The DARPA SEARCHLIGHT Dataset of Application Network Traffic

Calvin Ardi
USC/ISI
calvin@isi.edu

Connor Aubry
Sandia National Laboratories
caubry@sandia.gov

Brian Kocoloski
USC/ISI
bkocolos@isi.edu

David DeAngelis
USC/ISI
deangeli@isi.edu

Alefiya Hussain
USC/ISI
hussain@isi.edu

Matt Troglia
Sandia National Laboratories
mtrogli@sandia.gov

Stephen Schwab
USC/ISI
schwab@isi.edu

ABSTRACT

Researchers are in constant need of reliable data to develop and evaluate AI/ML methods for networks and cybersecurity. While Internet measurements can provide realistic data, such datasets lack ground truth about application flows. We present a ~750GB dataset that includes ~2000 systematically conducted experiments and the resulting packet captures with video streaming, video teleconferencing, and cloud-based document editing applications. This curated and labeled dataset has bidirectional and encrypted traffic with complete ground truth that can be widely used for assessments and evaluation of AI/ML algorithms.

CCS CONCEPTS

• **Networks** → *Application layer protocols*; **Network experimentation**; *Network measurement*; • **Information systems** → *Internet communications tools*.

KEYWORDS

datasets, network experimentation, network traffic

ACM Reference Format:
Calvin Ardi, Connor Aubry, Brian Kocoloski, David DeAngelis, Alefiya Hussain, Matt Troglia, and Stephen Schwab. 2022. The DARPA SEARCHLIGHT Dataset of Application Network Traffic. In *Cyber Security Experimentation and Test Workshop (CSET 2022), August 8, 2022, Virtual, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3546096.3546103

1 INTRODUCTION

Artificial intelligence and machine learning (AI/ML) methods are widely used to understand and develop networked and distributed systems. However, datasets to train and develop such systems are scarce and require knowing the complete ground truth to evaluate such methods. Additionally, there are many challenges in collecting and curating datasets for network traffic. Each network has unique characteristics with inherently stochastic traffic dynamics. Network traffic can have anomalies and misbehaviors that complicate creating training data sets, and contain sensitive personally identifiable information (PII) or intellectual property data, limiting wide access.

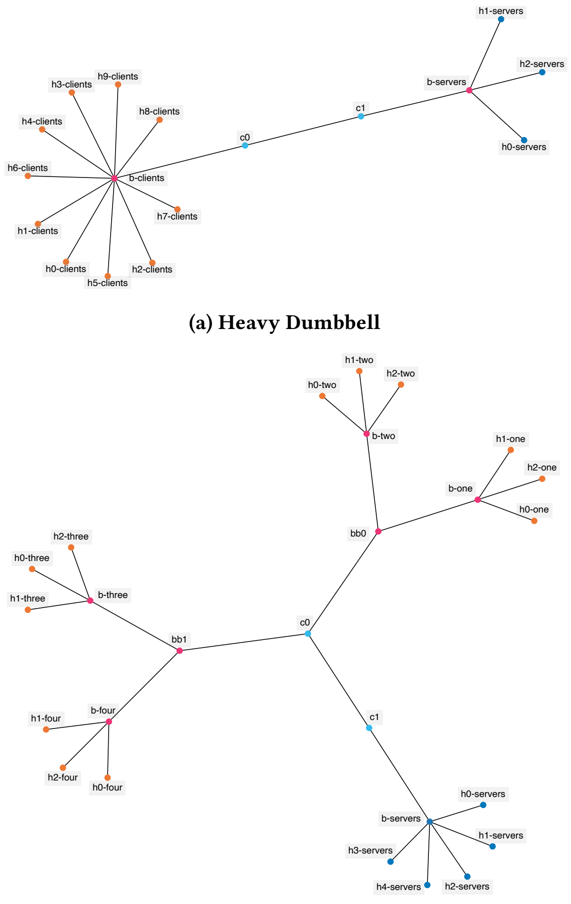
To help address the scarcity of publicly available networking datasets and enable networking research, we present a network traffic dataset that was systematically collected, curated, and labeled on an emulation testbed. Many researchers collect one-off network traffic datasets, draw conclusions, subject the comparisons to peer review, and publish results. While such paper-based datasets allow for inference of properties and behavior, they do not support direct assessments. This dataset was collected for the DARPA SEARCHLIGHT evaluation effort [7]. We believe that sharing this dataset will enable repeatable and directly comparative assessments of next generation networking technologies and applications in cybersecurity, traffic engineering, and network measurement.

The DARPA SEARCHLIGHT dataset, while *generated*, as it was collected on a emulation testbed, is a unique resource in several ways. First, it is *complete*: the bidirectional traffic from all the sources and destinations is captured. Second, it is *labeled*: all the flows in the network traffic are identified and associated with an application. Third, the traffic flows in this dataset have *varying levels of complexity*. Some traffic captures have only one application flow while some traffic captures have several simultaneous applications and flows. Finally, the dataset contains *multiple repeated samples* to account for the stochastic and dynamic nature of network traffic. We believe that this combination of dataset features will enable a wide range of AI/ML methods for network traffic analysis to be systematically developed and evaluated.

The COVID-19 pandemic resulted in major shifts in Internet traffic composition and patterns [25]. In building the dataset, we focused on using three *contemporary traffic applications*, video streaming, video teleconferencing, and cloud-based services, over both well-established transport protocols (TCP, UDP, HTTP) and the recently standardized QUIC. Additionally, ascertaining information in encrypted traffic, which has increased significantly with work-from-home and remote work [13], is not possible in most datasets. We include in this dataset a large collection of traces with IPsec [10] and WireGuard [9] encryption.

CSET 2022, August 8, 2022, Virtual, CA, USA

ACM ISBN 978-1-4503-9684-4/22/08…$15.00
https://doi.org/10.1145/3546096.3546103



(a) Heavy Dumbbell

(b) Tiered

Figure 1: Network Topologies in the SEARCHLIGHT Dataset

Our contribution is the initial release of the DARPA SEARCHLIGHT dataset, containing combinations of 2 network topologies, 3 major contemporary applications, and 2 encryption configurations. In this paper, we explain our design decisions in creating the dataset and characterize its content. In § 2, we describe the network topologies and software configurations used in the dataset, and we detail the traffic generator applications in § 3. We explain in § 4 how we build experiment scenarios using these topologies and applications. We describe dataset's contents in § 5, covering the use cases and context of our experiments in § 6. We compare in § 7 where our datasets fit in the spectrum of available datasets today and conclude in § 8. The DARPA SEARCHLIGHT dataset is freely and publicly accessible at https://mergetb.org/projects/searchlight/.

2 NETWORK CONFIGURATION

We design the network configurations in the DARPA SEARCHLIGHT dataset to provide a controlled environment that roughly approximates an enterprise network, while balancing network topology sizes and resource constraints. We generate the dataset on a network emulation testbed operated by the Merge testbed software platform [3], which provides services to help users define and instantiate network topologies, deploy routing, and automate experiments in an isolated and controlled environment.

Table 1: Link Capacities in Heavy Dumbbell Topology

Link(s)	Capacity (Mbps)
All end-hosts to their border router	30
All inter-router (border + core) links	50

Table 2: Link Capacities in Tiered Topology

Link(s)	Capacity (Mbps)
All end-hosts to their border router	10
`b-{one,two}` ↔ `bb0`	50
`b-{three,four}` ↔ `bb1`	50
`b-servers` ↔ `c1`	80
`{bb0,bb1}` ↔ `c0`	80
`c0` ↔ `c1`	100

Network Topologies. We deploy two different small-scale network topologies (Fig. 1): *heavy dumbbell*, which represents a deployment of nodes in a standard client/server model; and *tiered*, a tree-like hierarchical topology with traffic generating clients/servers at the leaves and multiple levels of routers above. These topologies were chosen because they are simple and well understood in the networking community and are large enough to support the number of clients and servers needed to generate *representative* datasets from the traffic generators we used. In comparison to the heavy dumbbell, the composition of routers in the tiered topology leads to interleaving of flows which results in more realistic and diverse packet forwarding behaviors in the network.

Heavy dumbbell (Fig. 1a) consists of 17 nodes with clients on one side and servers on the other: 10 clients connected to a border router, 3 servers connected to a border router, and 2 intermediate core routers that form a line with the border routers. Tiered (Fig. 1b) has 26 nodes: 4 client enclaves, each of which contain 3 clients connected to a border router (`b-{one,two,three,four}`), and 1 server enclave of 5 servers connected to its border router (`b-servers`). The enclaves are connected in a tree-like hierarchical structure rooted at common router node `c0`.

The Merge testbed allows users to configure bandwidth capacities (up to 10 Gbps), delay, and loss on network links. Table 1 and Table 2 show the capacities used for the heavy dumbbell and tiered topologies, respectively. We configure bandwidth capacity constraints to 10–100 Mbps (lower at the edges and higher at the core) due to certain technology limitations, leaving delay as default (sub-millisecond) and loss at 0 %. We additionally configure each link to be symmetric, or the same capacity in either direction, to reflect the enterprise environment of our topologies (compared to asymmetric speeds commonly found in residential or mobile connections).

Software Configuration. All nodes run an unmodified, baseline Ubuntu 20.04 LTS operating system. We create Ansible playbooks to install traffic generators (§ 3), data sources, and software dependencies. These playbooks make the setup portable to any other experiment environment without requiring custom OS images.

Table 3: Applications and Configurable Features

App.	Configurable Settings
`cbs`	speed := { fast (400 char/min), medium (200), slow (60) }
	bursty := { true, false }
	length := { *n* chars }
`vtc`	video := { true, false }
`video`	resolution := { 576p, 720p, 1080p }
	streaming := { DASH, HLS, HTML5 }
	transport := { HTTP/1.1, HTTP/1.1+TLS, HTTP/2, HTTP/3 }

All nodes are deployed as QEMU/KVM [4] virtual machines (VM) using the testbed's built-in node creation mechanisms. The testbed configures the VM's network interfaces and routing tables using systemd-networkd scripts. The testbed configures network links using a combination of network virtualization mechanisms on the underlying hypervisors and switches, including VLAN and VxLAN for network segment isolation and TAP devices for VM-based access to the network fabric.

Virtual Private Networks (VPN). With the continued growth of VPNs in many networked systems (especially due to COVID-19 [12]), a modern dataset should also include traffic captures where noes communicate over a VPN. Our dataset includes two VPN configurations: *site-to-site* (STS) and *point-to-point* (PTP). STS is common in distributed enterprise networks and used to logically combine physically separated networks. PTP, sometimes called a "road warrior" configuration, is often used by roaming clients connecting to a remote network. Each VPN is implemented using IPsec [10] and WireGuard [9] software.

In STS, a VPN tunnel is established between each enclave at each border router (`b-clients` ↔ `b-servers` in Fig. 1a and `c0` ↔ `b-servers` in Fig. 1b). The tunnels in our PTP configurations are between a single client and server: a client will tunnel all of its traffic through the server (`h0-clients` ⟶ `h0-servers`).

In the experiments that use VPN tunnels (STS or PTP), we provide packet captures containing encrypted traffic (captured on node `c0` in Fig. 1a and `c1` in Fig. 1b) as well as captures with unencapsulated traffic (captured on a traffic generating client/server end-host).

3 APPLICATIONS

Network traffic in DARPA SEARCHLIGHT is generated by three applications: cloud-based document editing, video teleconferencing, and video streaming. We select these applications based on their overall traffic share and applicability to enterprise settings. Video streaming represented 54 % of global Internet traffic in 2021 [26] (58 % in 2020 [25]). Similarly, video teleconferencing and collaborative document editing are essential tools in today's business environment. For reproducibility and control, we select and use representative applications of popular Internet services that can be run entirely within a testbed.

Cloud-based Document Editing (`cbs`). In `cbs`, one or more clients simultaneously edit a shared document on a remote server (akin to Google Docs or Microsoft Office 365). We instrument a web browser client using the Playwright web testing and automation framework [22] to type text in a document served by Etherpad [14]. To provide diversity in traffic patterns, we can configure text editing by its input speed (fast: 400 characters/min, medium: 200, slow: 60), burstiness (whether to pause randomly between words), and length (number of total characters to input).

Video Teleconferencing (`vtc`). VTC is two-way traffic containing audio and video streams between two or more endpoints (Zoom, Skype, Google Meet). We use an open-source VTC traffic generator [8] which emulates a live, multi-party audio-video conference with two or more clients having a conversation on Jitsi Meet [1], an open-source video conferencing software using WebRTC. During a video conference, each client takes a turn to "speak" some dialog (to avoid overlapping), pre-generated using GPT-2 [23] and a text-to-speech synthesizer, while displaying video from their "webcam", a virtual camera device that is streaming pre-generated video files. While the source video files are 1080p resolution, Jitsi Meet will dynamically adjust the streaming quality based on network conditions. `vtc` is configurable to any number of clients and each conference can run audio-only or with audio and video.

Video Streaming (`video`). Video streaming is defined as video traffic that is streamed on-demand (as opposed to live) from a server to client (like YouTube or Vimeo). In `video`, one or more clients connect to and stream a video from a remote server which hosts both video files and supporting website code (HTML, JavaScript). We use and extend an open-source video streaming traffic generator [2], which supports multiple video streaming (DASH, HLS, HTML5) and transport protocols (HTTP/1.1, HTTP/2, HTTP/3). Each client can be configured to watch video at a specified resolution over a streaming and transport protocol for some length of time. The server provides a static website with precomputed audio/video files and fragments supporting each resolution and format. We extend this traffic generator by instrumenting the client and server to record Quality of Experience (QoE) metrics and building experiment orchestration tools to use this generator across an arbitrary number of clients.

4 EXPERIMENT SCENARIOS

Using the topologies and applications described in the previous sections, we now develop and run several experiment scenarios. We create a diverse set of scenarios in order to highlight different variations of network traffic conditions, and we control the complexity in each scenario with well-defined application flows. Providing scenarios that range in complexity enable us to easily label and others to validate the ground truth in their own experiments.

4.1 Scenario Configurations

There are three general experiments in the DARPA SEARCHLIGHT dataset, one for each application (`cbs`, `vtc`, `video`). Each experiment: (1) has configurations for all combinations of configurable parameters in each traffic generator, (2) has separate runs with varying numbers of hosts, and (3) is ran over both network topologies.

Additionally, we control for the following variables:

clients one or more nodes as application clients
servers one or more nodes as application servers
application traffic generator deployed at client(s) and server(s)
application configuration varies per application; see Table 3
packet capture node node where packet capture was collected
encryption if a VPN tunnel was deployed
encryption parameters VPN tunnel configuration

Table 4: Number of Clients per Experiment Type

App.	exp-a	exp-b	exp-c
cbs	1	3	10
vtc	3	5	8
video	1	3	5

We conduct multiple experiment runs with increasing number of client hosts in order to emulate reasonably busy network conditions. Table 4 details the number of clients per application and experiment type. For example, we initially start with 1 client each in cbs and video to provide a baseline. We start with 3 clients in vtc specifically to make use of the centralized signaling server (a conference with 2 clients happens peer-to-peer).

Combining all application configurations and host variations, there are a total of 132 experiment scenarios in the DARPA SEARCHLIGHT dataset: 18 cbs, 108 video, and 6 vtc. Each scenario is run on both the heavy dumbbell and tiered topologies, with and without encryption using IPSec/WireGuard and their corresponding STS/PTP configurations (§ 2).

Packet captures are organized into their experiment's folder and the filename for any given run contains the host and application settings. For example, a packet capture located in experiment_1/exp-a-medium-false uses the cbs application with 1 client, medium typing speed, and typing burstiness disabled. As another example, experiment_3/exp-b-720-html5-http2 uses video with 3 clients watching video at 720p resolution, HTML5 streaming, over HTTP/2.

4.2 Scenario Execution

Executing a scenario is a controlled process of three major steps: configuration, running, and teardown. Like provisioning the experimental nodes in our testbed (§ 2), we use Ansible playbooks to orchestrate all aspects of an experiment scenario. Fig. 2 further breaks down these major steps. We first set up the topology (with VPNs if enabled), and install and provision the application traffic generators. Once configuration is complete, we start packet capture at specified nodes and launch traffic generator server and client applications on their respective nodes. The scenario is run for roughly 200 s before stopping the traffic generators and packet captures. We repeat the scenario (starting at packet captures) 3 times in total and then archive the measurements and tear down the topology.

5 ACCESSING THE DATASET

The DARPA SEARCHLIGHT dataset is ~750GB with ~2000 experiment runs. The dataset and documentation summarizing its contents are available at https://mergetb.org/projects/searchlight/.

Each experimental run includes: (1) a packet capture collected on a core router node (c0 or c1) that sees all traffic, as well a as pre-tunneled capture collected on a traffic generating end-host for the VPN configurations; (2) a summary text file describing all of the UDP, TCP, and IP (non-UDP/TCP) flows in the packet capture, including the number of packets and number of bytes per flow; (3) a CSV showing the packets per second (pps) per flow and bytes per second (bps) per flow over time; and (4) graphs showing the pps and bps per flow type over time. Each experiment also has a webpage with links to the data and summaries.

6 USE CASES AND CONTEXT

We initially collected the DARPA SEARCHLIGHT dataset for an evaluation effort of the same name [7, 20], and we believe that other researchers and students can use this dataset to directly compare and test their own technologies in traffic engineering, traffic analysis, and network measurement.

We used the dataset to evaluate the performance of traffic classification in plaintext and encrypted network flows, network topology inference and path discovery, and dynamic quality of service (QoS) enforcement on enterprise networks (§ 2). Similarly, other researchers used part the data to build and iterate on the aforementioned technologies.

Many of the techniques use AI/ML and researchers used the labeled ground truth data to develop and train their algorithms. After training on the simple scenarios, researchers were able to evaluate and further train on scenarios of increasing complexity (we covered configuration variations in experiment scenarios in § 4.1).

To easily understand an experiment's overall network behavior, we build minimal graphs for each flow. Fig. 3 shows visual representations of network flows of bytes (y-axis) over time (x-axis), binned by 1 s, in three different experiments. Each sparkline represents an individual flow (the truncated IP prefix is the same across all flows: 10.0.) and is deliberately minimized to emphasize a flow's activity relative to others. Flows from the same source node are of the same color and line style.

One of the advantages of using generated data is the minimization of noise and other background traffic. For example, researchers training on video streaming traffic are able to see its dynamics without additional interference. Fig. 3c shows 3 clients filling their initial video buffer (0–45 s) and the sawtooth pattern after 45 s represents the periodic buffer fills afterwards. For video streaming, prior work [2] has shown that these generators are roughly approximate of real-world traffic—we plan on introducing "noise" and background traffic as future work.

Fig. 3a and Fig. 3b show the diversity in network traffic behavior for cloud-based document editing (cbs) and video teleconferencing (vtc), respectively. In cbs, we see a mix of short, bursty flows and long-lived, continuous low-bitrate flows, and in vtc, we see long-lived flows over both TCP and UDP.

These different dynamics for each application adds unique challenges for developing and evaluating AI/ML for networking and distributed systems.

7 OTHER DATASET EFFORTS

The lack of representative datasets in the networking and cybersecurity community is well recognized [6, 15]. Many networking datasets are generally focused on specific measurements on the Internet with little or no ground truth [6].

Ring *et al.* [24] present a comprehensive overview of many datasets. Often these datasets have interesting features, but can either be unlabeled, heavily anonymized, or have a specific focus that makes them unrepresentative of actual traffic. For example, some datasets included slightly contrived anomalous behavior in order to provide enough interesting data.

There are several existing data repositories like CAIDA [18], Crawdad [27] and measurement platforms, such as RIPE Atlas [5],

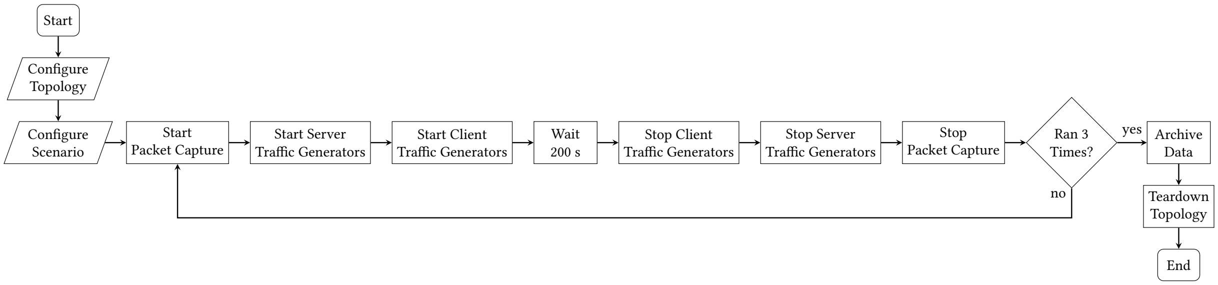
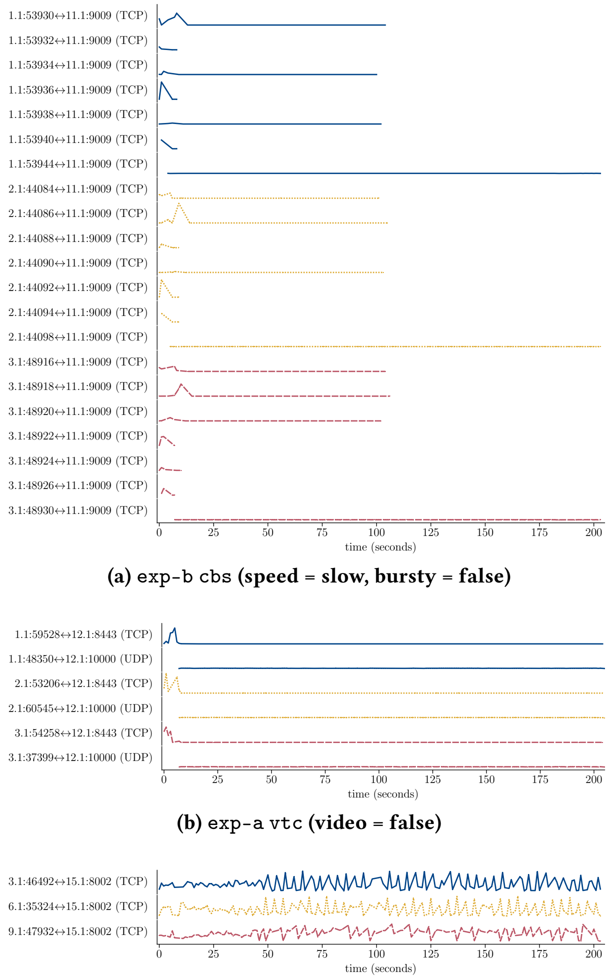


Figure 2: Flowchart for Executing an Experiment Scenario



(a) `exp-b cbs` **(speed = slow, bursty = false)**

(b) `exp-a vtc` **(video = false)**

(c) `exp-b video` **(res. = 576p, streaming = DASH, transport = HTTP/2)**

Figure 3: Visual Representation of Experiment Flows

Measurement Lab [11], LANDER [17], PREDICT [16]. While these datasets or platforms focus on specific types of network measurements (traffic flows, traceroute, DNS queries, wireless contact traces) they do not have the full, raw packet captures or complete visibility of a network as presented in the DARPA SEARCHLIGHT dataset.

The popular DARPA cybersecurity datasets [21], collected more than two decades ago, focuses on cyber attacks and while still widely used, are quite outdated. The LANL dataset [19] is derived from real-world enterprise data and focuses on end-host logs and flows. Due to heavy anonymization, this dataset has limited usability for a wide range of research studies [6].

8 FUTURE WORK AND CONCLUSION

In this paper we presented the DARPA SEARCHLIGHT dataset consisting of increasingly prevalent applications across the Internet. This dataset consists of packet captures and ground truth files from systematic experimentation in a variety of configurable parameters for video streaming, video teleconferencing, and cloud-based document editing applications. Additionally, the dataset files contain multiple iterations of an experiment for statistical rigor and varying topological configurations enabling the exploration of network complexity and encryption.

Future work will entail a more diverse set of network configurations to reflect the heterogeneous nature of interconnected networks and end-user clients. We plan to simulate dynamic network conditions like asymmetric or metered bandwidths with realistic latency and loss (often found in residential and mobile environments), deploy a variety of end-user clients and OSes (mobile devices, Microsoft Windows), and a mix of split-tunneled and full tunneled encrypted connections.

The DARPA SEARCHLIGHT dataset aims to provide a starting point for development and evaluation of data science and AI/ML methods for networked systems and cybersecurity in the climate of today's Internet. Our dataset is freely and publicly accessible at https://mergetb.org/projects/searchlight/.

ACKNOWLEDGMENTS

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). Work by USC/ISI was sponsored by Sandia National Laboratories (SNL) under PO2160586. SNL is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense, Department of Energy, or the U.S. Government.

REFERENCES

[1] 8x8. 2022. Jitsi Meet. https://jitsi.org/meet

[2] Calvin Ardi, Alefiya Hussain, and Stephen Schwab. 2021. Building Reproducible Video Streaming Traffic Generators. In *Cyber Security Experimentation and Test Workshop* (Virtual, CA, USA) *(CSET '21)*. Association for Computing Machinery, New York, NY, USA, 91–95. https://doi.org/10.1145/3474718.3474721

[3] MergeTB Authors. 2022. *The Merge Testbed Platform.* https://next.mergetb.org

[4] Fabrice Bellard. 2005. QEMU, a Fast and Portable Dynamic Translator. In *Proceedings of the Annual Conference on USENIX Annual Technical Conference* (Anaheim, CA) *(ATEC '05)*. USENIX Association, USA, 41.

[5] RIPE Network Coordination Center. 2022. RIPE Atlas. https://www.ripe.net/analyse/internet-measurements

[6] kc claffy, David Clark, John Heidemann, Fabian Bustamante, Mattijs Jonker, Aaron Schulman, and Ellen Zegura. 2021. Workshop on Overcoming Measurement Barriers to Internet Research (WOMBIR 2021) Final Report. *SIGCOMM Comput. Commun. Rev.* 51, 3 (July 2021), 33–40. https://doi.org/10.1145/3477482.3477489

[7] DARPA. 2022. Searchlight. https://www.darpa.mil/program/searchlight

[8] David DeAngelis, Alefiya Hussain, Brian Kocoloski, Calvin Ardi, and Stephen Schwab. 2022. Generating Representative Video Teleconferencing Traffic *(CSET '22)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3546096.3546107

[9] Jason A Donenfeld. 2017. Wireguard: Next Generation Kernel Network Tunnel. In *24th Annual Network and Distributed System Security Symposium* (San Diego, California, USA) *(NDSS '17)*. Internet Society. https://doi.org/10.14722/ndss.2017.23160

[10] Naganand Doraswamy and Dan Harkins. 2003. *IPSec: the new security standard for the Internet, intranets, and virtual private networks.* Prentice Hall Professional.

[11] Constantine Dovrolis, Krishna Gummadi, Aleksandar Kuzmanovic, and Sascha D. Meinrath. 2010. Measurement Lab: Overview and an Invitation to the Research Community. *SIGCOMM Comput. Commun. Rev.* 40, 3 (June 2010), 53–56. https://doi.org/10.1145/1823844.1823853

[12] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, Enric Pujol, Ingmar Poese, Christoph Dietzel, Daniel Wagner, Matthias Wichtlhuber, Juan Tapiador, Narseo Vallina-Rodriguez, Oliver Hohlfeld, and Georgios Smaragdakis. 2020. The Lockdown Effect: Implications of the COVID-19 Pandemic on Internet Traffic. In *Proceedings of the ACM Internet Measurement Conference* (Virtual Event, USA) *(IMC '20)*. Association for Computing Machinery, New York, NY, USA, 1–18. https://doi.org/10.1145/3419394.3423658

[13] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, Enric Pujol, Ingmar Poese, Christoph Dietzel, Daniel Wagner, Matthias Wichtlhuber, Juan Tapiador, Narseo Vallina-Rodriguez, Oliver Hohlfeld, and Georgios Smaragdakis. 2021. A Year in Lockdown: How the Waves of COVID-19 Impact Internet Traffic. *Commun. ACM* 64, 7 (June 2021), 101–108. https://doi.org/10.1145/3465212

[14] The Etherpad Foundation. 2022. Etherpad. https://etherpad.org

[15] Timur Friedman, Phillipa Gill, Sue Moon, Dave Clark, and Ítalo Cunha. 2022. The Networking Channel: Network Datasets: what exists, and what are the problems? https://networkingchannel.eu/network-datasets-what-exists-and-what-are-the-problems/

[16] John Heidemann and Christos Papadopoulos. 2009. Uses and Challenges for Network Datasets. In *Proceedings of the IEEE Cybersecurity Applications and Technologies Conference for Homeland Security (CATCH)*. IEEE, Washington, DC, USA, 73–82. https://doi.org/10.1109/CATCH.2009.29

[17] Alefiya Hussain, Genevieve Bartlett, Yuri Pryadkin, John Heidemann, Christos Papadopoulos, and Joseph Bannister. 2005. Experiences with a Continuous Network Tracing Infrastructure. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Mining Network Data* (Philadelphia, Pennsylvania, USA) *(MineNet '05)*. Association for Computing Machinery, New York, NY, USA, 185–190. https://doi.org/10.1145/1080173.1080181

[18] kc claffy. 2022. CAIDA Datasets. https://www.caida.org/catalog/datasets/completed-datasets/

[19] Alexander D. Kent. 2016. Cyber-Security Data Sources for Dynamic Network Research. In *Dynamic Networks in Cybersecurity*, Niall Adams and Nick Heard (Eds.). Imperial College Press, 37–65. https://doi.org/10.1142/9781786340757_0002

[20] Brian Kocoloski, Alefiya Hussain, Matthew Troglia, Calvin Ardi, Steven Cheng, Dave DeAngelis, Christopher Symonds, Michael Collins, Ryan Goodfellow, and Stephen Schwab. 2021. Case Studies in Experiment Design on a Minimega Based Network Emulation Testbed. In *Cyber Security Experimentation and Test Workshop* (Virtual, CA, USA) *(CSET '21)*. Association for Computing Machinery, New York, NY, USA, 83–90. https://doi.org/10.1145/3474718.3474730

[21] Richard Lippmann, Joshua W Haines, David J Fried, Jonathan Korba, and Kumar Das. 2000. The 1999 DARPA off-line intrusion detection evaluation. *Computer Networks* 34, 4 (2000), 579–595. https://doi.org/10.1016/S1389-1286(00)00139-0

[22] Microsoft. 2022. Playwright. https://playwright.dev

[23] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019). https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[24] Markus Ring, Sarah Wunderlich, Deniz Scheuring, Dieter Landes, and Andreas Hotho. 2019. A survey of network-based intrusion detection data sets. *Computers & Security* 86 (2019), 147–167. https://doi.org/10.1016/j.cose.2019.06.005

[25] Sandvine. 2020. The Global Internet Phenomena Report COVID-19 Spotlight. (7 May 2020). https://www.sandvine.com/phenomena

[26] Sandvine. 2022. 2022 Global Internet Phenomena Report. (20 Jan. 2022). https://www.sandvine.com/phenomena

[27] Jihwang Yeo, David Kotz, and Tristan Henderson. 2006. CRAWDAD: A Community Resource for Archiving Wireless Data at Dartmouth. *SIGCOMM Comput. Commun. Rev.* 36, 2 (April 2006), 21–22. https://doi.org/10.1145/1129582.1129588