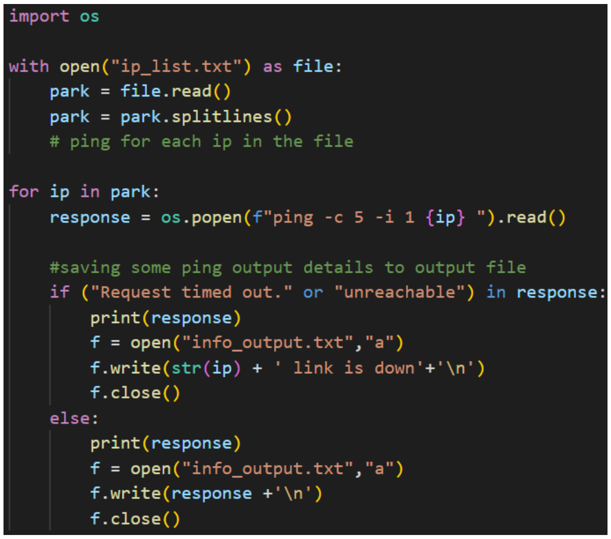
# The Effects of Website Popularity and Time of Day on Delay

## 1. Introduction

For this project, I intend to study websites of varying popularity at different times of the day to study the effects of these variables on delay. My goal is to better understand packet sending and receiving, and what kinds of factors affect the time it takes for this process. If it is found that there is a significant difference in delay based on website popularity or time of day, these sites can make adjustments to better handle the traffic they receive overall, or to perhaps implement a strategy for their most popular times of day. In addition to this, the ping tool I will be using also measures packet loss. If some websites have significant packet loss, either all the time or based on time of day, this can mean that there is something in the site causing a problem that the site should fix.

## 2. Procedure

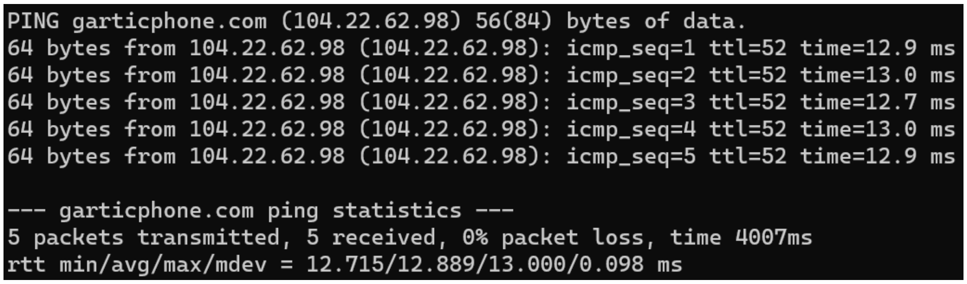
I chose to implement the ping tool for this study, which sends ICMP echo requests to a remote host and reports the ICMP echo responses. Access to the ping tool was done using the eecslab remote server. To run these ping commands, I wrote a python script that reads from a list of domain names, ip_list.txt, and runs a ping request for each. Since ping also allows the user to choose how many requests are sent, I chose to set that 5 so the program sends 5 requests to each host. I left the interval to the standard 1 second, so the program sends 5 requests in 1 second to one host before moving on to the next host. Then, the program also writes all outputs to the ping command to a file named info_output.txt.



```
import os

with open("ip_list.txt") as file:
    park = file.read()
    park = park.splitlines()
    # ping for each ip in the file

for ip in park:
    response = os.popen(f"ping -c 5 -i 1 {ip} ").read()

    #saving some ping output details to output file
    if ("Request timed out." or "unreachable") in response:
        print(response)
        f = open("info_output.txt","a")
        f.write(str(ip) + ' link is down'+'\n')
        f.close()
    else:
        print(response)
        f = open("info_output.txt","a")
        f.write(response +'\n')
        f.close()
```

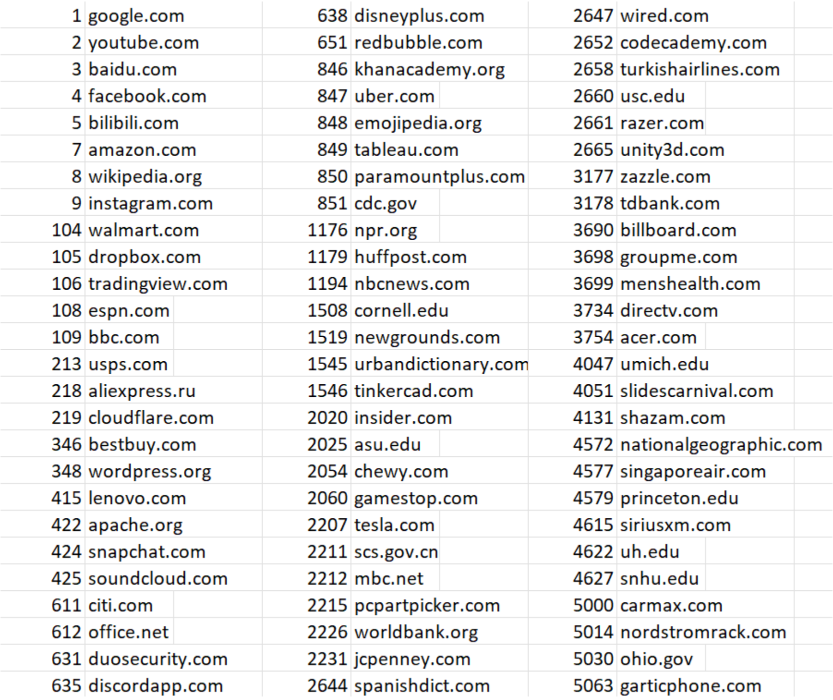
Python script for using ping on list of domain names, named ip_ping

The output files contain all important information needed for this study. For each domain, the output includes the time for each individual request, the minimum, average, max, and mdev values of those 5 requests, and the packets sent vs received vs lost.



```
PING garticphone.com (104.22.62.98) 56(84) bytes of data.
64 bytes from 104.22.62.98 (104.22.62.98): icmp_seq=1 ttl=52 time=12.9 ms
64 bytes from 104.22.62.98 (104.22.62.98): icmp_seq=2 ttl=52 time=13.0 ms
64 bytes from 104.22.62.98 (104.22.62.98): icmp_seq=3 ttl=52 time=12.7 ms
64 bytes from 104.22.62.98 (104.22.62.98): icmp_seq=4 ttl=52 time=13.0 ms
64 bytes from 104.22.62.98 (104.22.62.98): icmp_seq=5 ttl=52 time=12.9 ms

--- garticphone.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 12.715/12.889/13.000/0.098 ms
```

Example response for an ICMP echo request sent to a domain



| | | | | | |
|---|---|---|---|---|---|
| 1 | google.com | 638 | disneyplus.com | 2647 | wired.com |
| 2 | youtube.com | 651 | redbubble.com | 2652 | codecademy.com |
| 3 | baidu.com | 846 | khanacademy.org | 2658 | turkishairlines.com |
| 4 | facebook.com | 847 | uber.com | 2660 | usc.edu |
| 5 | bilibili.com | 848 | emojipedia.org | 2661 | razer.com |
| 7 | amazon.com | 849 | tableau.com | 2665 | unity3d.com |
| 8 | wikipedia.org | 850 | paramountplus.com | 3177 | zazzle.com |
| 9 | instagram.com | 851 | cdc.gov | 3178 | tdbank.com |
| 104 | walmart.com | 1176 | npr.org | 3690 | billboard.com |
| 105 | dropbox.com | 1179 | huffpost.com | 3698 | groupme.com |
| 106 | tradingview.com | 1194 | nbcnews.com | 3699 | menshealth.com |
| 108 | espn.com | 1508 | cornell.edu | 3734 | directv.com |
| 109 | bbc.com | 1519 | newgrounds.com | 3754 | acer.com |
| 213 | usps.com | 1545 | urbandictionary.com | 4047 | umich.edu |
| 218 | aliexpress.ru | 1546 | tinkercad.com | 4051 | slidescarnival.com |
| 219 | cloudflare.com | 2020 | insider.com | 4131 | shazam.com |
| 346 | bestbuy.com | 2025 | asu.edu | 4572 | nationalgeographic.com |
| 348 | wordpress.org | 2054 | chewy.com | 4577 | singaporeair.com |
| 415 | lenovo.com | 2060 | gamestop.com | 4579 | princeton.edu |
| 422 | apache.org | 2207 | tesla.com | 4615 | siriusxm.com |
| 424 | snapchat.com | 2211 | scs.gov.cn | 4622 | uh.edu |
| 425 | soundcloud.com | 2212 | mbc.net | 4627 | snhu.edu |
| 611 | citi.com | 2215 | pcpartpicker.com | 5000 | carmax.com |
| 612 | office.net | 2226 | worldbank.org | 5014 | nordstromrack.com |
| 631 | duosecurity.com | 2231 | jcpenney.com | 5030 | ohio.gov |
| 635 | discordapp.com | 2644 | spanishdict.com | 5063 | garticphone.com |

Domain names in ip_list.txt, with the left numbers representing their popularity from the list of top 1M websites

The steps for gathering data are as follows

(1) Run Python script in the remote server using “python3 ip_ping.py”
Run at the beginning of every hour from 2pm to 10pm for a total of 9 observation times

(2) When program is finished running through all domains in ip_list.txt, rename the output file based on the time of day
Ex. info_output_3pm.txt

(3) Move remote file to local directory under the data folder to collect data files in one place

(4) Collect average delay values from response packets for each domain

Analyzing the samples of ICMP echo requests was made possible by ping returning the average of the response time of the requests. These averages were collected to be used for final conclusions.

## 3. Results

*Effect of time of day on delay*

For the top 5 most popular websites, they stayed mainly consistent throughout the day. See Appendix A for the graphs of this information. However as we move towards looking at domains that are less popular, we can see more variety in delay, with one or many spikes. See Appendix C for some examples of domains with one spike throughout the day, and see Appendix B for domains with interesting data, meaning with significant variation throughout the day. The websites with only one spike in data is generally due to outliers in the data. For some hours of the day, certain sites had 1-2 of the delay measurements for a packet at a much higher value than the others, skewing that site's hourly average. Sites with significant variation like those shown in Appendix B, however, are not due to outliers. In these cases, the 5 packets for that hour were all at a differing value compared to other hours, meaning that this is a trend for this site rather than an outlier. It is also important to note that these spikes were at varying times for different sites, which means that they should be based on the websites, and not some factor on my end. This demonstrates how sites may be struggling to handle requests at certain times of the day, possibly due to a large increase in traffic at these times.

*Effect of website popularity on delay*

To look at the correlation between website popularity and delay, I looked at the same data as above, but generated graphs of the delay values for each website, ranging from higher to lower popularity, for each hour of the day. These graphs can be seen in Appendix D. These graphs show interesting trends, however the highest peaks seem to be for sites not based in the United States, which may have caused significantly higher delay. For that reason, I generated the graphs in Appendix E, which show the same data as Appendix D but without international sites including, bilibili.com, baidu.com, aliexpress.ru, turkishairlines.com, and singaporeair.com. These charts show fewer peaks, with each peak being less significant than those in Appendix D's. From the data I gathered, delay across websites of different popularity at one time does not seem to vary too greatly. There are a few websites of lower popularity that have higher delay than sites of higher popularity, however there are also sites of even lower popularity that have lower delay.

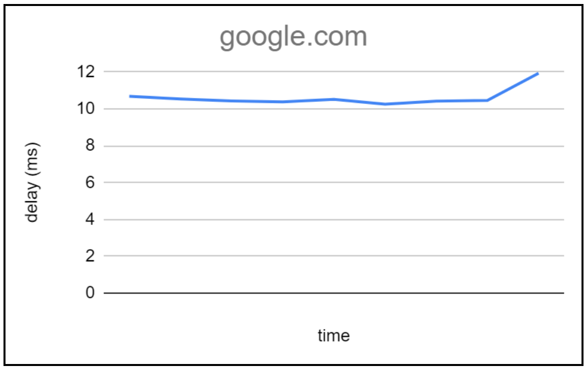
*Observations on Packet Loss*

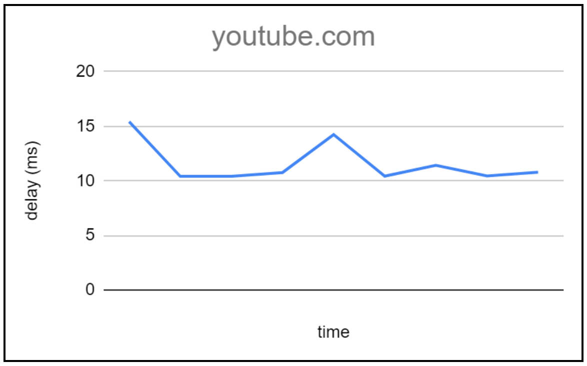
There were a fair number of websites that had 100% packet loss, with a significant range in their associated popularity. That being said, there was a general trend that a greater number of the sites with 100% packet loss were seen with sites with lower popularity. This could mean that their sites can use improvement in request handling, or it also could mean that there is some problem in their software giving the site problems when accepting data.

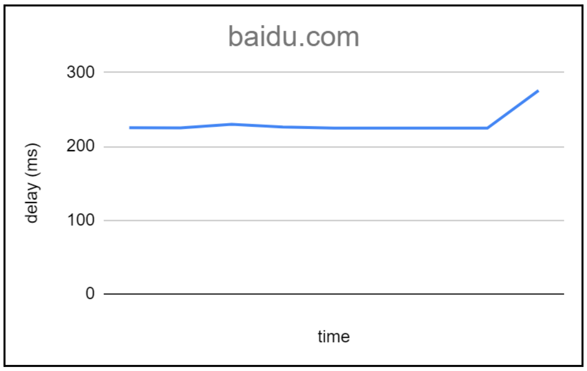
*Overall*

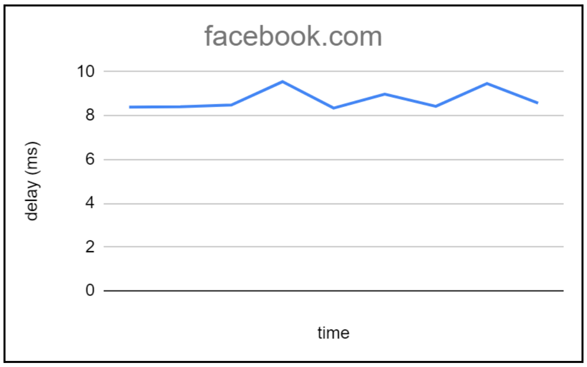
Based on the data my script gathered using the ping tool, it seems that there is a significant correlation within an individual site for delay vs time of day. This is likely due to the site having an increase in the amount of traffic at those peak delay times. This means that the website should take a look at their highest levels of traffic and adjust their request management to be able to better handle those times. Looking at website popularity vs delay, there seems to be only a slight correlation between those factors, with higher delay being found in sites of lower popularity. That being said, this correlation is weak and only seen in a few sites. Lastly, there is a slight correlation between packet loss and website popularity, with sites of lower popularity being more likely to have packet loss.

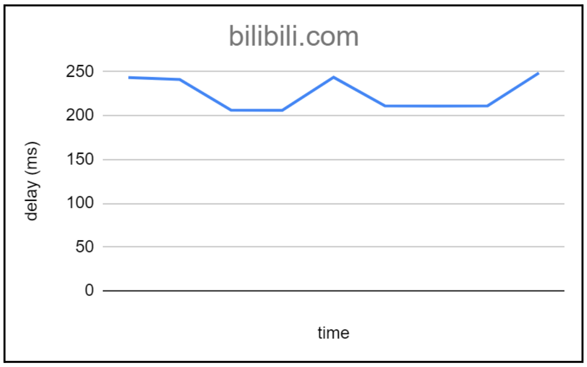
## Appendix A. Graph of top 5 websites, delay (ms) vs time



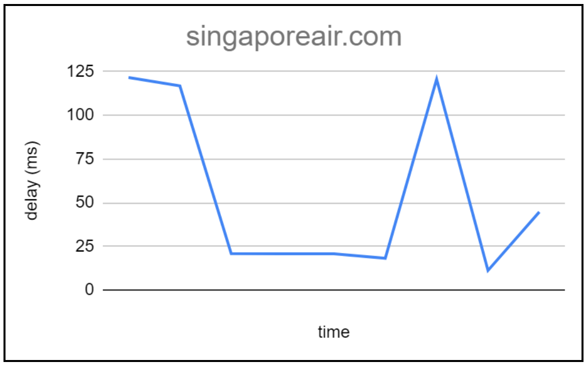


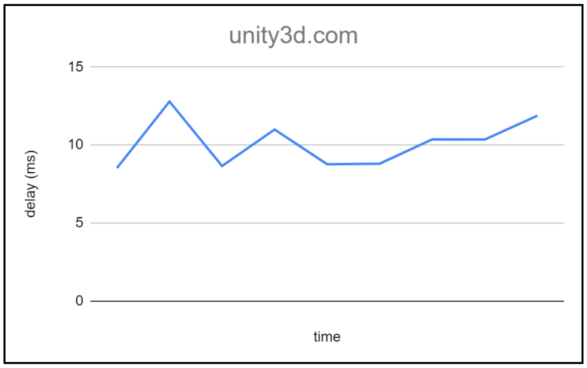


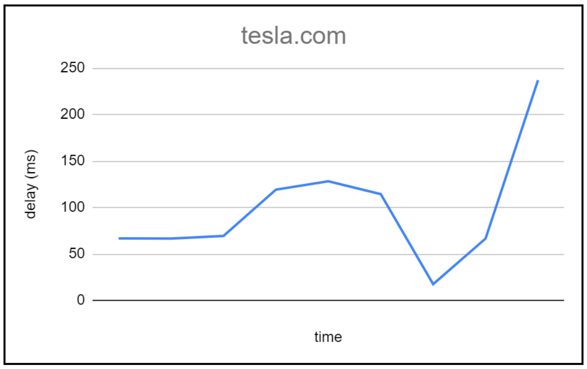


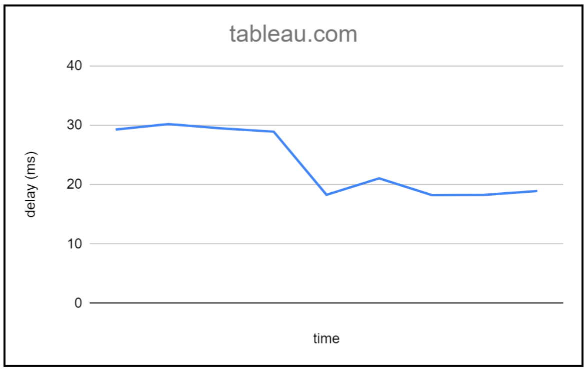


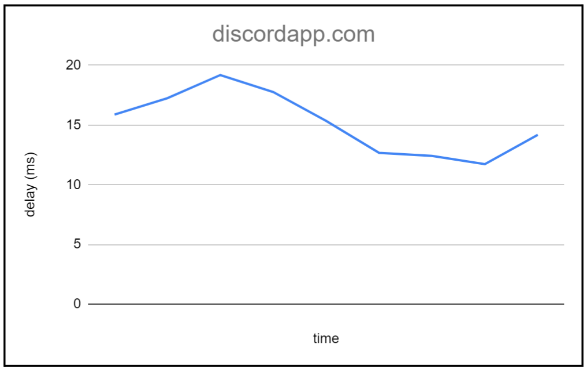
## Appendix B. Graphs of domains with interesting data

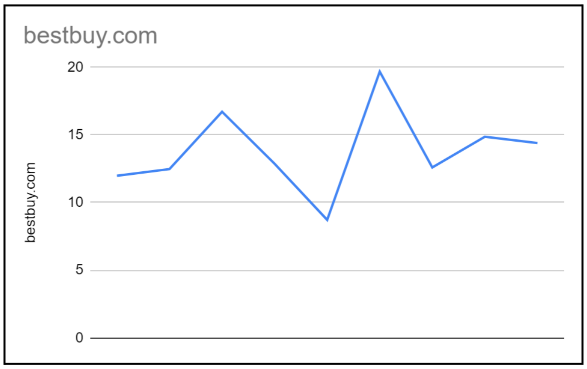




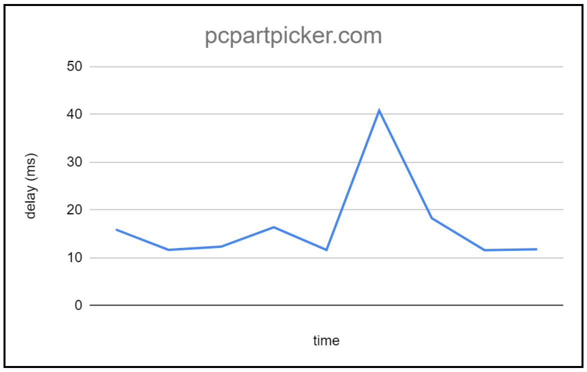


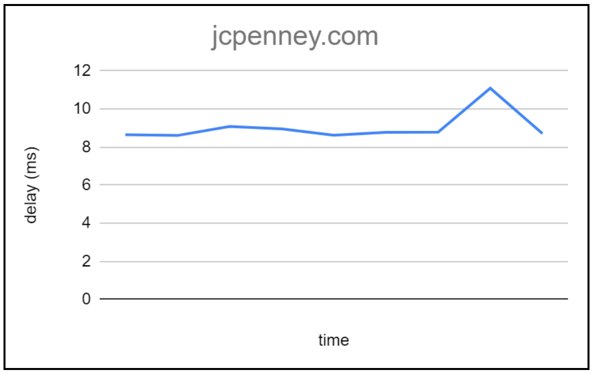


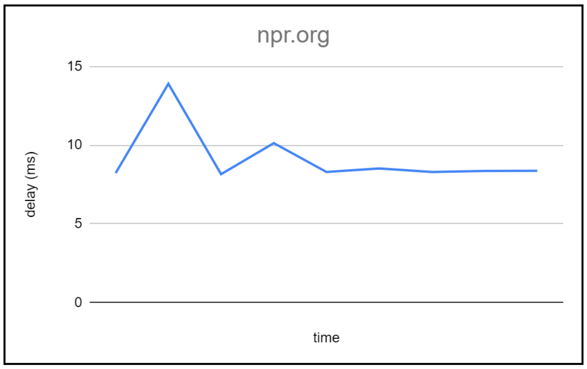


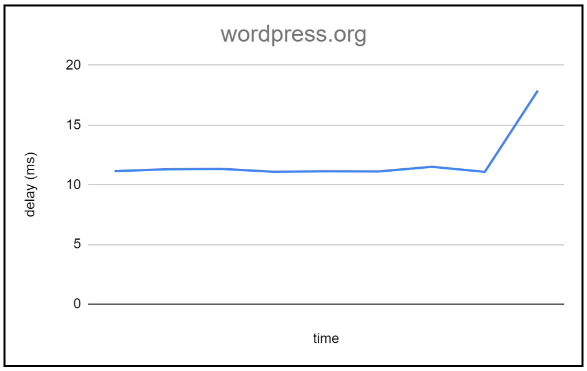


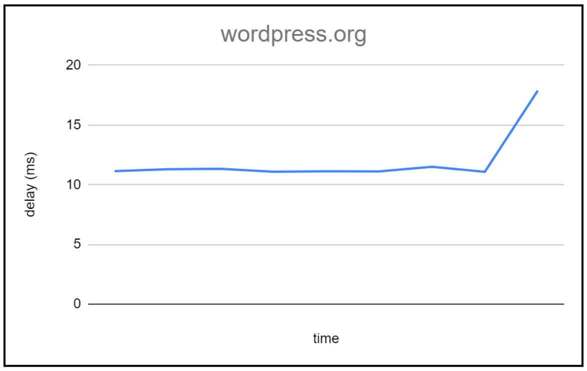
## Appendix C. Graphs with a single spike

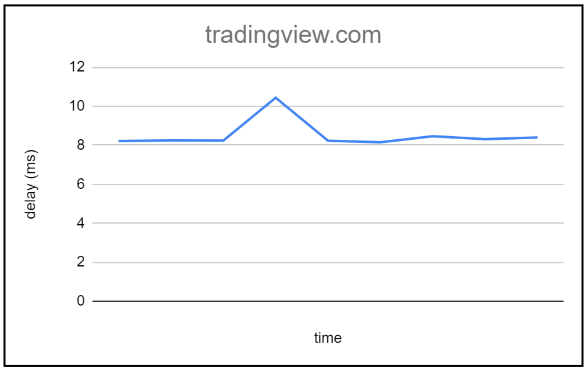




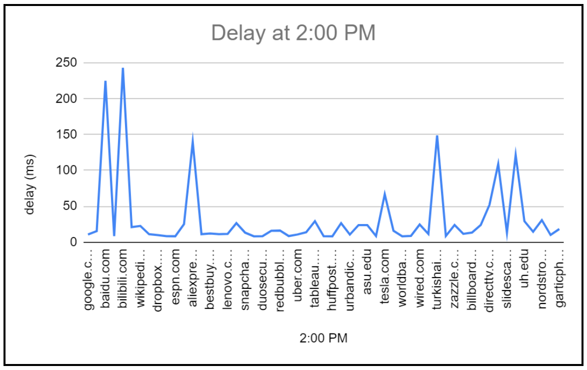


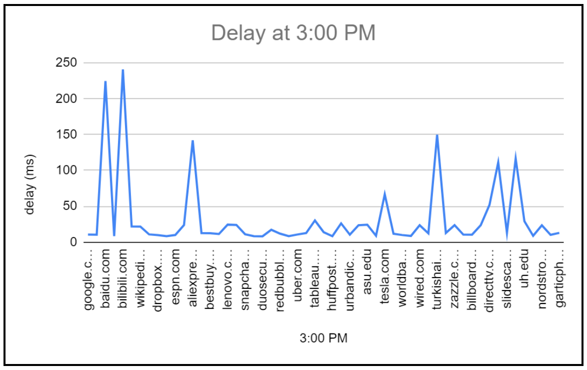


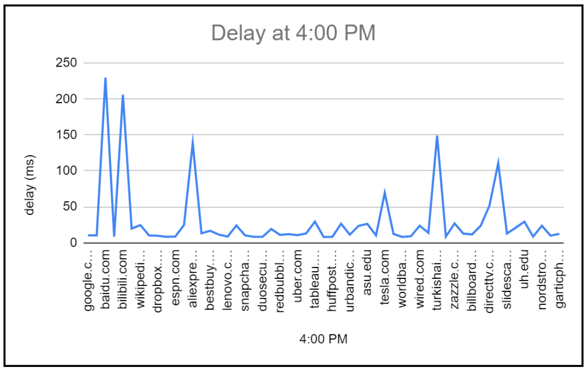


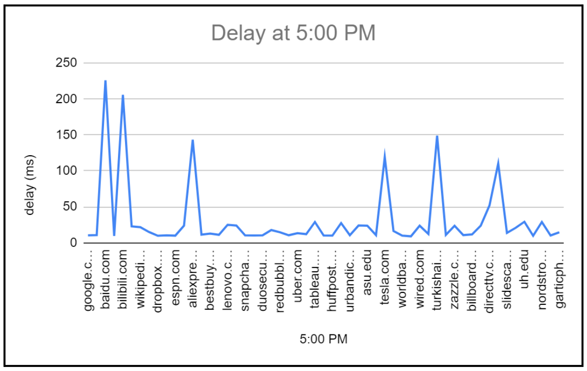


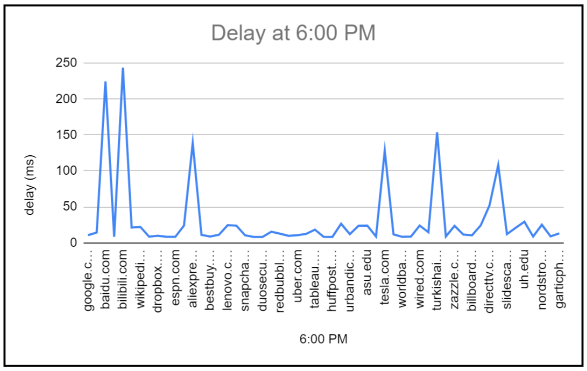
## Appendix D. Delay of Domains at Each Hour of the Day

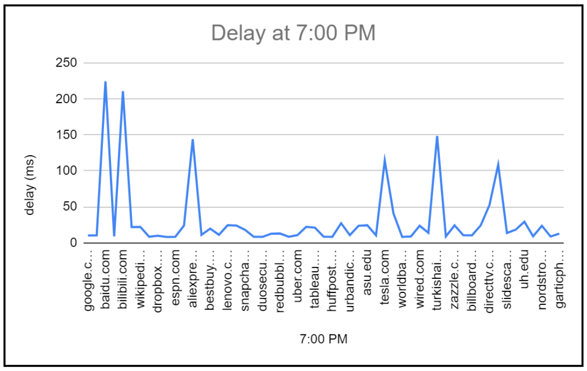

Delay at 2:00 PM

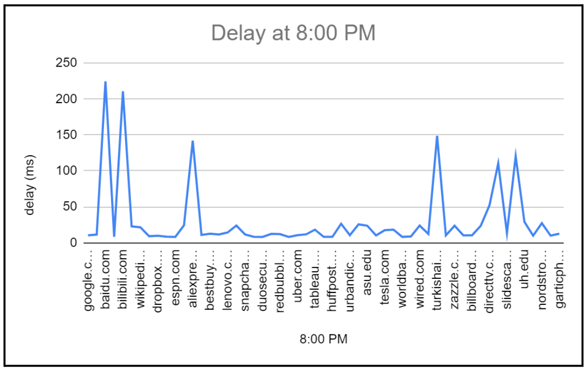

Delay at 3:00 PM

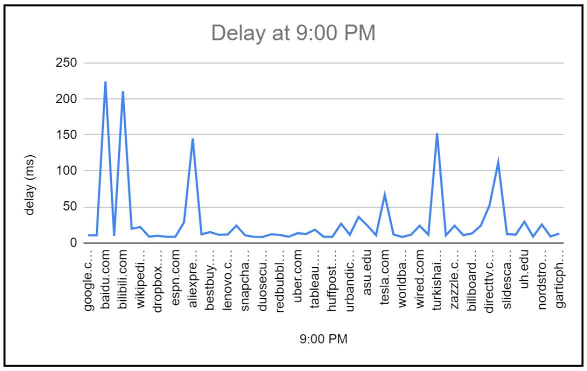

Delay at 4:00 PM

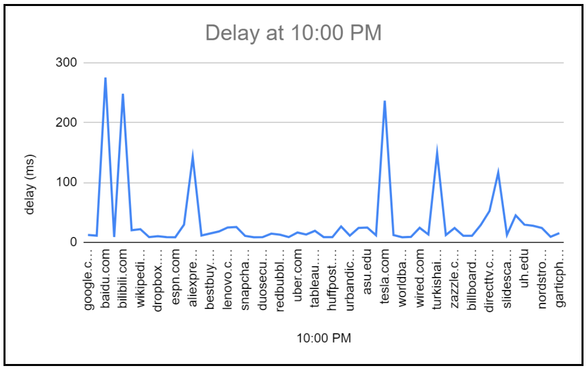

Delay at 5:00 PM


Delay at 6:00 PM

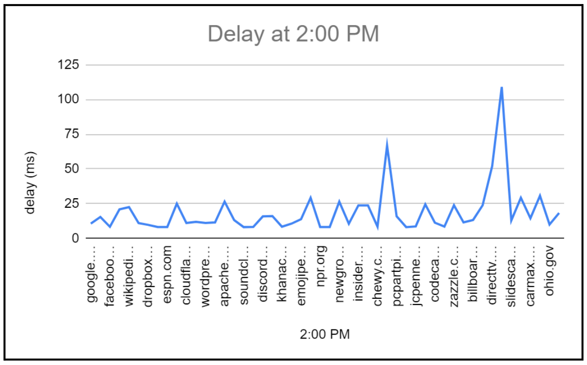

Delay at 7:00 PM

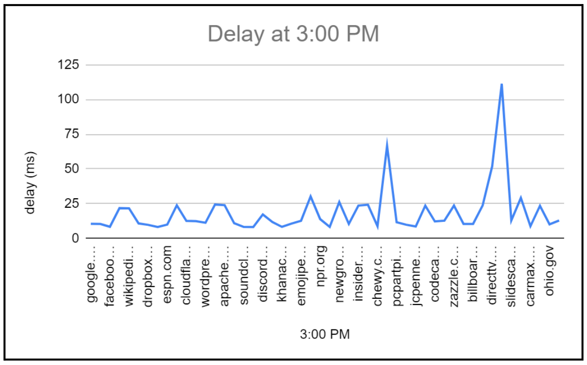

Delay at 8:00 PM

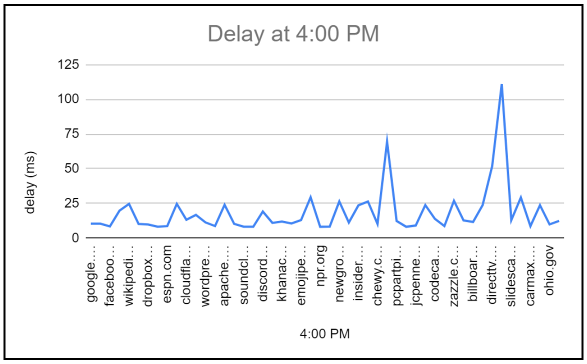

Delay at 9:00 PM

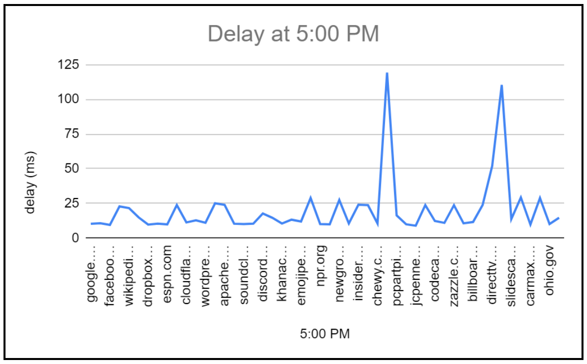


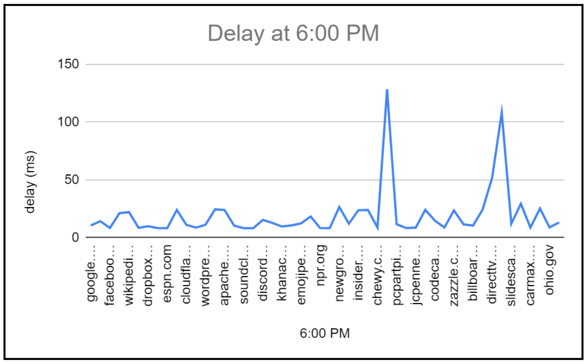
## Appendix E. Delay of Websites at Each Hour of the Day, Omitting International Sites

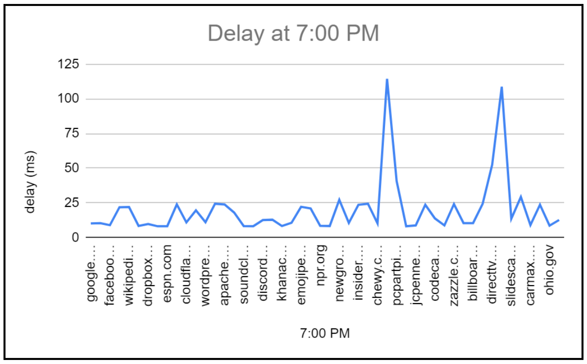


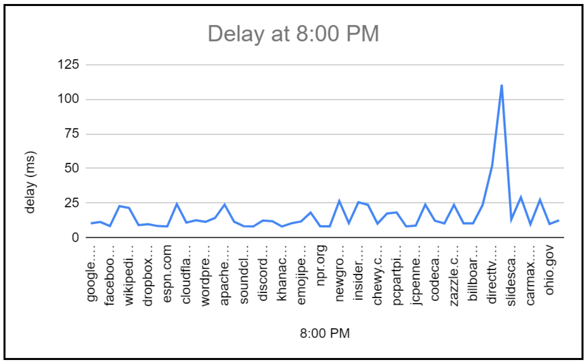


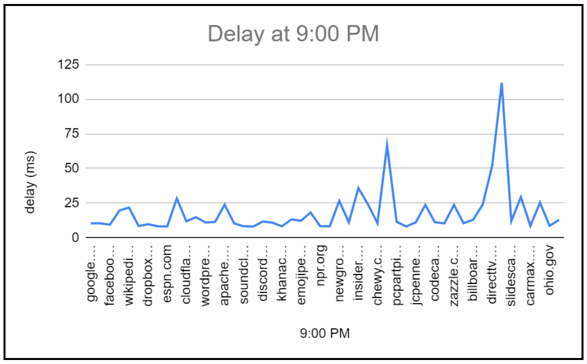


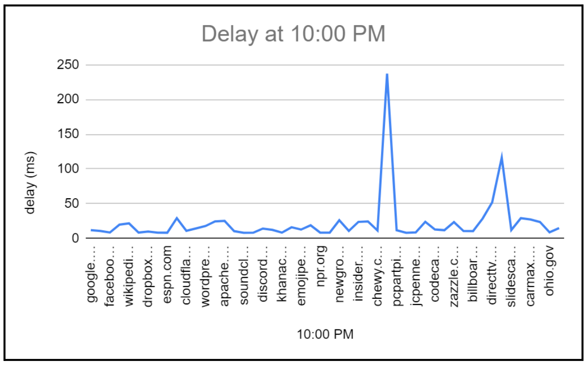







Delay at 8:00 PM
delay (ms)
125
100
75
50
25
0
google.…
faceboo…
wikipedi…
dropbox…
espn.com
cloudfla…
wordpre…
apache.…
soundcl…
discord…
khanac…
emojipe…
npr.org
newgro…
insider.…
chewy.c…
pcpartpi…
jcpenne…
codeca…
zazzle.c…
billboar…
directv.…
slidesca…
carmax.…
ohio.gov
8:00 PM


Delay at 9:00 PM
delay (ms)
125
100
75
50
25
0
google.…
faceboo…
wikipedi…
dropbox…
espn.com
cloudfla…
wordpre…
apache.…
soundcl…
discord…
khanac…
emojipe…
npr.org
newgro…
insider.…
chewy.c…
pcpartpi…
jcpenne…
codeca…
zazzle.c…
billboar…
directv.…
slidesca…
carmax.…
ohio.gov
9:00 PM


Delay at 10:00 PM
delay (ms)
250
200
150
100
50
0
google.…
faceboo…
wikipedi…
dropbox…
espn.com
cloudfla…
wordpre…
apache.…
soundcl…
discord…
khanac…
emojipe…
npr.org
newgro…
insider.…
chewy.c…
pcpartpi…
jcpenne…
codeca…
zazzle.c…
billboar…
directv.…
slidesca…
carmax.…
ohio.gov
10:00 PM