Assessing the Resilience of Modern Satellite Internet Services Against GPS Spoofing Attacks

Martin Hägglund, Gurjot Singh Gaba, Andrei Gurtov
Department of Computer and Information Sciences (IDA)
Linköping University
Linköping, Östergötland SE-581 83, Sweden
marha057@student.liu.se, gurjot.singh@liu.se, andrei.gurtov@liu.se

***Abstract*—The rise of satellite-based Wi-Fi through large-scale satellite constellations offers unprecedented global connectivity, requiring only a small receiver for internet access from virtually anywhere. However, this technology introduces new security vulnerabilities, with one such vulnerability being a reliance on GPS signals, which are susceptible to spoofing attacks. While previous research has explored different GPS spoofing techniques, the impact of such attacks on satellite-based internet systems remains largely unexplored. This study examined the effects of GPS spoofing on a satellite-based internet receiver through a controlled experiment using a HackRF One to conduct an impersonation attack. The receiver was spoofed to various locations to assess the effect on network performance. A total of 63 tests were successfully performed. Of these, 54 tests used the speed test of the app associated to the satellite receiver to measure download speed, latency, and packet loss percentage. The remaining 9 tests used Wireshark to monitor a 1 GB file download, recording average throughput, average packet size, and total download time. Experiment results indicate that network performance remained largely unaffected. For example average download speed changing at most $\pm$10 Mbit/s for all spoofing distances, even when a handheld GPS device was successfully spoofed. Two outliers were observed during testing, both occurring in the app speed tests. These findings suggest that the satellite signal receivers either employ additional built-in protections against spoofing or mitigate such attacks by leveraging the satellite constellation for location verification. While this study provides initial insights, further research utilizing more sophisticated attack techniques and hardware modifications is needed to comprehensively evaluate the resilience of modern satellite internet receivers against GPS spoofing attacks.**

***Index Terms*—GPS, Spoofing, HackRF One, Satellite communications, Wi-Fi.**

I. Introduction

The internet is becoming an increasingly integral part of daily life, facilitating essential activities in both private life and the workplace [1]. While reliable and secure Wi-Fi is easily accessible in urban areas of developed countries, individuals in rural regions or developing nations often struggle with connectivity limitations [2]. Even in urban settings, finding a secure and private internet connection outside one's home or workplace can be challenging, as public Wi-Fi networks are inherently vulnerable to cyber threats [3].

To address these challenges, satellite-based Wi-Fi solutions have emerged as a promising alternative. These systems use compact *satellite receivers* to connect with large *satellite constellations*, allowing users to establish private Wi-Fi networks virtually anywhere. This technology offers a cost-effective and flexible solution for individuals seeking connectivity in areas where conventional broadband and internet infrastructure is impractical. Notably, satellite-based internet has been deployed in extreme environments such as Antarctica, and war-affected regions like Ukraine, enabling stable connections in areas that would otherwise be inaccessible [4].

However, the increasing reliance on *Satellite Communication Systems* (SCSs) introduces significant security concerns. Unlike traditional network infrastructure, SCSs face unique cybersecurity challenges due to the high cost of satellite deployment and the difficulty of patching vulnerabilities once satellites are in orbit. Historically, satellite receivers have featured only basic security mechanisms, making them attractive targets for cyberattacks [5]. One such threat is GPS spoofing, which exploits the satellite receivers reliance on GPS signals for geolocation and operational stability. These attacks allow the attacker to arbitrarily change the targets perceived location, disrupting satellite communications as the target transmits to an incorrect sky location due to the difference in real and perceived geographic location. As growing availability of low-cost *Software Defined Radios* (SDRs) has lowered the bar for malicious actors, further research into the effects of these attacks are needed. While previous research has explored the theoretical feasibility of GPS spoofing, its direct impact on satellite communication performance remains largely unexplored.

A. Research Questions

This study aims to investigate the impact of GPS spoofing on Starlink receivers through an experimental approach. Specifically, we address the following research questions:

1) How feasible is a GPS spoofing attack on a Starlink receiver using a HackRF One, and what conditions are necessary for success?
2) How does GPS spoofing to varying distances affect the performance of a Starlink satellite communication system?

B. Paper Organization

The remainder of this paper is structured as follows: Section II provides an overview of *Starlink*, SDRs, GPS, and related research. Section III details the hardware and software

used in the practical experiment, along with the experimental setup. Section IV presents and analyzes the findings from the experiment and discusses their potential implications. Finally, Section V concludes the study by summarizing the results in relation to the research questions and offering recommendations for future research.

II. Background and Related Work

This section provides the necessary background information and reviews related research relevant to the experiment conducted in this study.

A. Global Positioning System

A *Global Positioning System* (*GPS*) is a radio-navigation SCS that allows users to find their location on or near the Earth with the help of highly accurate navigation pulses. A receiver is able to compute its position by analyzing signals from at least 4 satellites [6]. GPS signals always contain at least the following information: Highly accurate orbital data about the satellite itself (*ephemeris* data), approximate orbital data for all other satellites in the constellation (*almanac data*), satellite clock readings and satellite health status. The civilian GPS signal pulses are transmitted at a frequency of 1575.42 MHz (L1 band) [7]. While GPS is commonly used to mean all *global navigation satellite systems* (*GNSS*), in this paper it refers to the American Navstar GPS.

B. Software Defined Radio

A software defined radio (SDR) is a radio transmission system where the signal waveform can be modified without changing the underlying hardware. The limits to the waveform changes are determined by the underlying hardware, but should include at least multi-band operation, multiple modulation schemes, and half- or full-duplex communication [8].

C. Starlink

Starlink is a *SpaceX* service offering satellite based Wi-Fi to users globally. The service targets both individuals, companies and governments. The end user is provided with a compact, mobile satellite dish which connects to the Starlink satellite constellation. The satellites then communicate with a base station, connecting the user to the wider internet. The user terminal features a phased-array antenna, allowing it to establish a stable connection with the satellites while stationary or in motion, such as when mounted on a vehicle. Additionally, the terminal relies on GPS signals to determine its precise location and optimize connectivity [4].

D. Related Work

Extensive research has been conducted on GPS spoofing over the years. In 2008, Humphreys et al. successfully developed a working GPS spoofer using specialized hardware [9]. In the same study, they predicted that advancements in SDR technology would significantly reduce the hardware requirements for conducting sophisticated GPS spoofing attacks. This concern was further explored by Humphreys and Psiaki in a 2016 study, where they outlined a range of theoretical GPS spoofing attack methods with varying levels of complexity, along with potential countermeasures. Their findings indicated that no commercial GPS receiver, at the time, had sufficient protection against state-of-the-art GPS spoofing techniques [5]. Supporting these claims, Gaspar et al. (2020) demonstrated a successful GPS spoofing attack on a commercial drone in an outdoor environment using a bladeRF SDR, a Raspberry Pi, and open-source GPS signal generation software. While the time required to achieve a successful spoof varied, the drone was often compromised within three minutes, even when it had an established legitimate GPS lock [10].

Tedeschi, Sciancalepore, and Di Pietro provide a comprehensive summary of satellite communication security. Among other contributions, their paper includes a table outlining possible security schemes for GPS spoofing mitigation, along with the implementation requirements for each approach. These schemes are categorized into four types, all of which require either additional hardware, network communication capabilities, or the sharing of physical layer information. The authors also emphasize the inherent vulnerability of satellite communication systems to spoofing, due to long-range transmission, reliance on insecure legacy protocols, and the increasing accessibility of modern SDRs [11].

Despite this, research gaps persist. Very little research has been done investigating how GPS spoofing can affect systems that rely on precise positioning and timing for successful communication such as SCSs. This study aims to address this gap by conducting a practical experiment to assess the potential consequences of GPS spoofing on Starlink's satellite-based internet service.

III. Experiment Planning and Realisation

To address the research gaps identified in related work, a practical experiment was designed and conducted. The first phase involved evaluating and selecting the necessary hardware and software for the experiment, followed by planning the experimental procedure. The study was constrained by limited resources, as only a single HackRF One was available for performing the GPS spoofing attack, and a Starlink satellite receiver was used as the target device. Additionally, time constraints restricted the number of test iterations and the extent of attack method development. Despite these limitations, the experiment was structured to provide meaningful insights into the impact of GPS spoofing on Starlink's network performance.

A. Hardware and Software

The first step of this experiment was to confirm that the Starlink receiver relies on GPS signals to determine its location. Although official documentation does not explicitly state this, previous observations found that GPS is used during the startup process and continuously monitors whether the receiver is stationary or in motion. The chip used for this is able to receive signals from all GNSSs in parallell [12].

TABLE I: gps-sdr-sim program flow

GPS-SDR-SIM algorithm

Input data: *Location* given in latitude, longitude, height format
Daily GPS satellite ephemeris file
Simulation run time
(optional) Simulation start time
Output: Transmission file of specified run time
Begin
while *current time* $\leq$ *Simulation run time* **do**
Calculate satellite positions based on *current time* and ephemeris file
for each satellite **do**
Calculate satellite distance from *location*
Determine satellite visibility from *location*
if visible **do**
Generate satellite message frames and checksums
Compute Doppler shift due to satellite movement
end
end
Convert message data into a transmission data
end
end

Next, the HackRF One was examined to assess its capability for performing GPS spoofing attacks. According to its technical specifications, the HackRF One can transmit and receive signals between 1 MHz and 6 GHz, which encompasses the civilian GPS L1 frequency band (1575.42 MHz), confirming its suitability for GPS spoofing [13]. However, since direct extraction of GPS data from the Starlink receiver was infeasible without hardware modifications, a handheld GPS device was used to validate the spoofing attack. The Colorado 300 handheld GPS was selected for this purpose due to it being a reliable GPS which is expected to match or exceed the GPS capabilities of the Starlink receiver. Another crucial component used in the experiment was a miniGPS reference clock, which served as an external clock source for the HackRF One [14]. This was necessary as the internal clock of the HackRF One lacked sufficient precision, preventing it from accurately simulating a GPS satellite signal for successful spoofing.

When selecting the software for use in the experiment, time and resource constraints meant an existing open-source solution was preferred over creating a custom built program. *GPS-SDR-SIM* was chosen, as it enables the generation of GPS signal files based on freely available GPS satellite data [15]. These transmission files were then used by the HackRF One to conduct impersonation attacks, attempting to spoof nearby GPS receivers. A simplified explanation of the software's functionality is provided in Table I. The daily ephemeris file required by the software is freely available on the CDDIS archive website, accessible to registered users [16]. To analyze the impact of GPS spoofing on Starlink network performance, two software tools were utilized: the Starlink app and Wireshark. The Starlink app was used in the initial tests to gather a large dataset of maximum connection capacity measurements. In the later tests, Wireshark was employed to monitor long-term network performance, capturing how spoofing influenced network behavior during sustained data transfers.

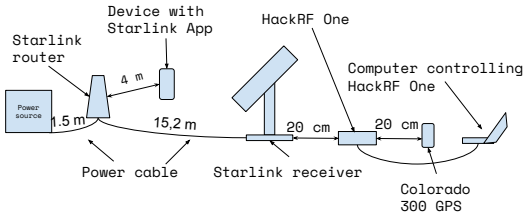
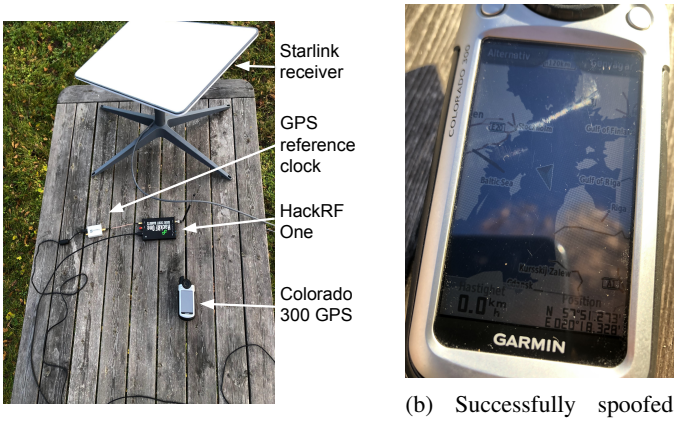


Fig. 1: Side view of experiment set-up



(a) On-site experiment setup

(b) Successfully spoofed GPS

Fig. 2: Experimental Setup and Evidence of Spoofing

B. Practical Experiment

With the hardware and software selected, the next step was to design and conduct the *practical experiment*. The primary objective was to evaluate the impact of GPS spoofing on *Starlink network performance*. To achieve this, five different spoofing distances were selected: 500 m, 1.000 m, 5.000 m, 10.000 m, and 50.000 m. These distances were chosen to assess the progressive degradation of network performance as the receiver was spoofed further from its actual location. To ensure comprehensive testing, each distance was evaluated in all four cardinal directions, resulting in a total of 21 test locations, five in each direction, along with the true geographic location serving as a baseline. Additionally, for the extended tests, the 50.000 m westward spoofed location was selected to determine the impact on sustained downloads. Since the Starlink subscription used in this study lacked coverage in the Baltic Sea, additional tests were conducted in which the receiver's GPS location was spoofed to simulate operation in international waters. The objective was to determine whether successful spoofing would completely disrupt connectivity.

To assess network performance, the initial tests focused on *download speed*, *latency*, and *error percentage*, while the extended tests measured average throughput, total download time, and average packet size. A planned metric, the percentage of dropped packets, was excluded after discovering that Wireshark classifies retransmitted packets as successful, consistently resulting in zero dropped packets. Whilst retransmits and other disruptions could be viewed in graph form, the

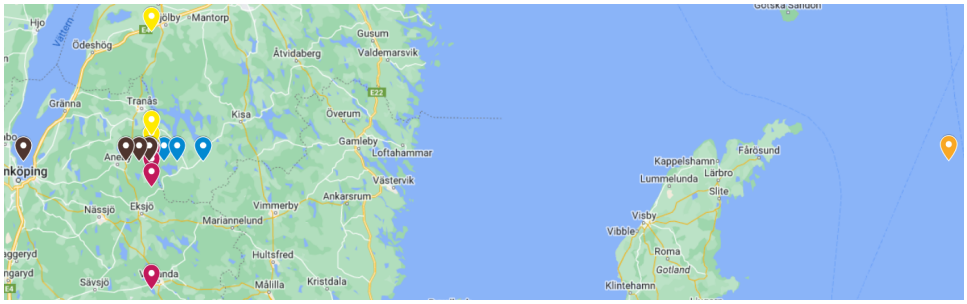


Fig. 3: All spoofing locations

inability to extract numeric information for a download made this unusable as an alternative.

Although GPS spoofing is most likely to affect the Starlink receiver during initialization, since it uses GPS to determine its initial setup location, the experiment was designed to simulate a realistic attack scenario. In practical settings, Starlink users rarely restart their receivers, making it impractical for attackers to exploit initialization vulnerabilities. Therefore, all spoofing attempts were conducted after the receiver had successfully connected to the Starlink constellation.

The experiment setup was constrained by physical limitations of the components involved. Since the Starlink router and receiver require a stable power source and an unobstructed view of the sky, the full length of the power cables was used to maximize the distance from the power source, allowing the router and receiver to be placed under an open sky. Testing revealed that for *successful spoofing*, the handheld GPS needed to be within 20 cm of the HackRF One, so this distance was maintained throughout the tests. Since there was no direct way to verify if the Starlink receiver had been spoofed, the experiment assumed that if the handheld GPS was successfully spoofed, the same spoofing signal had likely affected the Starlink receiver as well. The device used to measure download speed was placed 4 meters from the Starlink router, as maintaining a fixed distance was considered more critical than the exact placement itself. The same setup was used for the extended tests, except that a computer was used instead of the phone, with Wireshark used for data collection instead of the Starlink app.

The experiment was conducted over three separate days. The initial tests took place over two days in mid-November 2024, while the extended tests were performed on one day in mid-January 2025. The test setup followed the configuration shown in Fig. 1, with an onsite view provided in Fig. 2a. During the initial tests, the method involved activating the spoofer, waiting three minutes to verify spoofing success, and then conducting network tests. If spoofing failed, the spoofer was deactivated for at least two minutes before attempting again. A total of 54 successful tests were performed, including nine baseline tests. The extended tests in January followed a similar methodology, where once spoofing was confirmed via the handheld GPS, a Wireshark trace of a 1 GB file download was recorded. A total of 9 tests were conducted, with 2 baseline tests. Fig. 3 shows a map marking the locations where GPS spoofing was successfully performed.

IV. Results and Discussion

This section presents and analyzes the *experimental results*, discussing their significance and potential implications. To examine the relationship between spoofing distance and network performance, a *statistical analysis* was conducted using *Spearman's rank correlation coefficient* (Spearman correlation). This method was chosen as the data did not follow a normal distribution, and two of the measured data categories contained significant outliers, making parametric correlation tests unsuitable. Due to space limitations, the complete dataset are available in the GitHub repository [17].

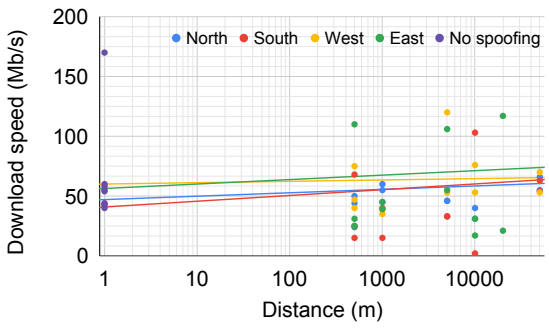
TABLE II: Spearman correlation results

Data category	P-value	Correlation
Download speed	0.5866	0.07565
Packet loss	0.4	-0.116863
Latency	0.8045	0.034486
Throughput	0.1159	0.561248
Download time	0.02038	-0.748331*
Packet size	0.123	0.552339

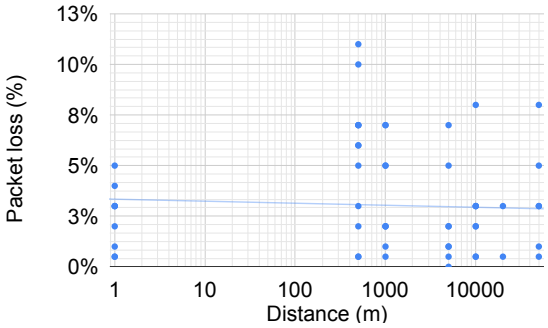
A. Initial Experiment Results

As discussed in Section III-B, the initial experiment yielded 54 successful test results over 2 testing days, including nine baseline tests. At least 2 tests were conducted for each spoofing distance and direction combination. A summary of the test results is provided in Fig. 4a, Fig. 4b and Fig. 4c. During the data analysis, it was discovered that the coordinates for the E 50.000 m spoofing location had been incorrectly identified, leading to tests being conducted at E 20.000 m instead. This discrepancy was accounted for in the final analysis. Additionally, 2 outliers were identified: 1 in download speed, where an anomalous value of 170 Mb/s was recorded, and 1 in latency, where a 289 ms delay was observed.

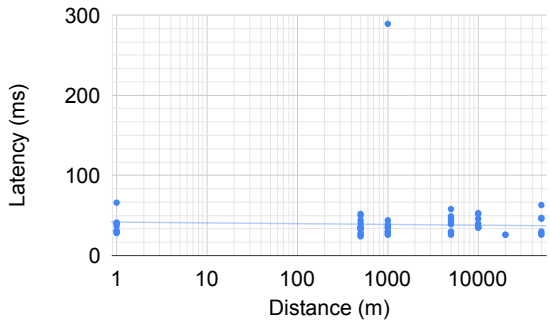
A Spearman correlation analysis was performed to examine potential relationships between spoofing distance and network performance, the results of which can be seen in Table II. However, no statistically significant correlations were found, despite the large sample size. Although the correlation values were unreliable, they suggest little to no correlation between spoofing distance and network performance. This trend was



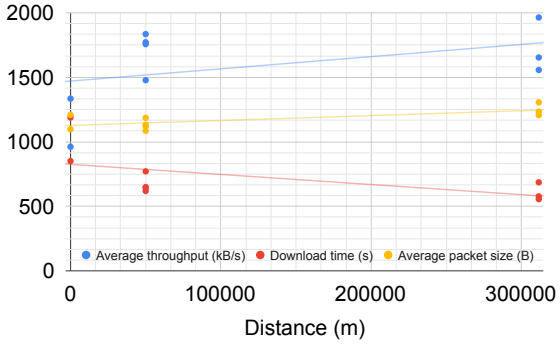
(a) Download speed, grouped by cardinal direction



(b) Interruption percentage



(c) Latency, no grouping



(d) Results of the extended tests

Fig. 4: Network performance metrics from initial and extended tests

further supported by visual inspection of the results, where trend lines unexpectedly indicated a slight improvement in performance as spoofing distance increased. The only notable effect observed across different spoofing distances was an increase in result variability during active spoofing, compared to the highly clustered baseline measurements. This pattern remained consistent across both testing days and throughout different times of the day when baseline tests were conducted with the exception of the 50.000 m tests which were highly clustered.

B. Extended Experiment Results

A total of 10 extended tests were conducted; however, due to an accidental file overwrite, 1 test result was lost, leaving 9 tests successfully recorded and analyzed. Of these, 2 were baseline tests, 4 were spoofed to the W 50.000 m location, and the remaining 3 were spoofed to the simulated location in the Baltic Sea. Each test lasted between 9 and 19 minutes, with a pause between consecutive tests. For data analysis, the average values for each test were computed using built-in Wireshark functions. These results are presented in Fig. 4d. Whilst download speeds were notably slower for these tests, this is likely due to browser limitations that were not a factor during the speed tests.

As shown in Table II, the Spearman correlations derived from the extended test data were considerably more reliable than those obtained from the initial tests. One of the correlation categories reached statistical significance, while the remaining two were close to the threshold. Interestingly, contrary to the hypothesis proposed in Section III-B, the Spearman correlations point to the performance of the network improving the greater the spoofing distance becomes. the correlation analysis indicates that network performance improves as the spoofing distance increases. This includes download time, which demonstrated a strong negative correlation with spoofing distance at -0.748331. Notably, this trend aligns with the findings from the initial tests, where the correlation values also suggested a slight improvement in network performance with increasing spoofing distance.

C. Discussion

The results from both the initial and extended tests strongly suggest that the Starlink receiver was *not successfully spoofed*, even when the handheld GPS indicated that the spoofing was successful. While the overall findings support this, the most compelling evidence comes from *the overseas spoofing attempts*. As noted in Section III-B, the Starlink service plan used during testing did not include oversea coverage. However, despite this limitation, the file downloads at the spoofed Baltic Sea location were completed successfully 3 times without any disruption. Moreover, these downloads were, on average, *the fastest* among all extended tests. If the spoofing had been fully effective and not mitigated, such downloads should not have been possible. This strongly suggests that the Starlink receiver was either *not spoofed at all* or *detected and counteracted* the spoofing attempt. An interesting and unexpected observation

is that network performance appeared to improve for several metrics when spoofing was active.

The most probable explanation is *random variation* in the recorded measurements. Despite conducting multiple baseline tests spread across different times of the day, it remains possible that baseline results were inherently more clustered than those obtained under spoofed conditions due to statistical randomness. This is supported by the *high p-values* obtained during the initial tests, which indicate a lack of statistical significance. Additionally, data from spoofing at 50 km from the true geographic location exhibited *clustering patterns similar to the baseline tests* in terms of both download speed and latency, further reinforcing this hypothesis. However, one factor contradicting this explanation is that download time during the extended tests achieved statistical significance, showing a strong negative correlation of -0.748. While this correlation suggests a meaningful trend, the small sample size raises concerns about result reliability, as a larger dataset may yield different conclusions even though each test result represents an average from an extended download session.

Another possible explanation is that spoofing was partially successful but detected, leading the Starlink receiver to implement mitigation measures, which in turn caused increased variance in the test results. The detection could occur in multiple ways. If the GPS chip uses signals from all GNSSs, having one report a different location could flag for spoofing. Alternatively, if the receiver detected a location change from the GPS feed without a need for a corresponding adjustment in the phased-array antenna, this could also indicate an ongoing spoofing attack. If this is flagged, it may have caused the receiver to switch from GPS-based positioning to determining its location through the Starlink constellation. In this scenario, the receiver would need to maintain continuous communication with one or more satellites to validate its location dynamically. As TCP communications involve an initial slow-start phase [18], it is possible that a receiver relying on the Starlink satellite network for location determination could have pre-emptively established connections with upcoming satellites, bypassing the slow-start phase. While this could improve network performance, it would also increase overall data exchange with the satellite constellation but leading to higher traffic loads on the system as a whole.

V. Conclusions and Future Work

Based on the findings presented in Section IV and Section III, the research questions posed in this study can now be addressed. Regarding the first research question, it is possible to conduct a successful GPS spoofing attack using a HackRF One, provided that it is equipped with a sufficiently precise external clock. This is necessary as the internal clock of the HackRF One lacks the required precision for GPS spoofing. As for the second research question, the results indicate that different levels of GPS spoofing had minimal impact on the network performance. The primary observable effect was an increase in the variability of download speeds and latency. However, it remains unclear whether this variation was caused by random network fluctuations, spoofing mitigation techniques, or other unknown factors.

Given the inconclusive nature of the results, there are several avenues for future research. Since the findings suggest that some form of change occurs when spoofing is active, further investigation is warranted. One potential improvement would be to modify the hardware to allow for direct measurement of GPS signals received by the receiver. This would provide a definitive confirmation of whether spoofing was successful. Additionally, employing more advanced attack techniques could increase the likelihood of a successful GPS spoofing attempt. The simplest improvement would be to repeat the experiment with an increased number of tests. Given that time constraints were a key limitation, conducting more tests would help reduce the influence of random variations in network performance, leading to more statistically robust conclusions.

References

[1] I. Okhrimenko, I. Sovik, S. Pyankova, and A. Lukyanova, "Digital transformation of the socio-economic system: Prospects for digitalization in society," *Revista Espacios*, vol. 40, no. 38, 2019.

[2] B. Whitacre and R. Gallardo, "State broadband policy: Impacts on availability," *Telecommunications Policy*, vol. 44, no. 9, 2020.

[3] I. McShane, M. Gregory, and C. Wilson, "Practicing safe public Wi-Fi: Assessing and managing data-security risks," *Available at SSRN 2895216*, 2016.

[4] SpaceX. "Starlink." (), [Online]. Available: https://www.starlink.com/ (visited on 01/31/2025).

[5] M. L. Psiaki and T. E. Humphreys, "GNSS spoofing and detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[6] T. S. Longsdon. "GPS." (2024), [Online]. Available: https://www.britannica.com/technology/GPS (visited on 02/03/2025).

[7] P. Kwan. "NAVSTAR GPS Space Segment/Navigation User Segment Interfaces." (2019), [Online]. Available: https://www.gps.gov/technical/icwg/IS-GPS-200K.pdf (visited on 02/04/2025).

[8] T. Ulversoy, "Software defined radio: Challenges and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 4, pp. 531–550, 2010.

[9] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, P. M. Kintner, *et al.*, "Assessing the spoofing threat: Development of a portable GPS civilian spoofer," in *Proceedings of the 21st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2008)*, 2008, pp. 2314–2325.

[10] J. Gaspar, R. Ferreira, P. Sebastião, and N. Souto, "Capture of uavs through gps spoofing using low-cost sdr platforms," *Wireless Personal Communications*, vol. 115, pp. 2729–2754, 2020.

[11] P. Tedeschi, S. Sciancalepore, and R. Di Pietro, "Satellite-based communications security: A survey of threats, solutions, and research challenges," *Computer Networks*, vol. 216, p. 109 246, 2022.

[12] O. Kutkov. "Connecting external GPS antenna to the starlink terminal." (), [Online]. Available: https://olegkutkov.me/2023/11/07/connecting-external-gps-antenna-to-the-starlink-terminal/ (visited on 01/31/2025).

[13] HackRF One team. "Hackrf one." (), [Online]. Available: https://greatscottgadgets.com/hackrf/one/ (visited on 01/31/2025).

[14] Leo Bodnar Electronics. "Mini precision gps reference clock." (), [Online]. Available: https://www.leobodnar.com/shop/index.php?main%5C_page=product%5C_info&products%5C_id=301 (visited on 02/03/2025).

[15] Github User Osqzsss. "Gps-sdr-sim." (), [Online]. Available: https://github.com/osqzss/gps-sdr-sim (visited on 02/03/2025).

[16] CDDIS. "Gnss data daily." Requires account to access. (), [Online]. Available: https://cddis.nasa.gov/archive/gnss/data/daily/ (visited on 02/03/2025).

[17] M. Hägglund. "Practical experiment result files." (), [Online]. Available: https://gitlab.liu.se/marha057/masters_thesis_result_storage_marha057/-/tree/main.

[18] L. Guo and J. Y. Lee, "Stateful-TCP—A new approach to accelerate TCP slow-start," *IEEE Access*, vol. 8, pp. 195 955–195 970, 2020.