anonymous networks and classify them into five categories in Section 2.1, from which we find six network topology requirements of the simulation platform in Section 2.3.

(2) We provide five design requirements for the current new anonymous network simulation platform in Section 2.2.

(3) We devise two kinds of an anonymous network topology for our simulation platform, that is, a basic one for a single small-scale experiment and a more complex one for a complex anonymous network experiment in Section 3.1. Our proposed simulation platform is also compared with five previous ways to conduct anonymous network experiments in Section 2.1.

(4) We implement our highly scalable simulation platform and reproduce two types of typical attacks, that is, time-linked Tor node reset attacks and website fingerprint attacks on the Tor network, on our simulation platform in Section 4. Experiments indicate our simulation platform can provide a practical anonymous network environment.

2. Problem Description

2.1. Compare with Existing Platform. The existing methods for conducting anonymous network experiments are roughly divided into several categories. Table 2 is the comparison of the advantages and disadvantages of five experimental platforms.

(1) Experiments in real networks. The advantage is that the threshold for conducting small-scale experiments is low, and the realism is high. But it can only provide local experiment results, observation angles, and control methods, and it is impossible to implement a global experiment scenario [31].

(2) Experiments with a large research network such as PlanetLab [32]. It consists of more than 1,000 servers distributed around the world. It can apply to several servers to form a network segment and deploy experiment software on it. But the operability is still limited, and experiments can only be carried out in a limited time frame and scale [31].

(3) Simulation experiments, such as ExperamenTor [33]. It uses ModelNet [34] as a virtual machine and network simulation to realize the experiment environment. Although the efficiency is partially improved, because of the age, we cannot download the available version.

(4) Emulation test, such as TorPs [35]. It simulates the process of the Tor network selection relay node to establish a link. TorPs is suitable for experiments related to improving or changing the link selection algorithm, but not applicable to operational experiments.

(5) Semi-simulation and semi-emulation, such as Shadow [30]. It implemented a network layer emulation and application layer simulation test environment. However, the use of Shadow as a research platform requires a certain amount of modification cost, and the modification will directly affect the efficiency, accuracy, and intuitiveness of the experiment results.

Compared with our design, these simulation platforms make different trade-offs in extensibility, global control, operability, and cost, which cannot meet the higher requirements of modern new experiments. For example, Shadow implements a low-level takeover to meet scalability, but it is less efficient.

Table 1: Statistics for Tor network security research.

Research	Attack	Defense
Link attack	Censorship, BGP attack, path selection, bridge attack	Censorship avoidance, timing-based avoidance, path selection algorithm, guard node selection algorithm
Traffic analysis	Bridge discovery attack, replay attack, man-in-the-middle attack, traffic correlation attack	Data mining, trust mechanism
Website fingerprinting attack	Based on website characteristics, based on website cache	Service site redesign, cache mask
Others	Side channel attack, DDoS, information leakage	Privacy information protection

Table 2: Comparison of the advantages and disadvantages of five experimental platforms.

Methods	Typical representative	Advantages	Disadvantages
Real networks	Real tor network	Low threshold and high realism	Cannot control the whole, high cost of large-scale case
Global research network	PlanetLab	Apply for several servers to form a network segment and deploy experiment software	Operability, time frame, and scale are limited
Simulation experiments	ExperamenTor	High overall control and low experimental deployment cost	No available version
Emulation test	TorPs	Suitable for experiments related to improving or changing the link selection algorithm	Unable to analyze traffic and test security
Semi-simulation and semi-emulation	Shadow	Offering a network layer emulation and application layer simulation test environment	High modification cost and modification affect results

2.2. Design Goals. Combined with the existing simulation platform analysis, the new platform needs to meet the following design requirements:

(1) Support multiprotocol. The experiment platform should support the environment reproduction of Tor, I2P, Freenet, and other anonymous networks designed or modified by themselves and can test and demonstrate the functions, performance, and security of these protocols.

(2) Support native code. The simulation platform should directly install and run the original code or installation package of each anonymous network software and its modified variants. Ensure the accuracy of the simulation results.

(3) Reproduce network conditions. The simulation platform can simulate different link bandwidths and network congestion, support background traffic, and support network monitoring, interference, and control at different levels (node level, network segment level, and self-made domain level). This satisfies the researchers' needs for each dimension of the network layer.

(4) Save resources. Although the current hardware resource cost is getting lower and lower, the simulation platform's node size has also been greatly improved. As a result, it is still necessary to prevent the hardware cost from exploding with the scale expansion, making the demand linear with the node size.

(5) Centralized control and observation. In order to achieve the global perspective simulation, the platform needs to have centralized management configuration means and a unified result observation method, which can be used by researchers to use the platform to carry out experiments to lower the threshold and provide convenience.

2.3. Platform Functions. Experiments on standard anonymous networks can be summed up in the following types.

2.3.1. Performance Analysis. Tor's anonymous network has been criticized for its network performance due to the particularity of its protocol [36, 37]. Researchers have been looking for ways to improve the Tor network's overall performance, either redesigning the protocol and architecture, optimizing Tor's own weighted link selection algorithm, or enhancing its handling of network congestion, packet encryption, and decryption at the application layer.

2.3.2. Passive Listening. In passive listening, the attacker can control the critical location nodes such as the malicious guard node and exit node, hijack the traffic, carry out the relevant time attack, fingerprint attack, and other attack means to achieve the goal of deanonymization. Traffic feature extraction and matching are often vital in passive listening.

2.3.3. Active Interference. Active interference refers to the attacker's artificial change of the link establishment process, traffic path selection, and other key points to remain anonymous. For example, an attacker at the AS

(autonomous system) level can change the network traffic of the Internet to achieve more granular traffic filtering by implementing BGP hijacking and asymmetric traffic analysis to obtain more accurate analysis results.

2.3.4. Denial of Service. The denial of service is an attack on the Tor network itself. The target of these attacks is often the relay node. Since the overhead of the relay node processing the data packet is much larger than the overhead of generating the data packet, the attacker can continuously send the CREATE data packet to consume the relay node's computing resources, thereby reducing the performance of the entire Tor network.

2.3.5. Application Layer Induction. In addition to the underlying protocol in the overall composition of the Tor network, the entire application layer system also exists in the surface web website. Browsers may have loopholes and problems, so it is also necessary to test and audit the anonymous network's service provider. Security vulnerabilities may occur on it.

3. Our Proposed Platform

3.1. Topology Design. We summarize the common anonymous network attack and defense experiment scenarios, extract the rule conditions that satisfy them, and design a standard network topology, which can project various simulation scenes into the topology without distortion. We use a white spot as the terminal node and use a black spot as the basic network node. Therefore, we design a basic topology that meets different conditions.

The topology shown in Figure 1 consists of N routing nodes and N terminal nodes. The routing nodes are connected end to end to form a ring network. Each routing node is externally connected with a terminal node. The routing node acts as the control node of the termination node. The control node of the point can realize monitoring and flow control. Adjacent routing nodes can form an autonomous system to reproduce the characteristics of the anonymous network in the home domain. After the ring network is destroyed on a link, another half of the line can be dynamically routed. This topology simplifies the network structure and node relationships but retains the various network features required for anonymous network experiments and can be used for a single small-scale experiment.

In addition, we have designed a more complex network topology, as shown in Figure 2. The topology separates the routing node used to monitor the terminal traffic from the basic routing node used to transmit information and is connected by an N-route node in a star topology to form a basic transmission network, each of which is on the basic routing node. A routing node is dedicated to the control node, and M terminal nodes are connected to the control node to form an autonomous system. In the experiment, the basic routing nodes are not touched, and only the terminal nodes and control nodes required by the simulation environment are operated. This topology is closer to the real

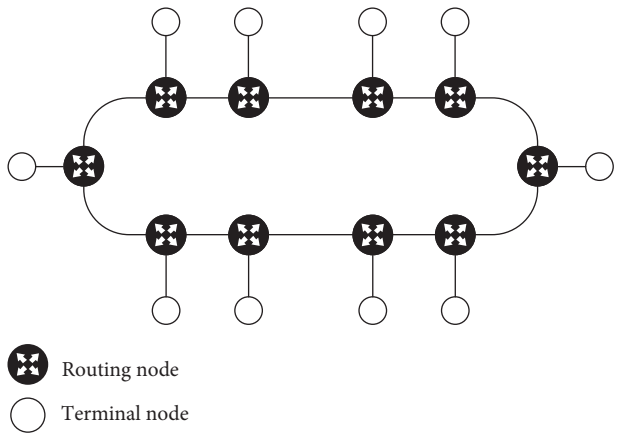

Figure 1: The basic topology of our platform.

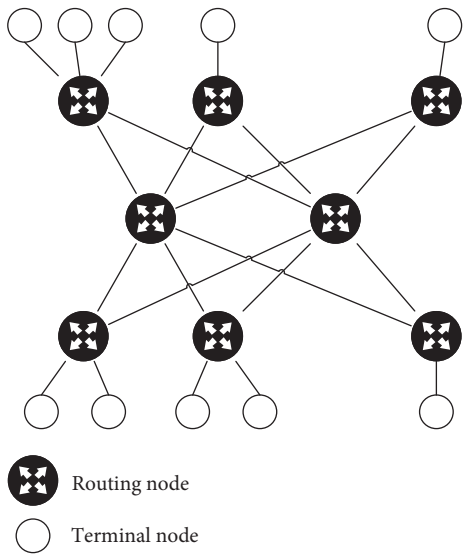

Figure 2: A more complex topology of our platform.

Internet environment, ensuring the availability of the underlying network for complex anonymous network experiments.

The focus of our design is not on traffic collection and stratification but on the design of the topology. Traffic in the protocol layer is the problem of upper-layer applications. Further research applications such as Tor can be deployed on our proposed topology.

3.2. Advantages. Our platform's network topology has the following advantages:

(1) It contains the basic network node responsible for transmission. The basic network node is equivalent to the infrastructure on the Internet and is only responsible for forwarding network packets, not participating in the encryption and decryption of anonymous networks.

(2) It runs anonymous network programs in the terminal nodes. Terminal nodes may assume the role of

dark web infrastructure such as hidden service directory, may also act as users of anonymous networks, run anonymous service-side programs or client services, and may also be used as a simulation of various services in the surface web so that experimenters from the anonymous network to initiate access to them, such as DNS, Web Server, etc.

(3) It interworks between any two nodes. The same experiment instance is performed in an interworking network. If multiple simulation experiments are required at the same time, numerous network instances can be pulled up.

(4) No single node can carry all the traffic in the network. The simulation platform should avoid a situation where the entire network is down due to the failure of the single node that carries all the traffic.

(5) The communication between any two terminal nodes has no less than two routing lines. The network infrastructure under the dark web (usually the Internet) has certain robustness and will not be faulty due to a single node. And it leads to the whole network.

(6) All communication traffic of any terminal node needs to flow through at least one corresponding basic network node. In the simulation experiment for the anonymous network, there may be scenarios in which the traffic of the terminal node is monitored or controlled in the system (rather than being handled by the node itself), such as being controlled by the operator. The node in this topology can implement this condition, which we call the control terminal node.

4. Evaluation

As the simulation platform is virtualized by KVM (Kernel-based virtual machine) and SDN network layer, each node's operation is consistent with that of the actual Tor network. No additional functions or redundant configuration is required. Log in to the corresponding node shell to perform steps such as compiling and running Tor, starting the Tor process, and changing the Tor configuration. This section mainly introduces the platform's feasibility and superiority by recreating the simulation approach of two typical attacks on our testing platform.

4.1. Experiment Setup. In order to achieve the design goals of the topology and simulation platform, we choose to use a host virtualization technology to create virtual nodes on a set of server clusters and use SDN technology to connect nodes according to the designed topology to form a virtual network, running different software. The network traffic sent by the virtual machine is transmitted by the SDN controller, using the same communication protocol (IP, TCP, and UDP) as the real network, and finally sent to receive the traffic. The design of the simulation platform is shown in Figure 3.

The simulation platform is built on a server cluster connected by LAN. KVM is used as the virtual machine controller; OVS (Open vSwitch) is used as the SDN switch; and CentOS is used as the operating system on both the server and the virtual machine. The foundation of the experimental platform is composed of KVM, OVS, and CentOS. The routing node is responsible for data forwarding at the network layer, and the complete Tor process is run in the terminal node. For different Tor roles, such as directory servers, we can flexibly allocate more resources to meet its computing and storage requirements.

The version of Tor selected for this paper is tor-0.3.4.8. System version is CentOS, and Linux version is 3.10.0-693.el7.x86 64. The simulation platform designed in this paper provides a related server and virtual machine images.

The platform provides a set of scripting tools, as the management layer of the simulation platform, which realizes the generation and configuration of the virtual machine and virtual network topology, pulls up and manages instances of the simulation environment, controls virtual machine terminals, and changes network bandwidth and congestion.

Another set of scripts implements the business layer of the simulation platform. It can install and configure the basic components of classic anonymous networks such as Tor, I2P, and Freenet and build an independent and complete anonymous network simulation environment. On this basis, the experimenter can also log in to the virtual machine to install other servers or client software and access the anonymous network. Figures 4 and 5 are examples of OnionRoute and HiddenService configuration.

Finally, the platform provides an observation layer. The participants can obtain the control of the corresponding terminal node or routing node according to the scenario setting and role assignment and specify the network traffic mirroring and log information of the location.

4.2. Time-Linked Tor Node Reset Attack

4.2.1. Attack Fundamentals. We divide the link in the process of communication between the client and the anonymous server into several parts for the subsequent explanation. The first is the link from the client to the rendezvous node through the 2-hop node, which we collectively call the CR link part. The link from the anonymous server to the rendezvous node through the 3-hop node is collectively referred to as the HR part.

In the dark web environment, only the hidden server's entry node knows the real IP of the anonymous service. Suppose we currently have a certain number of entry nodes, clients, and rendezvous nodes.

The attack is divided into three steps:

(1) Open the Tor2Web mode at the client we control. Select the RP that we control as the rendezvous node. The client sends a request to a specific anonymous server continuously for a certain period, and the anonymous server further establishes long-link communication with the client.

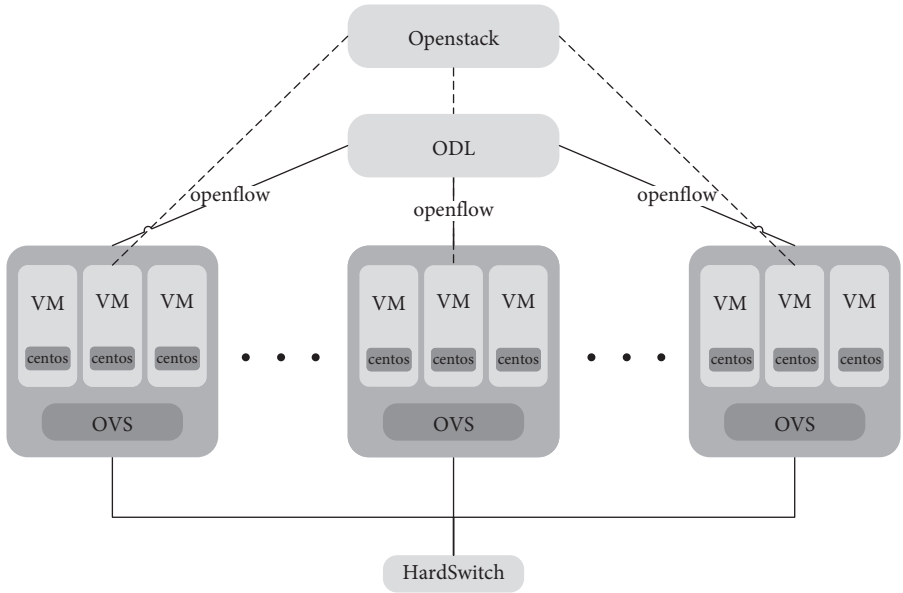

Figure 3: Simulation platform design.

```
TestingTorNetwork 1
DataDirectory /opt/tor/lib/
RunAsDaemon 1
ConnLimit 60
Nickname ORleaf23
ShutdownWaitLength 0
PidFile /opt/tor/lib/pid
Log notice file /opt/tor/log/notice.log
Log info file /opt/tor/log/info.log
Log debug file /opt/tor/log/debug.log
ProtocolWarnings 1
SafeLogging 0
DisableDebuggerAttachment 0
DirAuthority AS1leaf1   orport=5000 no-v2 hs v3ident=17CF1F16C3FE0FDDEBA2EC91BC205D716CC154B8 10.0.1.101
:7000 3715350A7D4BBB107F8A787646ED9A82BC4A1514
DirAuthority AS2leaf11 orport=5000 no-v2 hs v3ident=E70E228A58847B4A82299DF9274073BD4188D7FE 10.0.11.101
:7000 BA16379B767D6BB72BDFA6AC4726ADB2AFFE6DF2
DirAuthority AS3leaf21 orport=5000 no-v2 hs v3ident=58A5A331EB5899C5E0E1BFC7E4066760F13A1663 10.0.21.101
:7000 BDDB1055538DAE719FD0B1B3A8D50209E5853630

SocksPort 0
OrPort 5000
ControlPort 9051

Address 10.0.23.101
```

Figure 4: Example of OnionRoute configuration.

```
TestingTorNetwork 1
DataDirectory /opt/tor/lib/
RunAsDaemon 1
ConnLimit 60
Nickname HSleaf30
Address 10.0.30.101
ShutdownWaitLength 0
PidFile /opt/tor/lib/pid
Log notice file /opt/tor/log/notice.log
Log info file /opt/tor/log/info.log
Log debug file /opt/tor/log/debug.log
#UseEntryGuards 0
ProtocolWarnings 1
SafeLogging 0
DisableDebuggerAttachment 0
DirAuthority AS1leaf1   orport=5000 no-v2 hs v3ident=17CF1F16C3FE0FDDEBA2EC91BC205D716CC154B8 10.0.1.101
:7000 3715350A7D4BBB107F8A787646ED9A82BC4A1514
DirAuthority AS2leaf11 orport=5000 no-v2 hs v3ident=E70E228A58847B4A82299DF9274073BD4188D7FE 10.0.11.101
:7000 BA16379B767D6BB72BDFA6AC4726ADB2AFFE6DF2
DirAuthority AS3leaf21 orport=5000 no-v2 hs v3ident=58A5A331EB5899C5E0E1BFC7E4066760F13A1663 10.0.21.101
:7000 BDDB1055538DAE719FD0B1B3A8D50209E5853630
HiddenServiceDir /opt/tor/lib/hidden_service/
HiddenServicePort 80 127.0.0.1:80
SocksPort 0
OrPort 5000
ControlPort 9051
```

Figure 5: Example of HiddenService configuration.

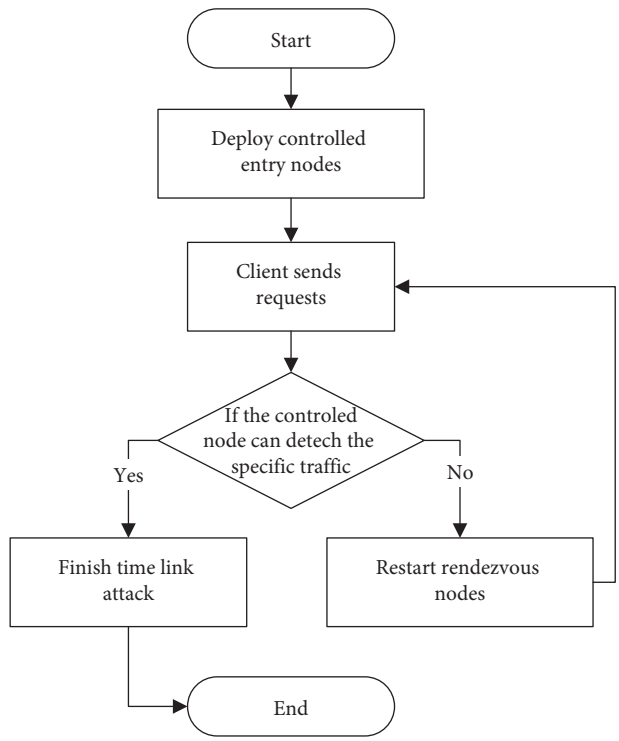

Figure 6: Time-linked Tor node reset attack flowchart.

(2) Select to turn off the Tor service at the RP, and the HS will disconnect the HR link at this time.

(3) Through our client to access a specific anonymous server again to send a continuous request within a certain period of time, re-establish the CR and HR long link. However, the anonymous server at this time has already established an HR of three hops different from the previous one, and thus, the selected entry node has also changed.

Since we have a certain number of entry nodes, we repeat step 2 and 3 until a specific anonymous server connects to the entry node we control.

Finally, according to the characteristics of packet time association and continuity, the IP of the anonymous server is determined so as to achieve the effect of the anonymous server to anonymity. Figure 6 shows the overall flowchart.

```
10:29:57.336942 IP centos44-7.commplex-main > 10.0.30.101.59696: Flags [.], ack 592
742, win 2716, options [nop,nop,TS val 2157455557 ecr 2157366435], length 0
10:29:57.372990 IP centos44-7.commplex-main > 10.0.30.101.59696: Flags [P.], seq 16
0641:161184, ack 592742, win 2716, options [nop,nop,TS val 2157455593 ecr 215736643
5], length 543
10:29:57.416251 IP 10.0.30.101.59696 > centos44-7.commplex-main: Flags [.], ack 161
184, win 1407, options [nop,nop,TS val 2157366519 ecr 2157455593], length 0
10:29:57.416267 IP centos44-7.commplex-main > 10.0.30.101.59696: Flags [P.], seq 16
1184:161727, ack 592742, win 2716, options [nop,nop,TS val 2157455636 ecr 215736651
9], length 543
10:29:57.420238 IP 10.0.30.101.59696 > centos44-7.commplex-main: Flags [P.], seq 59
2742:593285, ack 161727, win 1407, options [nop,nop,TS val 2157366522 ecr 215745563
6], length 543
10:29:57.448087 IP centos44-7.commplex-main > 10.0.30.101.59696: Flags [P.], seq 16
1727:162270, ack 593285, win 2716, options [nop,nop,TS val 2157455668 ecr 215736652
2], length 543
```

Figure 7: Time-linked Tor node reset attack result.

Table 3: Extracted features.

Feature name	Feature description
pkt_length	The overall packet size
fwd_pkt_length	The size and number of packets sent
bwd_pkt_length	The size and number of packets received
fwd_pkt_num_p	The proportion of packets sent as a percentage of the overall packet
bwd_pkt_num_p	The proportion of the accepted packet stake in the overall packet
nc_length_mv	The mean variance of the length of all forward packets before the flow changes direction
ncd_num_mv	The mean variance of the number of all forward packets before the flow changes direction
pkt_length_per_sec	The length of the packet per second
fwd_first_30_pkt_length	The length of the first 30 packets sent
bwd_first_30_pkt_num	The length of the first 30 packets received
Duration	The total transfer time

4.2.2. Reproduction Process. First, we modify the client and hidden server configuration files to enable the client Tor2-web mode. Also, specify the rendezvous node. After the client and hidden server configuration are completed, continuous requests are sent to the hidden server through the client for a certain period of time. Then use tcpdump crawl traffic data at all controllable portal nodes as well as at the client. Next, reset the Tor service at the rendezvous node, forcing the anonymous server to send a DESTROY instruction to destroy the link.

Repeat the reset service, and the client sends a continuous request for the steps so that the hidden server continuously destroys, re-selects the entry node, and re-establishes the link. Observing the traffic at the controllable node, in order to make the observation effect better, it is necessary to filter out the traffic of other interference items such as the directory server synchronization descriptor. The attack is stopped until the hidden server selects the controllable entry node as a node. Eventually, we can find the real IP (10.0.30.101) of the hidden server and complete the attack. Figure 7 shows the successful attack result.

4.3. Website Fingerprint Attacks on Tor Networks

4.3.1. Attack Fundamentals. Anonymous communication hides the address of the source and destination. The layered encryption of the traffic allows the attacker to detect the content information of the traffic. However, data traffic still has other dimensions. These features can be used to attack the anonymous system.

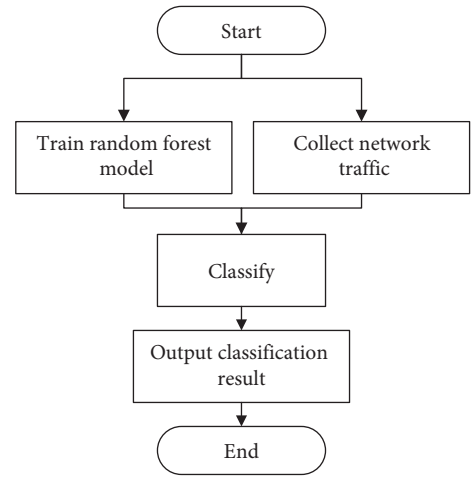

Figure 8: Website fingerprint attacks flow.

In order to achieve the purpose of the attack, the attacker first needs to obtain the fingerprint information of N websites, collect the traffic when visiting N websites, set up feature vectors, and train the classifier by using machine learning algorithms. When the target client accesses the website, the client can obtain the client traffic, and the import classifier determines whether the target client has accessed one or several of the N websites according to the classification result. Therefore, the prerequisite for a successful website fingerprint attack is to be able to collect traffic from the client. The attacker is required to be an ISP-level adversary.

Table 4: Classification test results.

Website	Number of client visits	The correct number of classifications	Accuracy
QQ	32	26	0.813
Tmall	26	22	0.85
Taobao	20	187	0.9
Sohu	8	67	0.75
JD	27	26	0.96
Weibo	6	57	0.83
Sina	33	29	0.88
360	10	87	0.8
Csdn	26	227	0.85
Bilibili	12	12	1

4.3.2. Reproduction Process. In this experiment, the Alexa China Top100 website was selected for fingerprint collection.

First, the local client opens Tor, connects to the Tor network, and accesses the above website. In the entry node, tcpdump is simulated by the ISP-level attacking attacker to grab the traffic sent and received by the client. The experiment will access each website 100 times with a random interval in between. The data set is the time delay and size of traffic data packets visiting the corresponding website. The extracted feature is a combination of these delays and sizes, such as the delay of the first 300 data, the size of the first 300 packets, and so on.

The selection of fingerprint features in this experiment is crucial for the entire attack because the selection characteristics largely determine the correct rate of classifier classification. Based on previous researchers' fingerprint attack experience on anonymous systems, this topic selects the extracted features shown in Table 3.

With the random forest algorithm using the Scikit-learn library in python, the data set is handed over to our classifier. The training is done by means of a tenfold cross-validation method. The ten-fold cross-validation method is used in the training, that is, the data set is divided into ten equal parts, one of which is used as the test set, and the rest are used for training and evaluation, so as to make the classification effect of our classifier better.

We have already built the classifier in the previous work, and then we will introduce the overall attack process. The flow chart of the attack process is shown in Figure 8. Randomly visit the website at the client to simulate normal user behavior and listen to traffic at the ingress route to simulate ISP-level adversaries. The monitored user traffic is processed by our script to extract features. The feature data is then passed to the classifier classification to obtain an output.

To test the accuracy of our classification, assume that the adversary is only interested in the Alexa Top10 website. Control the client to randomly access the website in Alexa Top10 for a total of 200 times, and the monitored traffic data is repeated. Finally, the statistical results in Table 4 are obtained.

The success rate of using fingerprint attacks on different websites to destroy client anonymity is basically above 80%, indicating that this attack is established in the Tor network.

Our design can meet the needs of different applications, and the experimental platform has good scalability. Users can quickly understand and familiarize themselves with the method of use, and the experimental platform has good ease of use. Compared with the existing research platform, our design has good generality.

5. Conclusion

This paper designs a new anonymous network research platform, which reduces the research threshold for anonymous networks and is suitable for comprehensive and efficient simulation of real anonymous network research. The research platform uses virtualization to quickly build the basic nodes in the anonymous network, avoiding deploying a large number of group hardware facilities in the network. The platform uses SDN, which can be defined and controlled by software programming, simplifying the network. The steps of network deployment meet the needs of the anonymous network for complex network environments. The platform can also deploy and run anonymous Tor source codes and other anonymous network source codes to ensure the network simulation operation's correctness and ensure that the virtual nodes can be linearly expanded. Finally, the paper highlights the platform's feasibility and superiority by processing two typical attacks on the platform. In the future, we will focus on technical research of anonymous networks in the platform, seeking to identify possible anonymous network vulnerabilities and implementing improvements.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 571–588, 2002.

[2] J. Xiong, M. Zhao, M. Z. A. Bhuiyan, L. Chen, and Y. Tian, "An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowdsensing of IoT," *IEEE*

Transactions on Industrial Informatics, vol. 17, no. 2, pp. 922–933, 2021.

[3] R. W. Gehl, *Weaving the Dark Web: Legitimacy on Freenet, Tor, and I2P*, MIT Press, Cambridge, MA, USA, 2018.

[4] P. Iacoban, *Measuring Accessibility of Popular Websites while Using the I2p Anonymity Network*, Delft University of Technology, Delft, Netherlands, 2021.

[5] A. Biryukov, I. Pustogarov, and R.-P. Weinmann, "Trawling for tor hidden services: detection, measurement, deanonymization," in *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, pp. 80–94, IEEE, Berkeley, CA, USA, May, 2013.

[6] D. Kavallieros, D. Myttas, E. Kermitsis, E. Lissaris, G. Giataganas, and E. Darra, "Understanding the dark web," in *Dark Web Investigation*, pp. 3–26, Springer, New york, NY, USA, 2021.

[7] K. Finklea, "Dark Web, Special Report for Congressional Research Service," R44101, 2015.

[8] J. Weber and E. W. Kruisbergen, "Criminal markets: the dark web, money laundering and counterstrategies - an overview of the 10th Research Conference on Organized Crime," *Trends in Organized Crime*, vol. 22, no. 3, pp. 346–356, 2019.

[9] J. Xiong, R. Bi, Y. Tian, X. Liu, and J. Ma, "Security and privacy in mobile crowdsensing: models, progresses, and trends," *Chinese Journal of Computers*, vol. 44, no. 09, pp. 1949–1966, 2021.

[10] W. Yu, X. Fu, S. Graham, X. Dong, and W. Zhao, "Dsss-based flow marking technique for invisible traceback," in *Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP'07)*, pp. 18–32, IEEE, Berkeley, CA, USA, May 2007.

[11] X. Wang and S. Douglas, "Robust correlation of encrypted attack traffic through stepping stones by manipulation of interpacket delays," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 20–29, Washington D.C. USA, October, 2003.

[12] X. Wang, S. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," in *Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP'07)*, pp. 116–130, IEEE, Berkeley, CA, USA, May, 2007.

[13] D. Agrawal, D. Kesdogan, and S. Penz, "Probabilistic treatment of mixes to hamper traffic analysis," in *Proceedings of the 2003 Symposium On Security And Privacy*, pp. 16–27, IEEE, Berkeley, CA, USA, May, 2003.

[14] M. Liberatore and B. N. Levine, "Inferring the source of encrypted http connections," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 255–263, Alexandria Virginia USA, October, 2006.

[15] T. G. Abbott, K. J. Lai, M. R. Lieberman, and E. C. Price, "Browser-based attacks on tor," in *International Workshop on Privacy Enhancing Technologies*, pp. 184–199, Springer, New york, NY, USA, 2007.

[16] F. W. Baumann, U. Odefey, S. Hudert, M. Falkenthal, and U. Breitenbücher, "Utilising the tor network for iot addressing and connectivity," in *Proceedings of the 8th International Conference on Cloud Computing and Services Science*, pp. 27–34, Madeira, Portugal, March, 2018.

[17] N. Phong Hoang and D. Pishva, "A tor-based anonymous communication approach to secure smart home appliances," in *Proceedings of the 2015 17th International Conference on Advanced Communication Technology (ICACT)*, pp. 517–525, IEEE, PyeongChang, Korea (South), July, 2015.

[18] L. Yang, C. Seasholtz, B. Luo, and F. Li, "Hide your hackable smart home from remote attacks: the multipath onion iot gateways," in *European Symposium on Research in Computer Security*, pp. 575–594, Springer, New york, NY, USA, 2018.

[19] Q. Abu Al-Haija, M. Krichen, and W. Abu Elhaija, "Machine-learning-based darknet traffic detection system for iot applications," *Electronics*, vol. 11, no. 4, p. 556, 2022.

[20] S. Patil and L. A. Raj, "Classification of traffic over collaborative iot and cloud platforms using deep learning recurrent lstm," *Computer Science*, vol. 22, no. 3, 2021.

[21] J. Hiller, P. Jan, M. Dahlmanns, H. Martin, A. Panchenko, and K. Wehrle, "Tailoring onion routing to the internet of things: security and privacy in untrusted environments," in *Proceedings of the 2019 IEEE 27th International Conference on Network Protocols (ICNP)*, pp. 1–12, IEEE, Chicago, IL, USA, October, 2019.

[22] R. Amin, M. Reisslein, and N. Shah, "Hybrid sdn networks: a survey of existing approaches," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3259–3306, 2018.

[23] T. Wong, H. Cui, Y. Shen, W. Lin, and T. Yu, "Anonymous network communication based on sdn," in *Proceedings of the 2018 4th International Conference on Universal Village (UV)*, pp. 1–5, Boston, MA, USA, October, 2018.

[24] H. Aldabbas and R. Amin, "A novel mechanism to handle address spoofing attacks in sdn based iot," *Cluster Computing*, vol. 24, no. 4, pp. 3011–3026, 2021.

[25] P. Wang, H. Liu, B. Wang, K. Dong, L. Wang, and S. Xu, "Simulation of dark network scene based on the big data environment," in *Proceedings of the International Conference on Information Technology and Electrical Engineering 2018*, October, 2018.

[26] Q. Wang and W. Cao, "A tor anonymity attack experiment platform driven by raspberry pi," in *Proceedings of the 2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan)*, pp. 569–574, Jinan, China, October, 2020.

[27] Z. H. O. N. G. Ying-Shou, L. I. Nan-Fang, Y. A. N. G. Li-Li, and Xu Wang, "Locating the Source of Message Diffusion in the Anonymous Network," *DEStech Transactions on Computer Science and Engineering*, 2017.

[28] J. Mark, M. Alberto, J. Gian, and V. Spyros, "The Peersim Simulator," 2003, http://peersim.sf.net.

[29] J. Tracey, "Building a Better Tor Experimentation Platform from the Magic of Dynamic elfs,", Master's Thesis, University of Waterloo , 2017.

[30] R. Jansen and N. Hooper, "Shadow: Running Tor in a Box for Accurate and Efficient Experimentation," NDSS, 2011.

[31] D. Komosny, S. Mrdovic, P. Ilko, M. Grejtak, and O. Pospichal, "Testing internet applications and services using planetlab," *Computer Standards & Interfaces*, vol. 53, pp. 33–38, 2017.

[32] B. Chun, D. Culler, T. Roscoe et al., "PlanetLab," *ACM SIGCOMM - Computer Communication Review*, vol. 33, no. 3, pp. 3–12, 2003.

[33] K. S. Bauer, M. Sherr, and D. Grunwald, *Experimentor: A Testbed for Safe and Realistic Tor Experimentation*Georgetown University, Washington D.C. USA, 2011.

[34] K. Venkatesh Vishwanath, D. Gupta, V. Amin, and K. Yocum, "Modelnet: towards a datacenter emulation environment," in *Proceedings of the 2009 IEEE Ninth International Conference on Peer-To-Peer Computing*, pp. 81-82, IEEE, Seattle, WA, USA, September, 2009.

[35] A. Johnson, C. Wacek, R. Jansen, M. Sherr, and S. Paul, "Users get routed: traffic correlation on tor by realistic adversaries," in *Proceedings of the 2013 ACM SIGSAC Conference on*

Computer & communications security, pp. 337–348, Berlin, Germany, May, 2013.

[36] A. Panchenko and J. Renner, "Path selection metrics for performance-improved onion routing," in *Proceedings of the 2009 Ninth Annual International Symposium on Applications and the Internet*, pp. 114–120, IEEE, Bellevue, WA, USA, July, 2009.

[37] M. AlSabah and I. Goldberg, "Performance and security improvements for tor," *ACM Computing Surveys*, vol. 49, no. 2, pp. 1–36, 2016.