Team 175 - Darkspace Traffic Analysis

Christopher Tran, John Nguyen, Kate Thresher, ChuHui Fu, Kenji Hagiwara

1 INTRODUCTION

Malicious online activities like Distributed Denial of Service (DDoS) attacks and the spread of internet worms can cause significant disruptions for users and administrators. Cybersecurity researchers, network administrators, and others interested in preventing malware have long sought effective methods to detect such activities. However, labeling and classifying network traffic data is challenging due to its high dimensionality and large volume, particularly when the traffic originates from evasive sources, such as malware. This problem becomes even more complex when dealing with darkspace internet traffic, which comprises unsolicited, mostly illegitimate network activity.

Network traffic visualization has the potential to greatly assist in the labeling and classification of such data, yet academic literature on this subject is scarce. Even fewer sources focus on visualizing darkspace internet traffic, despite it being an invaluable resource for cybersecurity research due to its illegitimate nature. Our project aims to evaluate the effectiveness of data visualization techniques from big data and machine learning to visualize, cluster, and classify darkspace network traffic.

2 PROBLEM DEFINITION

Unsolicited and harmful network activity cause significant disruptions for users and network administrators. Detecting these types of attacks can be challenging due to the complex nature of network data and because it often originates from sources trying to evade detection. This challenge is even greater with darkspace internet traffic, which consists of unsolicited and mostly harmful network activity.

Our objective is to evaluate dimensionality reduction and clustering techniques for data visualization and data exploration of darkspace network traffic.

3 LITERATURE SURVEY

We organized our literature review around three themes: (1) Darkspace Internet Traffic — What is darkspace internet traffic, and what can we expect to find in it? (2) Data Visualization Techniques for Big Data — What are data visualization techniques relevant for visualization for high dimensional, high volume data? (3) Applying Big Data Visualization Techniques to Network Traffic Data — How can these techniques be applied to darkspace network traffic?

3.1 Darkspace Internet Traffic

Darkspace IP addresses are those with no active host attached, meaning they do not respond to any requests [9]. Despite this, unsolicited data is frequently sent to these addresses making darkspace traffic a valuable source of information for cybersecurity researchers. This data often includes backscatter from DDoS attacks, automated propagation of viruses, or simply port scanners [11]. Darkspace traffic can reveal interesting phenomena, as sudden spikes in this traffic deviating from the typical background noise of the internet often indicate new vulnerabilities, misconfigurations, or large-scale cyberattacks [22].

Previous studies have examined darkspace traffic from different perspectives. For instance, [12] attempted to classify darkspace traffic using clustering techniques, reducing network traffic to two dimensional representations. Similarly, [4] used time-series analysis to detect probing attacks searching for vulnerabilities in darkspace IPs. These studies introduce the time-based aspect of network traffic, showing that connections viewed in isolation may be insignificant, but viewing them over time can reveal meaningful patterns. Our project aims to extend these analyses by producing visualizations that reflect real-world data rather than hypothetical clusters they describe.

3.2 Data Visualization Techniques for Big Data

Techniques like t-SNE [19] and UMAP [14] have gained popularity as methods for dimensionality reduction and data visualization. t-SNE is widely known for its application in visualizing high-dimensional data, despite limitations in scalability and preserving global structures, while UMAP offers a more versatile and scalable alternative. Both techniques have been successfully applied to diverse fields such as genomics [2], geological material identification [1], and cancer research [10], providing critical insights by reducing data complexity.

These papers explore how dimensionality reduction techniques help in uncovering patterns in their respective fields. Challenges faced by these techniques often include balancing the preservation of local versus global data structures, sensitivity to parameter choices, and computational performance, which aligns with greater literature specifically analyzing these data reduction techniques for their limitations and best practices [13], [7], [8], [21].

In our project, these visualization techniques and the literature on how best to use them will be critical in exploring and understanding darkspace traffic. Unlike the pre-labeled datasets used these evaluation, our dataset contains both labeled and unlabeled data, making the task of visualization and clustering more challenging.

3.3 Applying Big Data Visualization Techniques to Network Traffic Data

Several attempts have been made to visualize network traffic. For example, PCAP Funnel [17] and NetCapVis [18] offer graphical interfaces for visualizing network data. They focus on metrics such as active connections or distinguishing between incoming and outgoing traffic which are not relevant to darkspace traffic, where IPs do not respond to requests. Other studies, like that of Ruan et al. [16] have applied clustering algorithms to network traffic but failed to consider the importance of feature selection, leading to suboptimal performance, particularly with t-SNE.

Some research has focused on darkspace traffic, such as [23], which uses time-series analysis to detect anomalies. However, this method limits the scope of the visualizations to identifying specific patterns, like botnets or cyberattacks [6]. Our project aims to build on these efforts by providing more generalized visualization methods for darkspace traffic analysis, enabling the identification of broader trends within the data.

4 PROPOSED METHOD

4.1 Overview

We aim to simplify the process for users to identify network activity in darkspace traffic by providing a visual tool for threat hunting. Our primary goal is to build a dashboard that allows users to explore darkspace data using a variety of visualization techniques, including feature selection, dimensionality reduction, clustering, and coloring. We will be basing our dashboard on a publicly available dataset: the Annotated Anonymized Telescope Packets Sampler dataset [5].

In terms of novel ideas and approaches, our main contributions would be producing a visualization tool specifically designed for darkspace internet traffic while evaluating ideas found in big data visualizations, such as t-SNE and UMAP.

4.2 Intuition

Traditional visualization techniques of network traffic data mostly seek to produce graphs on just two features at once, for example, PCAP Funnel [17] and NetCapVis [18]. As a result, the researcher ends up combing through numerous graphs to try to find interesting network behavior.

t-SNE and UMAP have the potential to display higher dimensional data by reducing the sample to a two-dimensional point. Previous attempts at using t-SNE and UMAP like that of Ruan et al. [16] have not been very successful, as they do not carefully consider feature selection and hyperparameter tuning. We use these shortcomings to guide our research.

4.3 Description of Our Approach

Our methodology consists of several key components: data preprocessing, dimensionality reduction, clustering techniques, and the development of an interactive dashboard for visualization and analysis.

Prior to applying dimensionality reduction and clustering techniques, we performed essential data preprocessing steps to prepare the darkspace network traffic data for analysis.

Feature Selection. We extracted relevant features from the raw network traffic data (PCAPs), including source IP addresses, destination ports, packet length, duration and timestamp.

Standardization. The selected features were standardized using scikit-learn's `StandardScaler`, which transforms the data to have a mean of zero and a standard deviation of one. This standardization process centers the data and scales it to unit variance, ensuring that each feature contributes equally to the analysis.

4.3.1 *Dimensionality Reduction Techniques.*

From the literature review, we determined three key dimensionality reduction methods to try use for visualization: t-SNE and UMAP.

t-SNE (t-distributed Stochastic Neighbor Embedding). t-SNE is a non-linear dimensionality reduction technique intended specifically for data visualization, so it's limited to 2 and 3 components. However, t-SNE is computationally intensive and sensitive to parameter choices, such as perplexity, which is configurable in the dashboard.

UMAP (Uniform Manifold Approximation and Projection). UMAP was developed after t-SNE, providing an alternative that seeks to perform better than t-SNE and to be more generally useful as a dimensionality reduction technique by allowing any number of components. Like t-SNE, it's also sensitive to hyperparameter configuration, such as the number of neighbors and the minimum distance.

4.3.2 *Data and Data Processing.*

The Annotated Anonymized Telescope Packets Sampler dataset [5] contains packet capture data in PCAP format, for all data sent to a /24 darkspace subnet in the University of California San Diego ISP network for the week of August 14-20, 2022. Each .pcap file represents one hour of data, totaling 168 PCAP files representing 24 hours of data per day for 7 days. Each PCAP represents the data sent by a source IP address to a destination IP in the UCSD system. Individual .pcap files can range from 8000 - 50,000 kilobytes of data with up to hundreds of thousands of PCAPs per file.

Every packet capture has an accompanying metadata in JSON format which describe the source IP origin country and a heuristically generated label that attempts to identify the origin of the traffic. Note that majority of the traffic was unlabeled.

An individual packet capture is limited to a single data transfer and thus are not sufficiently feature rich to represent interesting network behavior. We aggregated multiple packet captures into a 15 minute timeslice, which we defined as a collection of aggregate statistics of all packet sent within a a minute window and from the same source IP. Since each timeslice represents a collection of packet captures, we are able to obtain a feature-rich representation of network behaviors. Each timeslice was calculated and then saved into a Parquet dataframe for easier use. Features were chosen based upon the information available in a packet capture with a mix of volume agnostic and volume statistics. This is to enable the feature selector to be able to configure between volume based and volume agnostic clustering.

The .pcap files and their contained packet data is parsed using the Scapy library [3]. Scipy was used to calculate some of the statistics [20]. Pandas and Numpy were used to calculate the timeslice statistics. All other libraries used for processing the data can be found in the Python Standard Library.

Note that this data processing step is incredibly intensive, it was taking over 12 hours on an AMD EPYC 7702 server that has 256 GB of RAM and 64 cores and 128 threads and using all 128 threads. This machine was provided via the SPHERE Testbed [15], a network testbed maintained by the University of Southern California and made available for access for students and researchers.

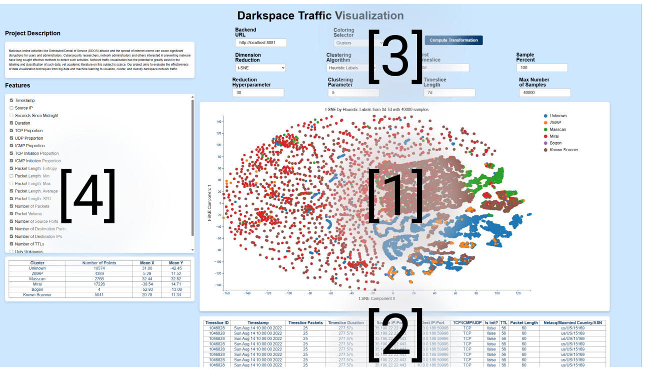
4.3.3 *Interactive Dashboard.*

To facilitate user interaction and make the analytical tools accessible, we developed an interactive dashboard with the features listed below. A screenshot of the website frontend is shown in **Fig. 1**.

This connects to a backend server which loads in the processed data, performs the transformations as requested, and provides data as necessary. As such, the backend server URL is configurable, depending on where you place it.

In the center, marked as 1, we have the primary focus: the graph, which displays the samples according to how the dimensionality reduction technique placed them. When a user highlights samples by dragging the mouse over a region, the packet information associated with the samples are displayed below the graph, marked at 2.

At the top, marked at 3, we have a number of selectors to control the dimensionality reduction technique and its hyperparameter, the clustering technique and its hyperparameter, how to color the samples, and which samples to include and how many samples to use.



Darkspace Traffic Visualization

Project Description

Malicious online activities like Distributed Denial of Service (DDOS) attacks and the spread of internet worms can cause significant disruptions for users and administrators. Cybersecurity researchers, network administrators and others interested in preventing malware have long sought effective methods to detect such activities. Network traffic visualization has the potential to greatly assist in the labeling and classification of such data, yet academic literature on this subject is scarce. Our project aims to evaluate the effectiveness of data visualization techniques from big data and machine learning to visualize, cluster, and classify darkspace network traffic.

Backend URL: http://localhost:8081
Coloring Selector: Clusters
Compute Transformation
Dimension Reduction: t-SNE
Clustering Algorithm: Heuristic Labels
First Timeslice: 0d
Sample Percent: 100
Reduction Hyperparameter: 30
Clustering Parameter: 5
Timeslice Length: 7d
Max Number of Samples: 40000

Features

- ☑ Timestamp
- ☐ Source IP
- ☐ Seconds Since Midnight
- ☑ Duration
- ☑ TCP Proportion
- ☑ UDP Proportion
- ☑ ICMP Proportion
- ☑ TCP Initiation Proportion
- ☑ ICMP Initiation Proportion
- ☑ Packet Length: Entropy
- ☐ Packet Length: Min
- ☐ Packet Length: Max
- ☑ Packet Length: Average
- ☑ Packet Length: STD
- ☑ Number of Packets
- ☑ Packet Volume
- ☑ Number of Source Ports
- ☑ Number of Destination Ports
- ☑ Number of Destination IPs
- ☑ Number of TTLs
- ☐ Only Unknowns

Cluster	Number of Points	Mean X	Mean Y
Unknown	10574	31.00	-42.45
ZMAP	4389	5.29	17.52
Masscan	2766	32.44	32.82
Mirai	17226	-39.54	14.71
Bogon	4	-52.93	-13.08
Known Scanner	5041	20.78	11.34

Timeslice ID	Timestamp	Timeslice Packets	Timeslice Duration	Source IP:Port	Dest IP:Port	TCP/ICMP/UDP	Is Init?	TTL	Packet Length	Netacq/Maxmind Country/ASN
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169
1046828	Sun Aug 14 10:00:00 2022	25	277.57s	35.190.22.22:443	10.0.0.186:59996	TCP	false	56	60	us/US/15169

Figure 1: Dashboard Front-End with Sections Numbered

On the left, marked at 4, we have the feature selection checkboxes, which will determine which features are used for dimensionality reduction and clustering, and a table belong display statistics about the clustering method.

5 EXPERIMENTS/EVALUATION

5.1 Testbed and Questions

Principally, this tool is meant as a visualization tool to help researchers identify darkspace network traffic. To evaluate this, we have 4 criteria that we used to judge the effectiveness of a reduction technique for use in visualization.

5.1.1 Evaluation Criteria.

- **Stability.** If you change the feature selection, the hyperparameter selection, or sample selection, does the shape stay consistent? What happens as you change which features to use?
- **Global Identification.** Can we identify how the data reduction technique clustered the timeslices on a global scale?
- **Local Identification.** Can we identify the network traffic activity inside of a cluster for the data reduction technique? Particularly, can we identify unlabeled traffic?

5.2 Experiments and Observations

5.2.1 UMAP.

Global Identification. Features are kind of mixed around, as you can see in Figure 2, which colors the samples in UMAP via TCP Proportion.

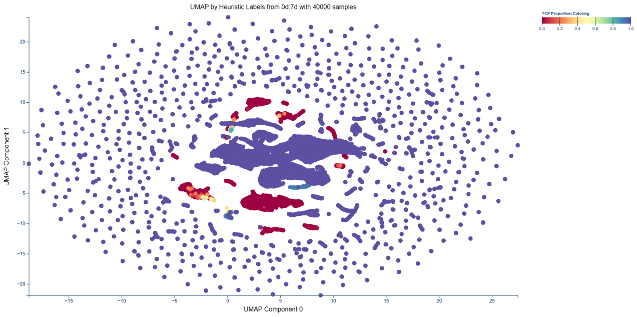


Figure 2: UMAP Colored by TCP

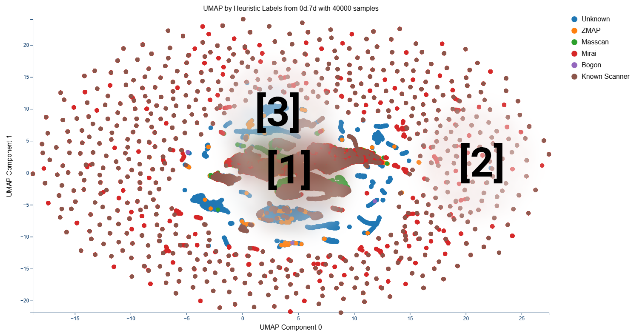


Figure 3: UMAP with Major Clusters Numbered

Local Identification. There are three main areas in Figure 3. Note that the visual size of a cluster does not necessarily correlate with the number of packets involved in that cluster.

The center, marked as major cluster 1, houses the majority of known scanning activity from TCP scans like ZMap, Masscan, Mirai, and Known scanners. Note that even though the graph looks like it's all of the label "Known Scanner," that is a limitation of the resolution, as the "Known Scanner" points are drawn on top of the other labels, obscuring the other colors.

The outside, marked at major cluster 2, has a forest of small timeslices, each usually involving a timeslice with just 1 packet, sent at a similar time. A lot of these involve telnet, which is a well known, very insecure protocol that is often targeted by cybercriminals.

The pre-labeled data mostly lives in the large cluster and the forest of small timeslices, everything after this point were not labeled in the dataset and is something that we were able to identify using the tool.

Between the center and the forest of timeslices, marked at major cluster 3, there's a ring of unknown points. After looking through the packets, the most obvious

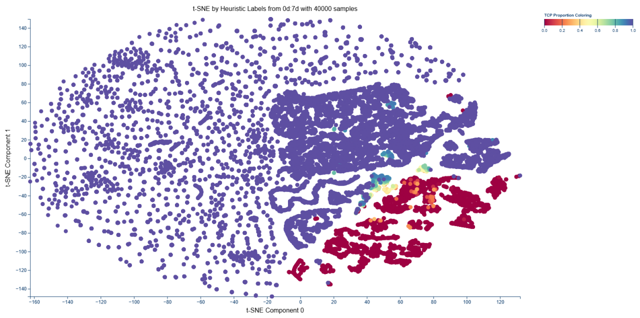


Figure 4: t-SNE Colored by TCP

clusters here are DNS scans, of which there are multiples. There's also a cluster for TCP backscatter. You can also find a cluster of traceroute scans. When identifying these clusters, it was somewhat noisy, as the forest of small points are interleaved with the unknown ring, and the clusters often had extra packets in them, so it was somewhat difficult to do.

Stability. Due to issues with the library implementation of UMAP, modifying the hyperparameters caused the program to crash, so this was not implemented.

In regards to stability, the two most important features are the timestamp and the duration of the packet captures. Outlier tend to be integrated well, concentrating them without causing significant changes of the overall shape. Without the timestamp and duration features, the shape changes and you lose the the forest of small timeslices and the outer ring becomes much noisier, now being interleaved with samples originally in the giant center cluster and samples previously in the forst of small timeslices.

Conclusion. UMAP performs decently, its biggest issue is that on the global scale, points do not have a lot of meaning, so that nearby clusters are not necessarily related to each other.

5.2.2 t-SNE.

Global Identification. Features have clear patterns on how they're placed, although sometimes it's on the local scale instead of the global scale. For example, as you can see in Figure 4, TCP was all globally placed to the left, and UDP was all placed to the right. However, duration and timestamp was more of a clustering thing, where opposite ends of the cluster would have the extreme values and interpolated in between. Outlier features would have a ground zero point and would be disperse outwardly from that point.

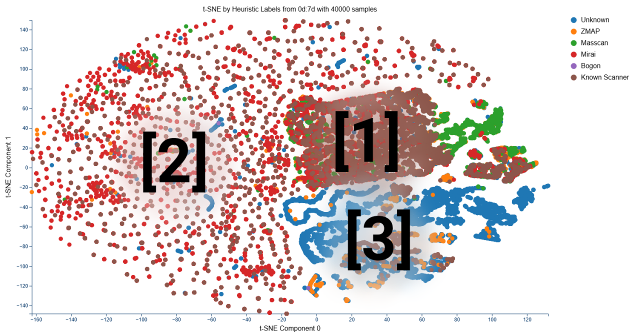


Figure 5: t-SNE with Major Clusters Numbered

Local Identification. There are three main areas, marked in Figured 5. Like UMAP, the visual size of a cluster does not necessarily correlate with the number of packets involved in that cluster, but t-SNE is a little better at this as compared to UMAP. This has mostly the same clusters as UMAP, but they're layed out much clearer.

There's a large cluster marked at 1, houses the majority of known scanning activity from TCP scans like ZMap, Masscan, Mirai, and Known Scanners. This is where most of the prelabeled packets live. Note that even though the graph looks like it's all of the label "Known Scanner," that is a limitation of the resolution, as the "Known Scanner" points are drawn on top of the other labels, obscuring the other colors.

There's the forest of small TCP timeslices marked at 2, usually involving a single TCP packet sent in the timeslice. A lot of these are on port 23, which is a well known insecure protocol frequently targeted by malware authors.

The pre-labeled data mostly lives in the large cluster and the forest of small timeslices, everything after this point were not labeled in the dataset and is something that we were able to identify using the tool. From here, we're going to talk about the minor clusters in marked at major cluster 3.

There's the cluster of TCP backscatter, the ICMP cluster, the traceroute point, the point of someone trying to scan every IP address in the /24 space.

There are several DNS scanning clusters, all close together. As you go further south, there's less activity within the timeslices, with the bottom ones having only 1 timeslice.

There's also some UDP clusters which have some DNS scans, but the rest of it being on random-ish looking ports, which would be interesting to further identify.

Many of these clusters are also found under UMAP, but it was generally easier to identify clusters under t-SNE, as the third major cluster is its own section instead of being a ring around the center.

Stability. At very low perplexity values, e.g. five, we observed a "forest of small dots" effect in the t-SNE visualization. While the general location of samples is identical, there are many more clusters, each of them much smaller, making it hard to use.

The default perplexity of thirty created a more meaningful visualization, forming distinct outlier clusters. Higher levels of complexity, e.g. fifty, eighty, and above, did not significantly alter structure, however, fewer clusters were produced, having absorbed more samples. Overall though, the general location of samples was similar across low and high perplexity values.

Across feature selection, the most significant feature is the timestamp, without it, the clusters tend to "unravel" into lines that are hard to read, significantly changing the placement of samples. Without duration, the graph's clusters are smaller and more diffuse, but the general shape remains. Outlier tend to be integrated well, concentrating them without causing significant changes of the overall shape.

Conclusion. t-SNE performs really well on almost all metrics, its main downside is that it runs slower than both UMAP. However, its clarity, both on the local scale and the global scale, along with its stability makes it worthwhile to use.

6 CONCLUSIONS AND DISCUSSION

We implemented a dashboard to visualize darkspace network traffic data. The dashboard allows the user to select which features to use, a variety of dimensionality techniques to use, a variety of clustering algorithms to apply, and a selection on which samples and how many of them to use, and how to color the samples based upon a feature.

Under both UMAP and t-SNE, but especially t-SNE, we were able to cluster and identify unlabeled network traffic behavior within those clusters.

The main limitation of the work is scale, as the dashboard does not support more than 40k samples due to browser memory limitations, and processing the data is incredibly computation expensive. Future work includes: more efficient graph rendering, more efficient data processing, alternative datasets, alternative features, and more in-depth evaluations of network traffic behavior within clusters.

The most interesting future work would be slicing by time and destination port instead of time and source IP address, as it could potentially easier to identify network behavior as destination port is usually more information than source IP.

6.0.1 Team Effort Distribution.

Listed in order of contributions.

Christopher Tran. Project idea formation, the literature review, project planning and management, report writing, determined features to use for reduction, setting up shared computational resources, the majority of the work on dashboard and backend server development, and majority of the work on network traffic identification within clusters in the visualizations.

John Ngyuen. Literature review, report writing, contributed to feature selection, did all of the data processing, and dashboard development.

Kate Thresher. Literature review, report writing, initial exploration of dimensionality reduction techniques and clustering, and dashboard and backend server development.

ChuHui Fu. Report writing, project planning, UI design, documentation, the initial exploration of dimensionality reduction techniques and clustering.

Kenji Hagiwara. Report writing, literature review, project planning, the initial exploration of dimensionality reduction techniques and clustering, and the evaluation of the stability of clusters.

REFERENCES

[1] Mehala Balamurali, Katherine L. Silversides, and Arman Melkumyan. 2019. A comparison of t-SNE, SOM and SPADE for identifying material type domains in geological data. *Computers Geosciences* 125 (2019), 78–89. https://doi.org/10.1016/j.cageo.2019.01.011

[2] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel W. H. Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W. Newell. 2019. Dimensionality reduction for

visualizing single-cell data using UMAP. *Nature Biotechnology* 37, 1 (01 Jan 2019), 38–44. https://doi.org/10.1038/nbt.4314

[3] Philippe Biondi. 2008 - 2024. Scapy. http://gts.sourceforge.net/.

[4] Elias Bou-Harb, Mourad Debbabi, and Chadi Assi. 2015. A Time Series Approach for Inferring Orchestrated Probing Campaigns by Analyzing Darknet Traffic. In *2015 10th International Conference on Availability, Reliability and Security*. 180–185. https://doi.org/10.1109/ARES.2015.9

[5] UC San Diego CAIDA. 2022. Annotated Anonymized Telescope Packets Sampler. https://catalog.caida.org/dataset/annotated_anonymized_telescope_packets_sampler. Dates used: Sep-Dec 2024. Accessed: Sep 2024.

[6] Alberto Dainotti, Alistair King, Kimberly Claffy, Ferdinando Papale, and Antonio Pescapé. 2015. Analysis of a "/0" Stealth Scan From a Botnet. *IEEE/ACM Transactions on Networking* 23, 2 (2015), 341–354. https://doi.org/10.1109/TNET.2013.2297678

[7] Alex Diaz-Papkovich, Luke Anderson-Trocmé, and Simon Gravel. 2021. A review of UMAP in population genetics. *Journal of Human Genetics* 66, 1 (01 Jan 2021), 85–91. https://doi.org/10.1038/s10038-020-00851-4

[8] Haiyang Huang, Yingfan Wang, Cynthia Rudin, and Edward P. Browne. 2022. Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization. *Communications Biology* 5, 1 (19 Jul 2022), 719. https://doi.org/10.1038/s42003-022-03628-x

[9] Félix Iglesias and Tanja Zseby. 2014. Modelling IP darkspace traffic by means of clustering techniques. In *2014 IEEE Conference on Communications and Network Security*. 166–174. https://doi.org/10.1109/CNS.2014.6997483

[10] Andrew R. Jamieson, Maryellen L. Giger, Karen Drukker, Hui Li, Yading Yuan, and Neha Bhooshan. 2010. Exploring nonlinear feature space dimension reduction and data representation in breast CADx with Laplacian eigenmaps and -SNE. *Medical Physics* 37, 1 (2010), 339–351. https://doi.org/10.1118/1.3267037

[11] Jeremy Kepner, Michael Jones, Daniel Andersen, Aydın Buluç, Chansup Byun, K Claffy, Timothy Davis, William Arcand, Jonathan Bernays, David Bestor, William Bergeron, Vijay Gadepally, Micheal Houle, Matthew Hubbell, Anna Klein, Chad Meiners, Lauren Milechin, Julie Mullen, Sandeep Pisharody, Andrew Prout, Albert Reuther, Antonio Rosa, Siddharth Samsi, Doug Stetson, Adam Tse, Charles Yee, and Peter Michaleas. 2021. Spatial Temporal Analysis of 40,000,000,000,000 Internet Darkspace Packets. In *2021 IEEE High Performance Extreme Computing Conference (HPEC)*. 1–8. https://doi.org/10.1109/HPEC49654.2021.9622790

[12] Sanjay Kumar, Harald Vranken, Joost van Dijk, and Timo Hamalainen. 2019. Deep in the Dark: A Novel Threat Detection System using Darknet Traffic. In *2019 IEEE International Conference on Big Data (Big Data)*. 4273–4279. https://doi.org/10.1109/BigData47090.2019.9006374

[13] Vivien Marx. 2024. Seeing data as t-SNE and UMAP do. *Nature Methods* 21, 6 (01 Jun 2024), 930–933. https://doi.org/10.1038/s41592-024-02301-x

[14] Leland McInnes, John Healy, and James Melville. 2020. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426 [stat.ML] https://arxiv.org/abs/1802.03426

[15] University of Southern California. 2023-2024. SPHERE Testbed. https://sphere-project.net/.

[16] Zichan Ruan, Yuantian Miao, Lei Pan, Yang Xiang, and Jun Zhang. 2018. Big network traffic data visualization. *Multimedia Tools and Applications* 77, 9 (May 2018), 11459–11487. https://doi.org/10.1007/s11042-017-5495-y

[17] Juraj Uhlár, Martin Holkovič, and Vít Rusňák. 2021. PCAP-Funnel: A Tool for Rapid Exploration of Packet Capture Files. In *2021 25th International Conference Information Visualisation (IV)*. 69–76. https://doi.org/10.1109/IV53921.2021.00021

[18] Alex Ulmer, David Sessler, and Jörn Kohlhammer. 2019. NetCapVis: Web-based Progressive Visual Analytics for Network Packet Captures. In *2019 IEEE Symposium on Visualization for Cyber Security (VizSec)*. 1–10. https://doi.org/10.1109/VizSec48167.2019.9161633

[19] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, 86 (2008), 2579–2605. http://jmlr.org/papers/v9/vandermaaten08a.html

[20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17 (2020), 261–272. https://doi.org/10.1038/s41592-019-0686-2

[21] Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. 2020. Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization. *J. Mach. Learn. Res.* 22 (2020), 201:1–201:73. https://api.semanticscholar.org/CorpusID:227745109

[22] Tanja Zseby, Nevil Brownlee, Alistair King, and kc claffy. 2014. Nightlights: Entropy-Based Metrics for Classifying Darkspace Traffic Patterns. In *Passive and Active Measurement*, Michalis Faloutsos and Aleksandar Kuzmanovic (Eds.). Springer International Publishing, Cham, 275–277.

[23] Tanja Zseby, Félix Iglesias Vázquez, Alistair King, and K. C. Claffy. 2016. Teaching Network Security With IP Darkspace Data. *IEEE Transactions on Education* 59, 1 (2016), 1–7. https://doi.org/10.1109/TE.2015.2417512