CS445 Term Project Technical Report

Project Title: Simulating a DDoS Attack and Network Performance Analysis Using ns-3

Course: CS445 – Computer Networking
Instructor: Dr. Salehin

Lamia Alkhaldi – Starr Nakamitsu

Abstract

This project demonstrates the use of the ns-3 network simulator to model a Distributed Denial of Service (DDoS) attack on a server. The simulation aims to observe how malicious traffic affects packet loss. Additionally, we discuss DDoS attacks in general and their impacts, and the code implementation for the ns-3 simulation.

Introduction

Distributed Denial of Service (DDoS) attacks pose a major threat to the reliability and availability of today's networks. In such attacks, multiple compromised systems generate large volumes of traffic toward a victim server, exhausting its resources and preventing legitimate clients from receiving service. This project simulates an ICMP Ping Flood attack using the ns-3 network simulator to analyze how high-volume malicious traffic affects a target server and the overall network performance.

Related technical content

The goal of a DDoS attack is to prevent a legitimate client from receiving services and effectively making the server(s), service(s), or networks unavailable for those users. This is done by "overloading or flooding a target machine". However, DDoS attacks are also split into three different categories: volume based attacks, protocol based attacks, and application layer attacks (Harshita, 2017). ICMP ping flood attacks fall under volume based attacks, which is what we will be simulating in ns-3.

An ICMP Ping Flood is a simple but effective DDoS technique where attackers transmit a massive number of ICMP Echo Request ("ping") packets to a target. The server must respond to each request with an Echo Reply, consuming CPU cycles, memory, queue space, and bandwidth. Under sustained load, the victim becomes slow or unresponsive.

ns-3 is a discrete-event network simulator widely used in academia due to its accurate modeling of network protocols, devices, and traffic patterns. It allows researchers to study complex attacks like ICMP flooding without exposing real systems to risk. ns-3 provides tools such as FlowMonitor and tracing mechanisms that make performance measurement precise and repeatable.

Literature Review

Several researchers have explored DDoS behavior using simulation-based approaches:
Etxenike et al. (2024) examined multiple cyber-attack scenarios, including DoS and MitM, showing how high-volume traffic saturates networks and disrupts service availability.

Blazić and Basićević (2024) proposed a DDoS mitigation technique using Turing-test-based filtering to distinguish legitimate from malicious traffic.

Lamptey (2025) extended ns-3 to simulate IoT malware generating distributed attack traffic, demonstrating how large botnets can produce devastating floods.

These studies reinforce the value of simulation as a safe, controlled way to understand attack patterns and evaluate defenses.

Simulation Methodology

Implementation

The code first creates nodes, which are meant to symbolize our client, router, server, and attackers. Then, a point-to-point creation is made between the client and router, router and server, and attackers to server. This process also includes setting up attributes for each node/device (e.g., data rate and delay). We chose realistic values for the point-to-point connection between the client and router (10Mbps for data rate and 5ms for delay) and router and server (10Mbps for data rate and 1ms for delay). However, the data rate for attacker nodes is significantly higher due to reports of DDoS attacks peaking between 1-10 Gbps (Genie Networks, 2021). IP addresses were then assigned to all the nodes before populating each of the node's routing tables through the command **Ipv4GlobalRoutingHelper::PopulateRoutingTables();** Lastly was setting up all the nodes so that they could perform their perspective tasks. A UDP echo server and UDP echo client were set up. Then, the client was set up to send ICMP ping requests to the server. The attacker nodes were also set up to send ICMP ping requests to the server. Lastly, we simply laid out the nodes so that they could appear visually and ran the simulation.

The **NetAnim** tool is used to create the simulation. By including the "ns3/netanim-module.h" in the code header, we can use the NetAnim GUI by exporting an XML file and uploading it to the offline animator tool. The code exports the XML file as "**icmp-ddos.xml**".

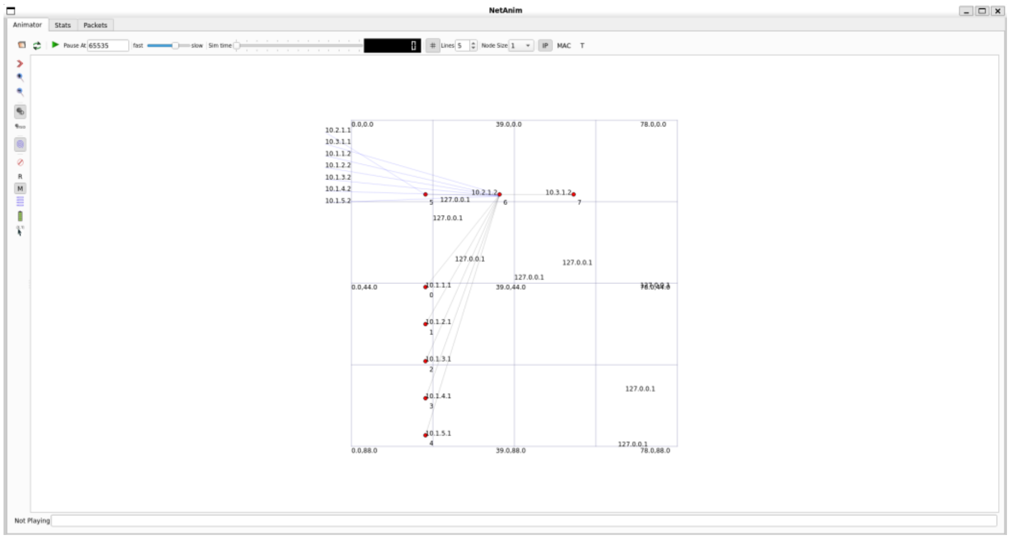
Network Topology

The simulated network contains 8 nodes:

- **5** Attacker nodes (Node 0-4)
- **1** Legitimate client node (Node 5)
- **1** Router node (Node 6)
- **1** Server node (Node 7)

Figure 1

Network topology



Note: Each node is labeled with their corresponding IP address and number (0-7)

Device and Traffic Parameters

Link Characteristics:

- Attacker => Router parameters
  - Data Rate: 1000 Mbps (high)
  - Delay: 2 ms
- Client => Router parameters
  - Data Rate: 10 Mbps (within normal range for one user; 5 Mbps – 40 Mbps)
  - Delay: 5 ms
- Router => Server parameters
  - Data Rate: 10 Mbps (normal range)
  - Delay: 1 ms

Attack Traffic (ICMP Ping Flood):

- Data Rate: 1000 ping requests/s
- Ping/payload size: 1024 bytes
- Duration: ~9s

Legitimate Traffic:

- Data rate: 2 ping requests/s
- Packet Size: 64 bytes
- Duration: 9s

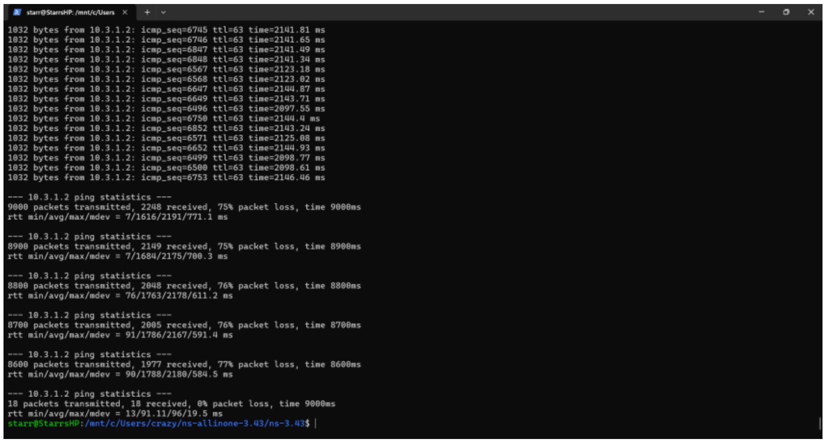
Results and Discussion

The server starts first 0.25 seconds into the simulation, while the client starts 1 second into the simulation. The first attacker node starts at 1.1 seconds into the simulation, and each attacker node follows 0.1 seconds after the other. All nodes stop running at 10 seconds, but the simulation runs a bit longer for 15 seconds. Thus, Figure 2 shows the ping statistics from the server. As assumed, there is substantial packet loss due to the attacker nodes constantly sending large ICMP requests. The first set of ping statistics is when the last attacker node (Node 4) stops sending ping requests, while the fourth set of ping statistics is when the first attacker node (Node 0) stops sending ping requests. The last set of ping statistics is while the simulation is still running, but the server is idle, and the client and attacker nodes are no longer sending ping requests.

We also want to admit that our simulation is heavily simplified. All the nodes are connected through point-to-point connections, and does not capture the complexity of real networks. Additionally, DDoS attacks are unlikely to happen through point-to-point connections. Instead, DDoS attacks are much more distributed using many bots, networks, and multi-hops to attack their target (Kopp, D., Dietzel, C., & Hohlfeld, O., 2021). We chose ICMP flood simply due to familiarity after working closely with it during previous labs. However, the simulation and results clearly show the effects of ICMP flood attacks, which was the goal of this project.

Figure 2

Terminal output for simulation

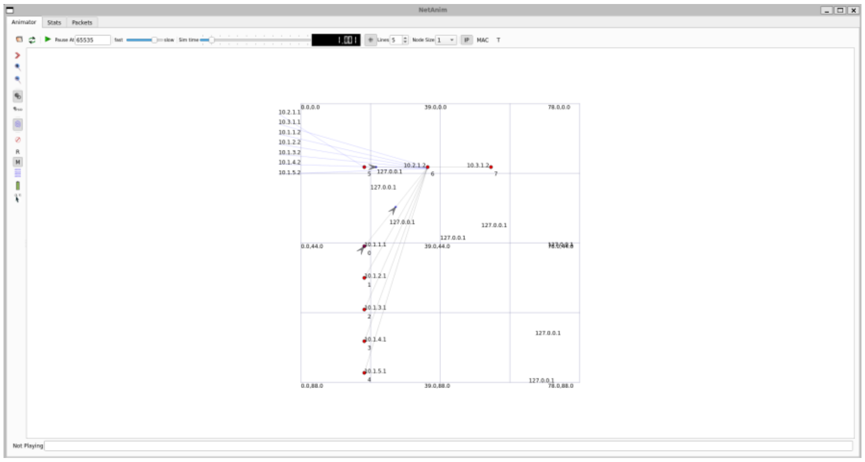


```
1032 bytes from 10.3.1.2: icmp_seq=6745 ttl=63 time=2141.81 ms
1032 bytes from 10.3.1.2: icmp_seq=6746 ttl=63 time=2141.65 ms
1032 bytes from 10.3.1.2: icmp_seq=6847 ttl=63 time=2141.49 ms
1032 bytes from 10.3.1.2: icmp_seq=6848 ttl=63 time=2141.34 ms
1032 bytes from 10.3.1.2: icmp_seq=6567 ttl=63 time=2123.18 ms
1032 bytes from 10.3.1.2: icmp_seq=6568 ttl=63 time=2123.02 ms
1032 bytes from 10.3.1.2: icmp_seq=6647 ttl=63 time=2144.87 ms
1032 bytes from 10.3.1.2: icmp_seq=6649 ttl=63 time=2143.71 ms
1032 bytes from 10.3.1.2: icmp_seq=6496 ttl=63 time=2097.55 ms
1032 bytes from 10.3.1.2: icmp_seq=6750 ttl=63 time=2144.4 ms
1032 bytes from 10.3.1.2: icmp_seq=6852 ttl=63 time=2143.24 ms
1032 bytes from 10.3.1.2: icmp_seq=6571 ttl=63 time=2125.08 ms
1032 bytes from 10.3.1.2: icmp_seq=6652 ttl=63 time=2144.93 ms
1032 bytes from 10.3.1.2: icmp_seq=6499 ttl=63 time=2098.77 ms
1032 bytes from 10.3.1.2: icmp_seq=6500 ttl=63 time=2098.61 ms
1032 bytes from 10.3.1.2: icmp_seq=6753 ttl=63 time=2146.46 ms

--- 10.3.1.2 ping statistics ---
9000 packets transmitted, 2248 received, 75% packet loss, time 9000ms
rtt min/avg/max/mdev = 7/1616/2191/771.1 ms

--- 10.3.1.2 ping statistics ---
8900 packets transmitted, 2149 received, 75% packet loss, time 8900ms
rtt min/avg/max/mdev = 7/1684/2175/700.3 ms

--- 10.3.1.2 ping statistics ---
8800 packets transmitted, 2048 received, 76% packet loss, time 8800ms
rtt min/avg/max/mdev = 76/1763/2178/611.2 ms

--- 10.3.1.2 ping statistics ---
8700 packets transmitted, 2005 received, 76% packet loss, time 8700ms
rtt min/avg/max/mdev = 91/1786/2167/591.4 ms

--- 10.3.1.2 ping statistics ---
8600 packets transmitted, 1977 received, 77% packet loss, time 8600ms
rtt min/avg/max/mdev = 90/1788/2180/584.5 ms

--- 10.3.1.2 ping statistics ---
18 packets transmitted, 18 received, 0% packet loss, time 9000ms
rtt min/avg/max/mdev = 13/91.11/96/19.5 ms
starr@StarrsHP:/mnt/c/Users/crazy/ns-allinone-3.43/ns-3.43$ |
```

Note: Server packet loss and round-trip-time reports printed to the terminal. Shown above the ping statistics are replies sent from the server.

Figure 3

Simulation in NetAnim at ~1s


Pause At 65535

Sim time 1.001

Lines 5

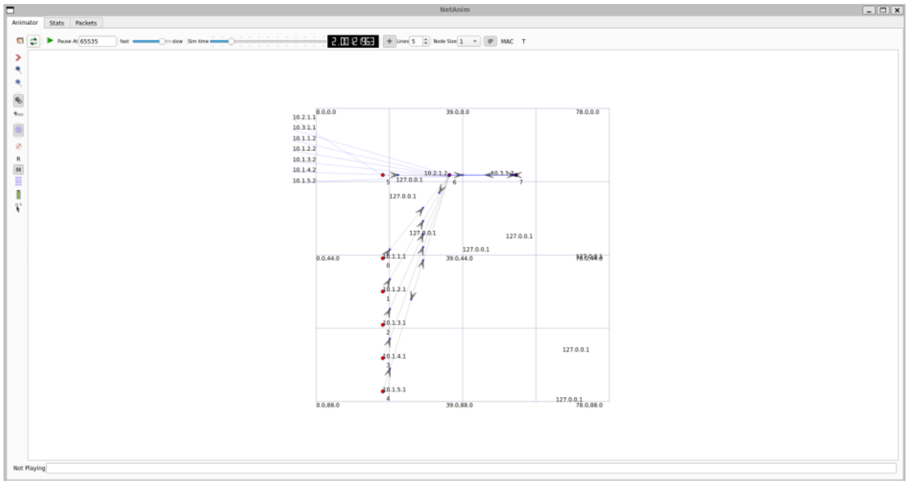
Node Size 1

Not Playing

Note: The client and first attacker node (node 0) begin sending ICMP requests to the server (depicted by arrows on each path)

Figure 4

Simulation in NetAnim at ~2s


Pause At 65535

Sim time 2.002963

Lines 5

Node Size 1

Not Playing

Note: All attacker nodes are sending ICMP requests while server is trying to respond to all the incoming requests (depicted by arrows on each path)

Conclusion

This report outlines the design and methodology for simulating an ICMP Ping Flood attack using ns-3. Based on previous research on DDoS attacks and their impact on their target, the simulation we created in ns-3 aligns with the fact that DDoS attacks can greatly impact communication between the target and legitimate clients. Using simulation tools such as ns-3 is a great way to not only visualize the effects of DDoS attacks, but to learn about all the parts and how they function in these scenarios.

References

Blazić and Basićević, 'Combating DDoS Attacks with Turing Tests,' 2024

Etxenike et al., 'Network Simulation with Complex Cyber-attack Scenarios,' 2024.

Genie Networks (2021). DDoS Threat Analysis Report [10]. *https://www.genie-networks.com/wp-content/uploads/2022/04/DDoS_Report_2021_En.pdf*

Harshita, Harshita. "Detection and Prevention of ICMP Flood DDOS Attack." *International Journal of New Technology and Research*, vol. 3, no. 3, Mar. 2017.

Kopp, D., Dietzel, C., & Hohlfeld, O. (2021, March). DDoS never dies? An IXP perspective on DDoS amplification attacks. In *International Conference on Passive and Active Network Measurement* (pp. 284-301). Cham: Springer International Publishing.

Lamptey, 'Extending Network Simulator ns-3 for Analyzing IoT Malware,' 2025.

Ns-3 (2025). Documentation. *https://www.nsnam.org/releases/ns-3-46/documentation/*

Ns-3 (2023). NetAnim. *https://www.nsnam.org/wiki/NetAnim*

Ns-3 Simulations (2025). How To Implement ICMP Attack In Ns3. *https://ns3simulation.com/how-to-implement-icmp-attack-in-ns3/*

Ns-3 Simulations (2025). How To Implement DDoS Attack In Ns3. *https://ns3simulation.com/how-to-implement-ddos-attack-in-ns3/*

Upadhyay Saket (2020). DDoS Simulation in NS-3 [C++]. *Medium, InfoSec Write-ups* *https://infosecwriteups.com/ddos-simulation-in-ns-3-c-12f031a7b38c*