Science DMZ Networks: How Different Are They Really?

Emily Mutter and Susmit Shannigrahi
Computer Science Department, Tennessee Tech University
Cookeville, TN, USA
{ebmutter42, sshannigrahi}@tntech.edu

***Abstract*—The Science Demilitarized Zone (Science DMZ) is a network environment optimized for scientific applications. The Science DMZ model provides a reference set of network design patterns, tuned hosts and protocol stacks dedicated to large data transfers and streamlined security postures that significantly improve data transfer performance, accelerating scientific collaboration and discovery. Over the past decade, many universities and organizations have adopted this model for their research computing. Despite becoming increasingly popular, there is a lack of quantitative studies comparing such a specialized network to conventional production networks regarding network characteristics and data transfer performance. But does a Science DMZ exhibit significantly different behavior than a general-purpose campus network? Does it improve application performance compared a to general-purpose network? Through a two-year-long quantitative network measurement study, we find that a Science DMZ exhibits lower latency, higher throughput, and lower jitter behaviors. We also see several non-intuitive results. For example, a DMZ may take a longer route to external destinations and experience higher latency than the campus network. While the DMZ model benefits researchers, the benefits are not automatic, careful network tuning based on specific use cases is required to realize the full potential of Science DMZs.**

I. Introduction

Science and engineering applications are generating data at an unprecedented rate, producing hundreds of terabytes to petabytes of data within a very short time [1]–[8]. Additionally, scientific collaborations are becoming increasingly global, which means the researchers must transfer these datasets over the wide area networks to various scientific facilities [9]–[13]. Such data transfers can occur between instruments [13], storage servers [11], [14], [15], computing systems [14], [16], and cloud computing platforms [17], [18]. General-purpose enterprise networks are often unsuitable for these types of data transfers since these networks prioritize general usability and security over performance. Scientific data transfers can face several challenges, such as bandwidth throttling, packet loss, slow throughput due to firewalls, intrusion detection systems, and other middleboxes, resulting in lower throughput, higher latency, and increased jitter and packet loss [19] [20] [21]. These challenges ultimately result in lower scientific productivity.

Organizations often tailor a portion of their network for scientific data transfers to address these challenges. Such a network is generally called a Science DMZ. Science DMZs prioritize data transfer performance through streamlined security postures, such as simple rule-based access control lists rather than stateful firewalls, and network tuning, such as large Ethernet frames and larger TCP windows.

Science DMZ networks are widely deployed at US academic campuses and other countries. By the latest count, more than 200 Science DMZs [22] are in the US alone. While they are widely deployed, there is a lack of comparative, quantitative studies on how Science DMZ networks differ from their general-purpose counterparts. To address this gap, we have observed a general-purpose production network alongside a Science DMZ at a university campus over the past two years. We have deployed multiple measurement instruments in both networks and external facilities. We have used a number of standard network measurement tools (iperf3, ping, traceroute) and developed our own comparison software to measure network parameters such as RTT, Throughput, Jitter, and Packet loss. Externally, we have looked into network traffic to and from large cloud platforms (Google Cloud) and the RIPE Atlas measurement platform.

These long-running measurements allowed us to understand the nuances in performance differences on both networks. We confirm that a Science DMZ generally provides a better environment for data-intensive research. However, such benefits are not automatic, and these networks may be susceptible to higher latency, packet loss, and longer paths. Therefore, careful network planning and optimization based on the requirements of specific use cases (e.g., bulk data vs. real-time) must be a part of such infrastructure.

II. Background

A. General-purpose Networks vs. Science DMZs

Campus networks are typically designed to serve large numbers of users and devices, support various applications (e.g., email, web browsing, and video), and provide security and quality of service [23]. Campus networks are also equipped with firewalls to maintain network security that often takes precedence over quality of service [23]. Because most general-purpose data flows are small (KBs-MBs) and have a short duration, moderate bandwidth, latency, and loss rates are usually sufficient for these flows. Most traditional applications on a campus network can adapt to the network's bandwidth and are not overly sensitive to packet loss or jitter.

On the contrary, scientific data is often at terabyte- and petabyte-scale [3], [12], [14], [18], [23]–[25]. When packet loss occurs during such transfers, TCP reduces throughput to levels where it can take days to complete a single data transfer [17], [20], [26]. Energy Sciences Network (ESNet) developed the Science DMZ architecture to address these issues and transfer scientific data faster. A Science DMZ is a portion of a network designed for high-performance scientific applications. It is often separated from the campus network either physically or logically [23]. Science DMZs also have a different security posture than enterprise networks. Instead of using multi-layer firewalls as in enterprise networks, Science DMZs use simple stateless Access Control Lists (ACLs) that allow line-rate packet processing [23] [27]. These steps decrease packet loss and congestion and increase throughput [27]. The Science DMZs are also often limited to specific (and vetted) users and devices, eliminating many of the threats on general-purpose networks and allowing Science DMZs to be equipped with more lenient security policies [23].

B. State of Science DMZ Deployment

The Science DMZ model, since its inception by Dart et al. [20], has seen widespread adoption and evolution, addressing the growing data-intensive demands of scientific research. There are currently more than 200 [22] deployments across various organizations. The model's effectiveness in handling large-scale data transfers has been recognized across various scientific disciplines [28]–[31]. [32] discuss the implementation of medical science DMZs, providing a secure yet high-performance network environment crucial for handling sensitive medical data. Gonzalez et al. [29] and Liu et al. [33] have explored the challenges and solutions in monitoring and optimizing data transfers over international research network connections. These studies underscore the importance of efficient data transfer protocols, as also highlighted by Kissel et al. [19], to support the high-bandwidth requirements of global scientific collaborations. The evolution of Science DMZs encompasses advancements in data rate management using machine learning [34], scalable designs considering the nature of research traffic [21], and explicit feedback mechanisms for congestion control [35]. Gegan et al. [36] and Mazloum et al. [37] have contributed to enhancing security and measurement capabilities within Science DMZs and general purpose networks, addressing the critical need for secure data environments in the wake of cybersecurity threats.

C. Studies on Science DMZ Performance

A few studies have looked at Science DMZ and application performance. A study by Crichigno et al. [38] provides a comprehensive guide to a Science DMZ and describes some performance measurements. It examines TCP attributes, their impact on network performance, the significance of specific data transfer tools and security measures in Science DMZs, and how such software and equipment can create bottlenecks. Vega et al. [35] shows that a P4-based controller that enhances data transfer rates can significantly improve network performance compared to non-dedicated Science DMZ cyberinfrastructure by an average of 21.7%. Calyam et al. [39] present a case study demonstrating the architecture's effectiveness in enhancing remote scientific collaboration and simplifying network management for High-Throughput Computing services. In [40], researchers studied the effect of the Science DMZ on network performance. They show that the DMZ scenario returns the overall best results compared to the no DMZ, no firewall, and no DMZ, no firewall scenarios. There have been several other studies on Science DMZ performance and specific tunings [21], [34]–[37], [41]–[43]. However, these studies focused on particular aspects of a DMZ, such as data transfer performance and network tuning but does not demonstrate quantitative improvements of a DMZ over general-purpose networks.

III. Measurement Infrastructure Setup

In this study, we summarize the tools and infrastructure we used to compare the Science DMZ and the campus commodity network on our university campus.

A. Measurement Tools and Infrastructure

RIPE Atlas: The RIPE Atlas network is a collection of "probes" that conduct measurements and provide a real-time understanding of the condition of the Internet. Probes can conduct ping, traceroute, SSL/TLS, and other measurements to select targets [44]. We utilize RIPE Atlas to perform ping and traceroute to and from servers on our campus.

perfSONAR: perfSONAR (performance Service-Oriented Network monitoring ARchitecture) is an open-source network measurement toolkit [45]. It provides many tools within one package to test and measure network performance. These tools include latency, throughput, trace, and disk-to-disk measurements. perfSONAR identifies areas of poor performance, by both location within the network and by a window of time in which they occur, and flags these problem spots. For this study, we created dedicated perfSONAR nodes and utilized publicly available ones.

Google Cloud: Google Cloud is a platform that is traditionally not used for network measurements. However, in our case, it is evident that several science use cases are utilizing the Google Cloud for their computations. As such, we quantified the network parameters to and from the cloud. **Standard tools:** In addition to these distributed measurement platforms, we utilized several standard tools, such as ping and traceroute. Traceroute provides the option to use both UDP and ICMP, and we utilized both. For performance measurements, we utilized iPerf3 [45] - a command-line tool that measures the throughput between two IP endpoints. It also returns bandwidth, throughput, packet loss, and jitter from the tests. Finally, we used tcpdump, libpcap and Wireshark to capture and analyze traffic traces.

B. Measurement Servers

For these measurements, we created measurement servers within the campus network as well as on the DMZ. Figures 1 and 2 show these servers. The measurement server on the campus network is referred to as Leo. Leo ran a perfSONAR instance and had installed standard tools such as iperf3, ping, and traceroute. On the Science DMZ, we used three other nodes: DTN1, DTN2, and perfSONAR1. We used DTN1 and DTN2 for data transfer experiments and perfSONAR1 for network measurement experiments. Externally, we used RIPE Atlas [44], publicly available perfSONAR nodes and Google Cloud (GCP) for our measurements.

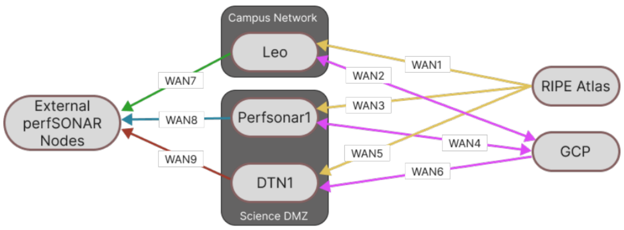


Fig. 1: WAN Routes

C. Network Routes

A commercial ISP provided Layer3 network connectivity to the campus network. Internet2 [46], a network specifically designed to support scientific applications, provided Layer3 connectivity to the Science DMZ. Internally, the campus network was connected to the provider using a 10Gbps link. All traffic passes through a gateway/firewall box that performs packet inspection. The Science DMZ network was connected to Internet2 at 10Gbps. This connection was served by a gateway and a security appliance using access control lists for security. The campus and the Science DMZ network were logically separate. Even though they shared physical fibers, these networks used their own VLANs and traffic was completely separated. Figure 1 shows the external routes. The colored lines in Figure 1 show external (logical) connectivity to external measurement points (mainly RIPE Atlas and GCP). Figure 2 shows local connections between Leo, DTNs, perfSONAR, and the gateways.

D. Experiments

TABLE I: Measurement parameters for comparative analysis.

LAN-side Measurements	WAN-Side Measurements
Throughput	Everything observed on LAN side
RTT between nodes	BGP routes to/from external vantage points
RTT between the node and the gateway	Path length between campus and external vantage points
Jitter	-
Packet loss	-

We summarize our measurement experiments in Table I. For this work, we conducted "ping" tests to measure network latency, packet loss, and jitter. We utilized "traceroute" to collect latency associated with network paths and identify intermediate hops between the source and destination nodes within each route. We utilized Iperf3 to observe throughput between external sources, the campus network, and the DMZ.

We originated these tests inbound from RIPE Atlas and Google Cloud Platform (GCP) virtual machines and outbound from the three local nodes (Leo, perfSONAR1, and DTN1).

1) Internal clients → External servers Experiments: Tests are run with one node of the campus network (Leo) and two nodes of the DMZ (perfSONAR1 and DTN1) posing as clients.

We run ping and traceroute to 12 select perfSONAR nodes within the United States every 30 minutes. We send only ten packets during these tests so as not to overwhelm the external servers. We also used these clients to perform ping and traceroute from GCP VM instances hosted within the United States.

We used Iperf3 throughput experiments between two on-campus clients (Leo and perfSONAR1) and GCP VM instances, which were executed every 12 hours.

We perform the data transfer experiments using Leo and DTN1 as clients. We downloaded Linux ISOs from publicly available mirrors every four hours on both nodes and captured the packet headers using tcpdump. These packet capture datasets allowed us to analyze interpacket delay, packet loss, round-trip time, packet retransmissions and download time. We observed the average daily value of these metrics in Wireshark, and we calculated the average RTT and interpacket delay externally and then plotted the daily values from DTN and Leo side-by-side.

2) External Client → Internal Server Experiments: We ran ping tests from RIPE Atlas to the local nodes every hour and traceroute tests every six hours. The ping and traceroute measurements send three packets of size 48 bytes during each execution. We executed a set of five experiments for each of these tests. For each experiment, we utilized five different RIPE Atlas source probes located within the United States.

Using the same method, we also run the ping and traceroute tests from GCP and the local nodes. Every 30 minutes, ping and traceroute tests run from GCP to the local nodes.

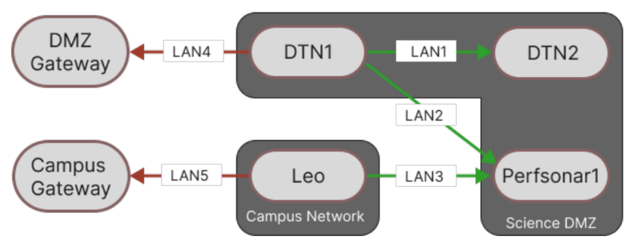


Fig. 2: LAN Routes

3) Internal Clients ↔ Internal Servers Experiments: As previously mentioned, ping tests are performed to measure network latency, packet loss, and jitter, and traceroute tests are conducted to collect latency associated with network paths and identify intermediate hops between the source and destination nodes within each route. These tests are executed between the local network nodes (Leo, perfSONAR1, DTN1, DTN2), as well as between select local network nodes and the gateway to the campus network.

Ping and traceroute tests are executed on these routes using the same method. Every 30 minutes, ping and traceroute tests run from Leo to DTN1, from perfSONAR1 to DTN1, from

DTN1 to DTN2, and from both Leo and DTN1 to the gateway. Ping is designated to send only ten packets during the test.

4) BGP Experiments: For BGP experiments, we utilized a BGP dump from our Science DMZ BGP border router, which we manage. We obtained the BGP routes from our upstream provider on the campus network.

E. Data Analysis

We parsed the collected data from ping, traceroute, and iperf3 into JSON and used Pandas, Seaborn, and Matplotlib to analyze and graph the results.

We examined the ping data to interpret latency, packet loss, and jitter. We analyzed the latency by taking all round-trip time (RTT) occurrences and graphing them with a Cumulative Distribution Function (CDF). We plotted daily packet loss by dividing the sum of all packets lost over a day by all packets sent over a day. We determined jitter by finding the difference in latency of subsequent packets. The jitter is then averaged daily and plotted with the standard deviation from that average.

We used traceroute data to calculate network latency and hop counts associated with network paths. We plot this by categorizing the measurements by the number of hops traversed in the network path and then averaging the latency observed for each route length.

Finally, we used iperf3 and downloaded datasets for throughput insight. We plot this by averaging the bitrates from each day, categorizing them into "sender" and "receiver," and then plotting the averages per day.

IV. Results

In this section, we discuss the comparative results from our experiments. We ran our experiments at regular intervals, as we described in the previous section.

A. Path Lengths

Different upstream providers serve the DMZ and the commodity network in this study. A commercial ISP serves the campus network while the DMZ is served by Internet2, which is a specialized network for research. These experiments compare the path lengths of network destinations to/from internal and external vantage points. Figures 3a and 3d show the average latency and path lengths between RIPE Atlas, Leo (located in the campus network), perfSONAR1, and DTN1 (both located in the DMZ). In both experiments, the maximum hop counts are 19 hops, and the minimum is 8 hops.

The latency and hop counts are lower between these servers and GCP, shown in Figures 3b and 3e. The hop count to these servers is 10 hops compared to 19 from RIPE Atlas. RIPE probes are hosted by various organizations and served by various ISPs. However, Google has a more optimized peering presence, leading to lower hop counts. The latency between GCP and these servers is also lower. Both for the DMZ and the campus network, the maximum latency is 300ms. But the DMZ exhibits lower latency at all route lengths in common with the campus network by $\sim$3% - 6.78%.

As exhibited in Figures 3c and 3f, when traffic is outbound to external perfSONAR nodes, Leo experiences routes with ranges 1-2 hops shorter than DMZ routes, and there is a point when the commodity network performs faster than the DMZ by 12.5% at 10 hops. However, the DMZ tends to have a latency 20% - 36.7% lower than Leo, exclusively comparing common path lengths. Plots of the two DMZ nodes are very similar for this experiment, so Figure 3f was selected to represent both nodes. However, we noticed one difference. The DTN1 node on the DMZ has a latency, at the longest path length of 14 hops, that is $\sim$6.75% lower than that of the perfSONAR1 node on the DMZ. In these outbound experiments, the path lengths are between 7-12 hops on the campus network and 9-14 hops on the DMZ side. Since IP routing can be asymmetric, there is a mismatch between the hop counts from the inbound and the outbound experiments.

Takeaways: Given that a specialized research network serves the DMZ, Internet2, we expected this to have lower hop counts for inbound and outbound traffic. However, the DMZ experiments consistently show higher hop counts than the campus network. This observation is critical for delay-sensitive research applications, such as AR-VR, since moving them into the DMZ will potentially increase their hop count, resulting in end-to-end delay.

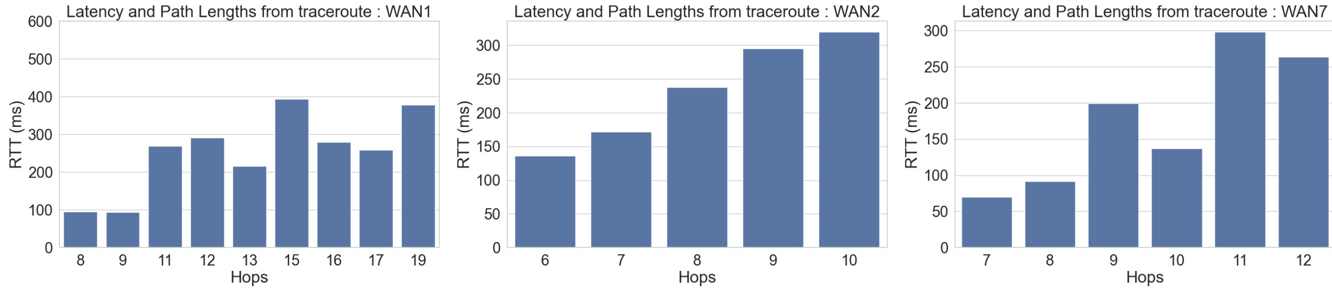
We conclude that just placing research use cases into a DMZ may not automatically improve their performance/latency. Careful discussions and planning with upstream providers are needed to optimize routing and/or physical path. On our campus, we discovered the upstream provider routing traffic using a longer but less congested physical path rather than a short but more heavily used physical path.

B. Latency

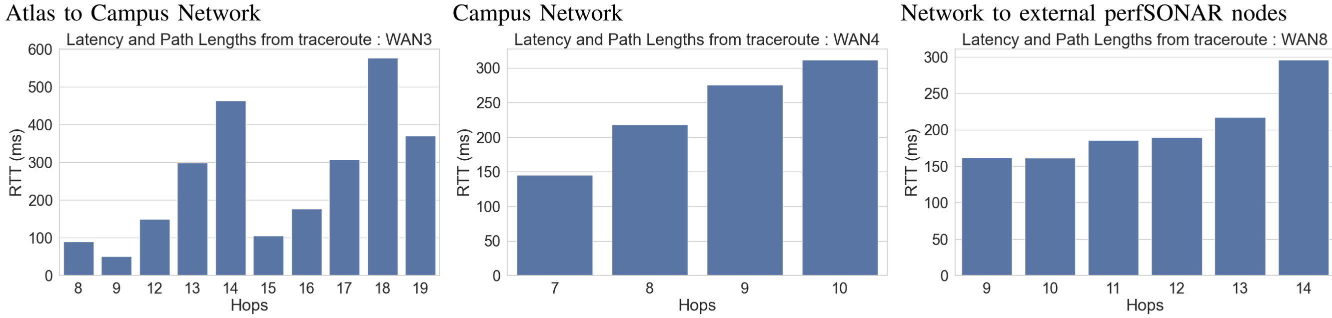
Comparing the distribution in latency in Figures 3a-3f, we find that the latencies are between 100-400ms on the campus network and 50-600ms on the DMZ. There is a significant spike in latency at the penultimate hop (DMZ gateway) for the DMZ experiments. More interestingly, the latencies are slightly higher on the DMZ for outbound experiments since the paths are typically longer. On the paths with higher hop counts, both the campus network and the DMZ experience similar latency as Figures 3c and 3f show.

Figures 4a-4c compare the latency for inbound WAN traffic from RIPE Atlas and GCP and for outbound WAN traffic to perfSONAR nodes. The 95 percentile latency from RIPE Atlas to both the DMZ and campus network is around 80 ms. The 95 percentile latency from GCP to the campus network is around 35 ms, and to the DMZ is near 37 ms. This can again be attributed to better peering provided by GCP.

Based on Figures 3a and 3d, when traffic is inbound from RIPE Atlas, the range of hops is the same to reach Leo and the two DMZ nodes. Comparing only standard path lengths, both nodes on the DMZ have similar latency, represented by Figure 3d. A difference in latency was noted when the route averaged 16 hops for the DMZ nodes. At that point, the DTN1 node had 11% lower latency. Both DMZ nodes often exhibit



(a) Latency and Path Lengths from RIPE Atlas to Campus Network (b) Latency and Path Lengths from GCP to Campus Network (c) Latency and Path Lengths from Campus Network to external perfSONAR nodes



(d) Latency and Path Lengths from RIPE Atlas to DMZ (e) Latency and Path Lengths from GCP to DMZ (f) Latency and Path Lengths from DMZ to external perfSONAR nodes

Fig. 3: Comparison of path lengths from different vantage points

34-73% lower latency than Leo, but Leo has path lengths that have 13-30% lower latency than the DMZ.

Based on Figures 3b and 3e, when traffic is inbound from Google Cloud, both nodes on the DMZ tend to have similar latency, with an occasional ∼2% difference. Due to close similarities in their plots, only Figure 3e represents the DMZ nodes for this experiment. The campus network tends to have similar latency to the DMZ or higher latency by ∼2% - 24%.

For the outbound experiments presented in Figure 4c, 95 percentile latency to external perfSONAR nodes is also around 35 ms on the DMZ side. On the campus network , the 95 percentile latency is near 55 ms. When traffic is outbound to perfSONAR nodes, both nodes on the DMZ exhibit similar latency, while the campus network experiences latency that is 30.43% - 83% slower.

Internally, we find the latency between the campus and DMZ nodes to be very low. However, given that the path length is minimal, the effect of the firewall is really pronounced here. Most pings between campus network servers and the DMZ exhibit a 10ms delay. The inline firewall and access control lists (ACLs) add 8ms latency to each packet, which is very large. Most of these additional delays can be attributed to the firewall and packet inspection middleware.

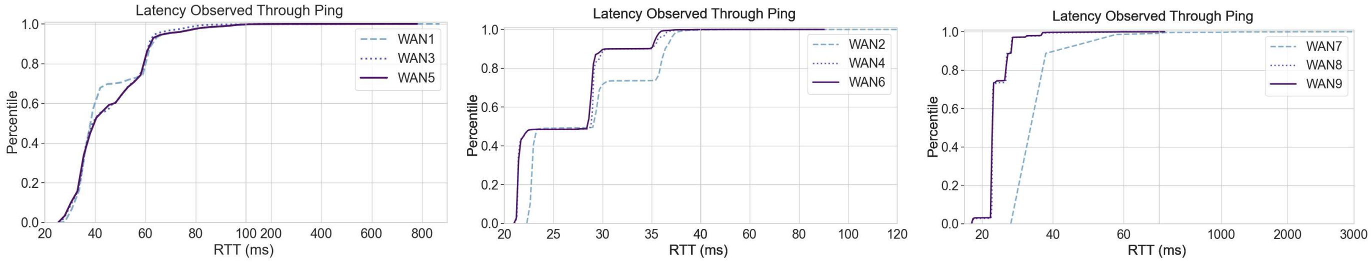
Takeaway: Both the campus network and the DMZ exhibit similar latency but the campus network occasionally shows lower average latency by as much as ∼20ms (5% - 30.5%). We find the measurements often get delayed on the DMZ (e.g., pings not arriving), which affects results poorly. For internal measurements, we find that firewalls negatively affect performance, even when measurement boxes are placed on the same campus/data center.

C. Packet Loss

The DMZ experiences more packet loss than the campus network for inbound traffic from RIPE Atlas. While Leo exhibits a period of 100% packet loss due to the campus node being down, as Figure 5a shows, both nodes of the DMZ experience 50% genuine packet loss even when the network was up. However, the packet loss is more consistent on the campus network, where we can observe 1-2% packet losses.

The perfSONAR1 node on the DMZ exhibits more packet loss than the DTN1 node on the DMZ; it loses ∼5% more packets than DTN1 over three months, as Figure 5a shows. This is potentially because more experiments were conducted on the perfSONAR1 node than on DTN1. When traffic is incoming from Google Cloud, there is no packet loss pattern across DMZ or campus network. When traffic is outbound to external perfSONAR nodes, the campus network experiences more packet loss than the DMZ, but the perfSONAR1 node experiences more packet loss than the DTN1 node. perfSONAR1 exhibits ∼2% more packet loss than DTN1. The campus network exhibits ∼.2% more packet loss than perfSONAR1 and ∼5.7% more packet loss than DTN1 as Figures 5b shows.

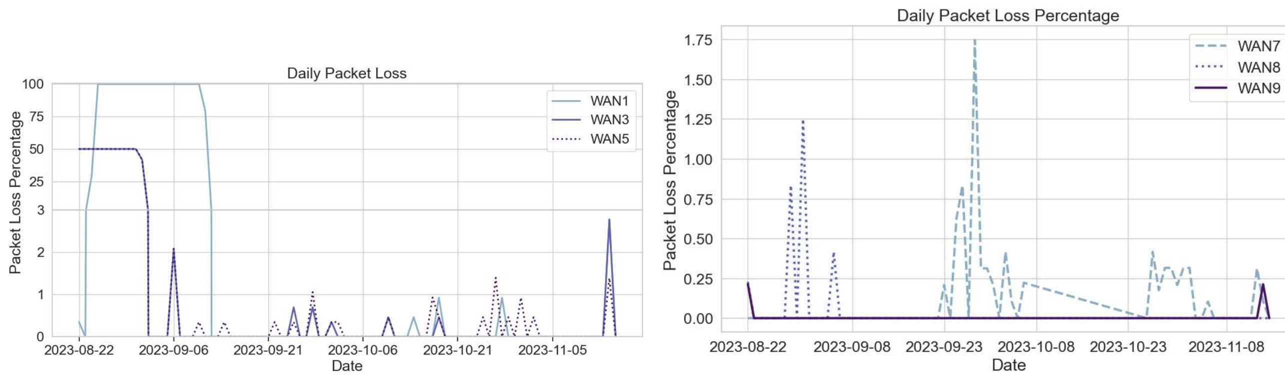
Takeaways: The campus network experiences more regular packet loss. Firewalls and middleboxes contribute to these packet loss events. Packet loss also occurs on the outbound paths from campus, again, potentially due to the presence of firewalls. This observation is important since large data transfers are sensitive to packet loss. Placing research use cases on a shared campus network will affect data transfer performance. Such use cases should be placed in a DMZ network, which has a lower loss rate due to the simplified nature of such networks.



(a) Latency comparison between RIPE Atlas to campus and RIPE Atlas to DMZ nodes

(b) Latency from GCP to campus and DMZ nodes

(c) Latency from campus and DMZ to external perfSONAR nodes



(a) Packet loss from RIPE Atlas to campus and DMZ nodes

(b) Packet loss from campus/DMZ to external perfSONAR nodes

D. Jitter

Jitter is an important matrix for video and other real-time applications. In these experiments, we compare the jitter between the campus network and the DMZ.

When traffic is inbound from RIPE Atlas, Leo, the campus node, exhibits lower average jitter than the DTN1 or perfSONAR1 nodes as Figures 6d and 6e show. Jitter on the campus route tends to be 60-78% lower than on the DMZ routes on average. DTN1 tends to exhibit higher variation in its daily jitter than perfSONAR1 by as much as 37 milliseconds, but the two DMZ nodes exhibit similar overall performance.

When traffic is inbound from GCP to Leo and the DMZ nodes, all three routes exhibit similar average jitter patterns between 0-1 milliseconds, only ever differing by fractions of milliseconds. Figure 6c represents the average jitter pattern, and differences in standard deviation from all three nodes are noted. Leo's route often experiences more deviation in its jitter than the DMZ nodes by as much as two milliseconds. The DTN1 node and perfSONAR1 node experienced a similar jitter pattern, so only the perfSONAR1 plot was selected to convey this experiment. However, the two nodes' difference in variation was noted. The DTN1 node experiences more deviation than the perfSONAR1 node by as much as 1.5 milliseconds.

When traffic is outbound to perfSONAR nodes, as shown in Figures 6a and 6b, Leo and the DMZ nodes typically have an average jitter between 0-1 milliseconds. However, Leo often reaches higher jitter rates up to 10-63 milliseconds greater than the DMZ nodes.

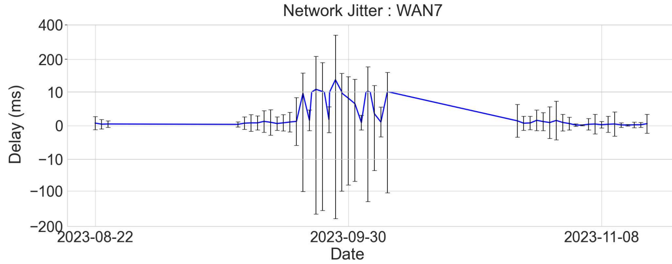
Takeaways: Campus networks experience more jitter than their DMZ counterparts. The average jitter on the campus network is higher due to a higher number of competing flows.

E. Data Transfer Throughput

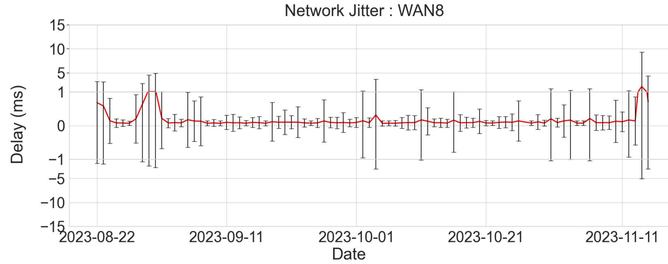
One of the main reasons for creating DMZs is the higher data transfer rate that it enables. This section compares data transfer rates between the DMZ and the campus network. As mentioned earlier, for these tests, we downloaded publicly available Linux ISOs. We performed both experiments back to back to reduce variations in network conditions. Additionally, we did not tune the TCP stacks on the hosts. While such tuning significantly improves the data transfer rates, we wanted to establish a baseline comparison. Further tuning will improve data transfer performance in both DMZ and campus networks.

As Figure 7a shows, the average throughput was much higher on the DMZ when compared to the campus network. The host on the campus network could achieve only 50Mbps, while the host on the DMZ achieved close to 1Gbps. The slower data transfers are a result of packet loss and in-line firewall. On the other hand, the DMZ performs well since it only uses ACLs, and the loss rate is also low.

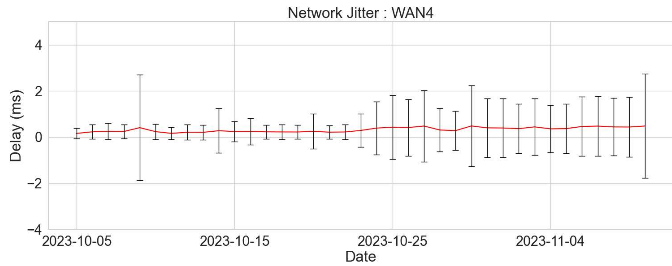
We also looked at the TCP window sizes for these transfers, shown in Figures 7b and 7c. We looked at both the "Bytes out" window size (bytes in flight) and the received window size, and the DTN had more oversized windows in both cases. The received window size was larger on Leo several times, but the throughput was low. This observation is consistent with what we would expect on a lossy link. Figure 7d corroborates these observations. We ran regular iperf3 tests between hosts



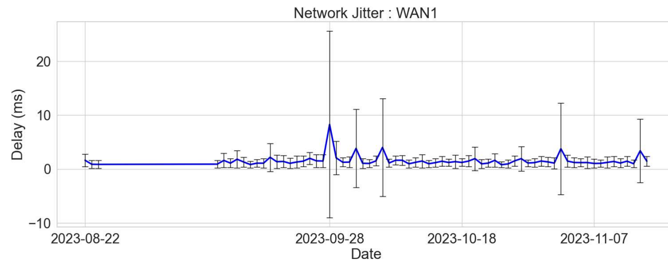
(a) Jitter observed from Campus Network to external perfSONAR nodes



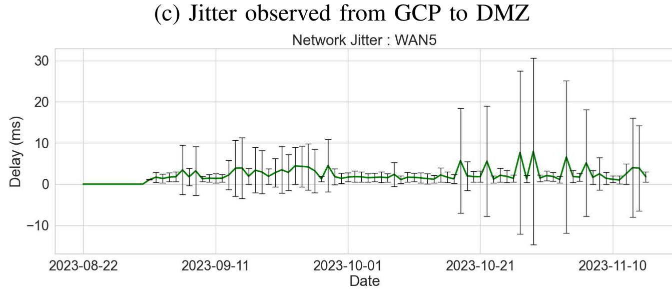
(b) Jitter observed from DMZ to external perfSONAR nodes



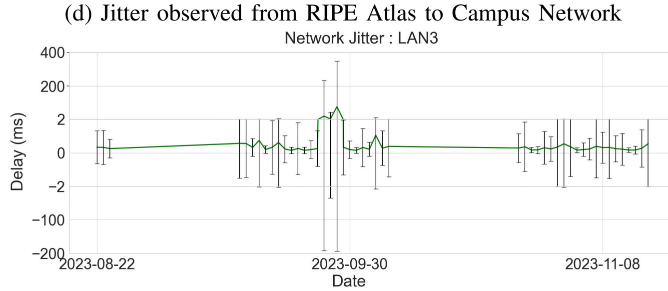
(c) Jitter observed from GCP to DMZ



(d) Jitter observed from RIPE Atlas to Campus Network



(e) Jitter observed from RIPE Atlas to DMZ



(f) Jitter observed from Campus to DMZ

Fig. 6: Jitter Comparison

on Google Cloud, DMZ, and the campus network. The DMZ host consistently outperforms the campus host in both upload and download performance.

Takeaways: The general purpose network performs significantly worse than a DMZ regarding file transfer performance since it has more packet loss, the TCP window is smaller, and firewalls add latency to the packets.

F. BGP Path Comparison

This section compares the BGP path lengths between the DMZ and the campus network. We downloaded the BGP tables from the DMZ BGP router and campus ISP's BGP router. First, we noticed that the commercial ISP had more additional routes than the Science DMZ router. The campus network had 715,810 BGP routes compared to 94,773 on the Science DMZ router. The campus BGP table also had three entries per destination as backup routes. We believe these are artifacts of BGP configurations. Other than having more route options in case of a failure and the capability of better load balancing, more BGP routes provide no additional advantages.

We then compared BGP hop counts between these networks. Figure 7e shows the distribution. The general purpose network generally had a large number of paths with hop counts six or less (note the split Y axis). The DMZ also showed similar patterns. Since the DMZ had less number of routes, we separated the intersection of these two tables and compared them in Figure 7f. We found that the path lengths for the DMZ were slightly lower for shorter-length paths (hop counts <3). For other DMZ routes, the hop count was larger than that of the campus routes. While BGP and IP path lengths are not always strictly correlated, these observations corroborate our findings in the previous experiments.

Takeaways: The DMZ has less path diversity and longer path lengths than the campus network. While this may not directly affect performance, the resiliency of the DMZ can be improved by using additional fallback routes. Further, the path length can be reduced by creating better peering, which requires negotiation with the upstream provider.

V. Conclusions

Science DMZs represent a paradigm shift in network design, tailored explicitly for scientific applications and distinct from traditional campus or general-purpose networks. The core principles of the Science DMZ, such as optimized paths for large data transfers and minimized security interference, position it as an advantageous environment for research and scientific collaboration. Over recent years, its adoption by numerous universities and organizations highlights its value

(a) Average daily throughput observed from Data Transfer Experiments

(b) Average daily "bytes out" window size observed from Data Transfer Experiments

(c) Average daily "received" window size observed from Data Transfer Experiments

(d) Average daily throughput observed from iperf3 experiments to GCP

(e) BGP Path Lengths

(f) BGP Path Lengths

Fig. 7: Comparative Analysis of Data Transfer Metrics and BGP Path Lengths. DTN1 is a Data Transfer Node on the DMZ, Leo is a server on the Campus Network

in the academic and research communities. Our comprehensive study over two years presents a nuanced picture. We confirm that the Science DMZ exhibits lower latency, higher throughput, and better file transfer performance. Packet loss, smaller TCP windows, and added latency from firewalls in campus networks significantly hinder their efficiency in handling large-scale data transfers.

Science DMZs are not without limitations. We observed non-intuitive results such as higher latency in specific scenarios and increased hop counts compared to campus networks. These findings suggest that while the Science DMZ can enhance certain aspects of network performance, it may not uniformly outperform campus networks in all areas, particularly in delay-sensitive applications like AR/VR. Our study reveals that the DMZ has less path diversity and longer path lengths than campus networks. While this impacts performance, strategic enhancements, such as developing better peering agreements and incorporating fallback routes, could mitigate these limitations. In summary, the Science DMZ model offers distinct advantages for specific research applications. However, it is not a one-size-fits-all solution. Such deployments must be carefully tailored to the particular needs and use cases of the communities they serve.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. OAC-2018373. We also thank David Hales and Michael Renfro for their help with setting up the measurements.

REFERENCES

[1] S. Mastorakis, A. Skiadopoulos, S. Shannigrahi, A. Likens, B. Nour, and N. Stergiou, "Networking and computing in biomechanical research: Challenges and directions," *IEEE Communications Magazine*, vol. 59, no. 6, pp. 103–109, 2021.

[2] M. W. Al Azad, S. Shannigrahi, N. Stergiou, F. R. Ortega, and S. Mastorakis, "Cledge: A hybrid cloud-edge computing framework over information centric networking," in *2021 IEEE 46th Conference on Local Computer Networks (LCN)*. IEEE, 2021, pp. 589–596.

[3] S. Liu, V. Patil, T. Yu, A. Afanasyev, F. A. Feltus, S. Shannigrahi, and L. Zhang, "Designing hydra with centralized versus decentralized control: A comparative study," in *Proceedings of the Interdisciplinary Workshop on (de) Centralization in the Internet*, 2021, pp. 4–10.

[4] C. Papadopoulos, A. Afanasyev, and S. Shannigrahi, "A name-based secure communications architecture for vehicular networks," in *2021 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2021, pp. 178–181.

[5] I. Dasgupta, S. Shannigrahi, and M. Zink, "A hybrid ndn-ip architecture for live video streaming: A qoe analysis," in *2021 IEEE International Symposium on Multimedia (ISM)*. IEEE, 2021, pp. 148–157.

[6] M. A. Islam, M. Ismail, R. Atat, and S. Shannigrahi, "Software-defined routing strategy based on reinforcement learning in smart power grid," in *SoutheastCon 2022*. IEEE, 2022, pp. 64–70.

[7] I. Dasgupta, S. Shannigrahi, and M. Zink, "A hybrid ndn-ip architecture for live video streaming: From host-based to content-based delivery to improve qoe," *International Journal of Semantic Computing*, vol. 16, no. 02, pp. 163–187, 2022.

[8] Z. Threet, C. Papadopoulos, W. Lambert, P. Podder, S. Thanasoulas, A. Afanasyev, S. Ghafoor, and S. Shannigrahi, "Securing automotive architectures with named data networking," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 2663–2668.

[9] R. Thompson, M. Ismail, and S. Shannigrahi, "Vehicle-to-vehicle charging coordination over information centric networking," in *2022 IEEE 47th Conference on Local Computer Networks (LCN)*. IEEE, 2022, pp. 363–366.

[10] Y. Wu, F. V. Mutlu, Y. Liu, E. Yeh, R. Liu, C. Iordache, J. Balcas, H. Newman, R. Sirvinskas, M. Lo *et al.*, "N-dise: Ndn-based data

distribution for large-scale data-intensive science," in *Proceedings of the 9th ACM Conference on Information-Centric Networking*, 2022, pp. 103–113.

[11] E. Yeh, H. Newman, L. Zhang, J. Cong, S. Shannigrahi, Y. Wu, C. Iordache, V. Mutlu, S. Timilsina, S. Song *et al.*, "N-dise: Ndn-based data distribution for large-scale data-intensive science," 2022.

[12] I. D. S. M. M. Z. R. Thompson, E. Lyons and S. Shannigrahi, "An information centric framework for weather sensing data," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*. 10.1109/ICCWorkshops53468.2022.9882164, 2022, pp. 1–6.

[13] M. A. Islam, M. Ismail, R. Atat, O. Boyaci, and S. Shannigrahi, "Software-defined network-based proactive routing strategy in smart power grids using graph neural network and reinforcement learning," *E-Prime-Advances in Electrical Engineering, Electronics and Energy*, vol. 5, p. 100187, 2023.

[14] J. Presley, X. Wang, X. Ai, T. Yu, T. Brandel, P. Podder, V. Patil, A. Afanasyev, F. Feltus, L. Zhang *et al.*, "Hydra: A scalable decentralized p2p storage federation for large scientific datasets," in *2024 International Conference on Computing, Networking and Communications (ICNC)*, 2024, pp. 1–7.

[15] X. Wang, X. Ai, F. A. Feltus, and S. Shannigrahi, "Gnsga: A decentralized data replication algorithm for big science data," in *2023 IFIP Networking Conference (IFIP Networking)*. IEEE, 2023, pp. 1–9.

[16] J. S. S. S. Sankalpa Timilsina, Davide Pesavento and L. Benmohamed., "Capture and analysis of traffic traces on a wide-area ndn testbed," in *ACM Conference on Information-Centric Networking (ACM ICN '23)*, 2023, pp. 1–7.

[17] C. Iordache, S. Shannigrahi, Y. Wu, S. Song, F. V. Multu, J. Balcas, R. Širvinskas, S. Timilsina, D. Pesavento, H. Newman *et al.*, "A named data networking based fast open storage system plugin for xrootd," in *EPJ Web of Conferences*, vol. 295. EDP Sciences, 2024, p. 01004.

[18] R. K. Thelagathoti, S. Mastorakis, A. Shah, H. Bedi, and S. Shannigrahi, "Named data networking for content delivery network workflows," in *2020 IEEE 9th International Conference on Cloud Networking (CloudNet)*. 10.1109/CloudNet51028.2020.9335806, 2020, pp. 1–7.

[19] E. Kissel, M. Swany, B. Tierney, and E. Pouyoul, "Efficient wide area data transfer protocols for 100 gbps networks and beyond," in *Proceedings of the Third International Workshop on Network-Aware Data Management*, ser. NDM '13. New York, NY, USA: Association for Computing Machinery, 2013. [Online]. Available: https://doi.org/10.1145/2534695.2534699

[20] E. Dart, L. Rotman, B. Tierney, M. Hester, and J. Zurawski, "The science dmz: A network design pattern for data-intensive science," ser. SC '13. New York, NY, USA: Association for Computing Machinery, 2013. [Online]. Available: https://doi.org/10.1145/2503210.2503245

[21] C. Lee, M. Jang, M. Noh, and W. Seok, "Scalable design and algorithm for science dmz by considering the nature of research traffic," *The Journal of Supercomputing*, vol. 77, pp. 2979–2997, 2021.

[22] "NSF Award Search: Simple Search Results. — nsf.gov," https://www.nsf.gov/awardsearch/simpleSearchResult ?queryText=dmz, [Accessed 13-03-2024].

[23] J. Crichigno, E. Bou-Harb, and N. Ghani, "A comprehensive tutorial on science dmz," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 2041–2078, 2019.

[24] C. Ogle, D. Reddick, C. McKnight, T. Biggs, R. Pauly, S. P. Ficklin, F. A. Feltus, and S. Shannigrahi, "Named data networking for genomics data management and integrated workflows," *Frontiers in Big Data*, vol. 4, no. 1, p. 16, 2021.

[25] S. Shannigrahi, S. Mastorakis, and F. R. Ortega, "Next-generation networking and edge computing for mixed reality real-time interactive systems," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2020, pp. 1–6.

[26] S. Shannigrahi and C. Partridge, "Big data, transmission errors, and the internet," in *2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S)*. IEEE, 2023, pp. 142–145.

[27] The Energy Science Network. [Online]. Available: https://www.es.net

[28] S. Peisert, E. Dart, W. Barnett, E. Balas, J. Cuff, R. L. Grossman, A. Berman, A. Shankar, and B. Tierney, "The medical science dmz: a network design pattern for data-intensive medical science," *Journal of the American Medical Informatics Association*, vol. 25, no. 3, pp. 267–274, 2018.

[29] A. Gonzalez, J. Leigh, S. Peisert, B. Tierney, E. Balas, P. Radulovic, and J. M. Schopf, "Big data and analysis of data transfers for international research networks using netsage," in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, pp. 344–351.

[30] J. Presley, X. Wang, T. Brandel, X. Ai, P. Podder, T. Yu, V. Patil, L. Zhang, A. Afanasyev, F. A. Feltus *et al.*, "Hydra–a federated data repository over ndn," *arXiv preprint arXiv:2211.00919*, 2022.

[31] H. Lim, A. Ni, D. Kim, Y.-B. Ko, S. Shannigrahi, and C. Papadopoulos, "Ndn construction for big science: Lessons learned from establishing a testbed," *IEEE Network*, vol. 32, no. 6, pp. 124–136, 2018.

[32] S. Peisert, E. Dart, W. Barnett, E. Balas, J. Cuff, R. L. Grossman, A. Berman, A. Shankar, and B. Tierney, "The medical science DMZ: a network design pattern for data-intensive medical science," *Journal of the American Medical Informatics Association*, vol. 25, no. 3, pp. 267–274, 10 2017. [Online]. Available: https://doi.org/10.1093/jamia/ocx104

[33] Z. Liu, P. Balaprakash, R. Kettimuthu, and I. Foster, "Explaining wide area data transfer performance," in *Proceedings of the 26th International Symposium on High-Performance Parallel and Distributed Computing*, ser. HPDC '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 167–178. [Online]. Available: https://doi.org/10.1145/3078597.3078605

[34] C. V. Caicedo, E. F. Kfoury, J. Gomez, J. E. Pezoa, M. Figueroa, and J. Crichigno, "Machine learning controller for data rate management in science dmz networks," *Computer Networks*, p. 110237, 2024.

[35] C. F. Vega Caicedo *et al.*, "Explicit feedback for congestion control in science dmz cyberinfrastructures based on programable data-plane switches." 2023.

[36] R. Gegan, C. Mao, D. Ghosal, M. Bishop, and S. Peisert, "Anomaly detection for science dmzs using system performance data," in *2020 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2020, pp. 492–496.

[37] A. Mazloum, J. Gomez, E. Kfoury, and J. Crichigno, "Enhancing perfsonar measurement capabilities using p4 programmable data planes," in *Proceedings of the SC'23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis*, 2023, pp. 819–829.

[38] J. Crichigno, E. Bou-Harb, and N. Ghani, "A comprehensive tutorial on science dmz," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 2041–2078, 2018.

[39] P. Calyam, A. Berryman, E. Saule, H. Subramoni, P. Schopis, G. Springer, U. Catalyurek, and D. K. Panda, "Wide-area overlay networking to manage science dmz accelerated flows," in *2014 International Conference on Computing, Networking and Communications (ICNC)*, 2014, pp. 269–275.

[40] B. Rababah, S. Zhou, and M. Bader, "Evaluation the performance of dmz," *International Journal of Wireless and Microwave Technologies*, vol. 8, no. 1, pp. 1–13, 2018.

[41] I. Abhinit, H. Addleman, K. Benninger, D. DuRousseau, M. Krenz, and B. Meade, "Science dmz: Secure high performance data transfer," Tech. Rep., 2022.

[42] J. Crichigno, E. Kfoury, E. Bou-Harb, and N. Ghani, "Application and security aspects for large flows," in *High-Speed Networks: A Tutorial*. Springer, 2021, pp. 329–341.

[43] ——, "Data-link and network layer considerations for large data transfers," *High-Speed Networks: A Tutorial*, pp. 105–213, 2021.

[44] "Atlas console," 2022. [Online]. Available: https://atlas.ripe.net/new-docs/

[45] J. Zurawski, S. Balasubramanian, A. Brown, E. Kissel, A. Lake, M. Swany, B. Tierney, and M. Zekauskas, "perfsonar: On-board diagnostics for big data," in *Workshop on Big Data and Science: Infrastructure and Services*. Citeseer, 2013.

[46] M. Kratz, M. Ackerman, T. Hanss, and S. Corbato, "Ngi and internet2: Accelerating the creation of tomorrow's internet," in *MEDINFO 2001*. IOS Press, 2001, pp. 28–32.