

Experiment Results (All Values in ms):

8.8.8.8 Server

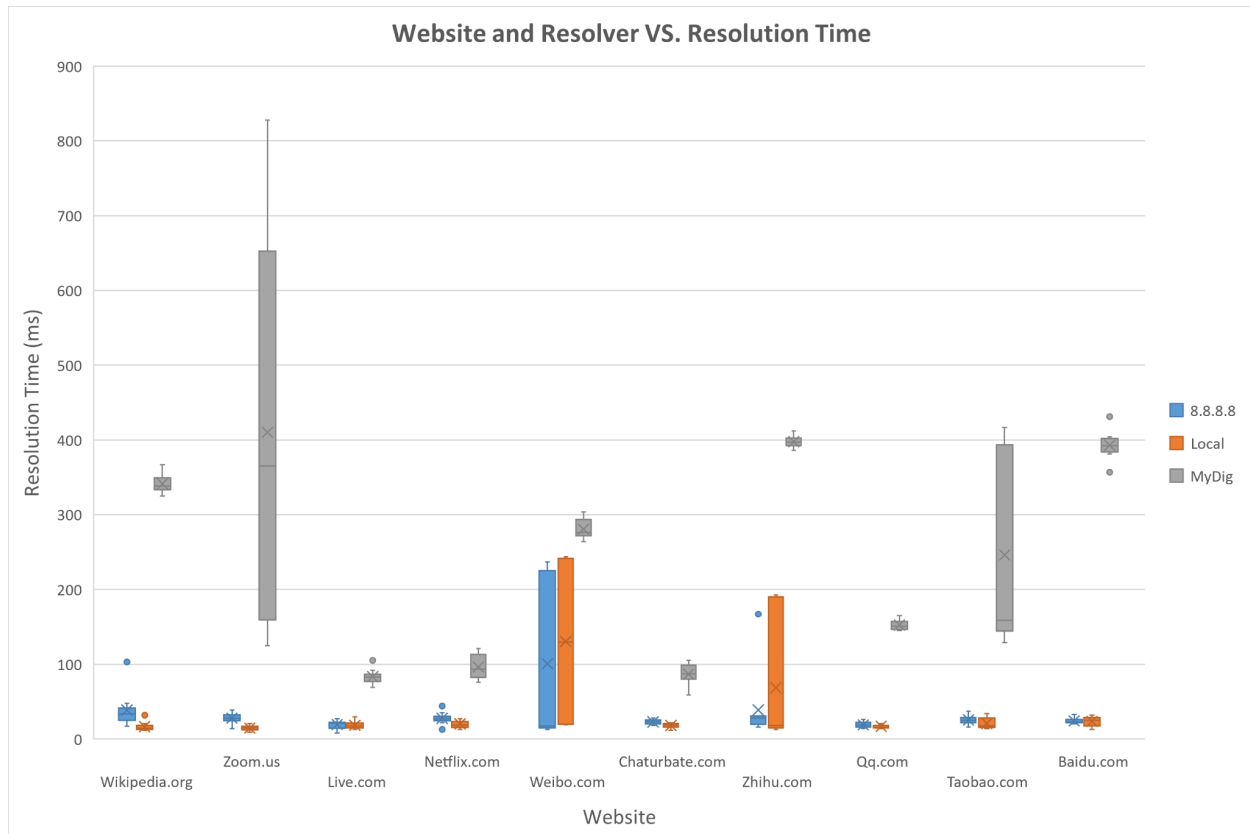
[27, 39, 17, 19, 30, 48, 103, 38, 30, 37] Wikipedia.org
[37, 30, 26, 29, 31, 27, 28, 27, 20, 14] Zoom.us
[13, 21, 8, 27, 21, 22, 22, 24, 15, 22] Live.com
[44, 22, 26, 25, 13, 29, 28, 26, 35, 28] Netflix.com
[24, 31, 29, 20, 25, 19, 251, 24, 28, 34] 360.cn
[28, 23, 27, 20, 21, 25, 18, 23, 24, 20] Chaturbate.com
[18, 167, 16, 29, 21, 31, 31, 28, 28, 20] Zhihu.com
[18, 15, 26, 19, 16, 19, 16, 14, 23, 22] Qq.com
[22, 24, 21, 27, 37, 22, 26, 26, 34, 16] Taobao.com
[23, 21, 33, 27, 20, 25, 23, 23, 22, 25] Baidu.com

Local Server

[32, 12, 15, 14, 12, 13, 17, 15, 14, 21] Wikipedia.org
[21, 13, 16, 14, 10, 17, 9, 16, 14, 17] Zoom.us
[19, 17, 18, 22, 30, 15, 21, 15, 15, 13] Live.com
[22, 14, 27, 20, 18, 16, 17, 19, 37, 13] Netflix.com
[83, 84, 81, 87, 7, 83, 24, 25, 94, 21] 360.cn
[21, 22, 16, 15, 16, 20, 18, 12, 21, 19] Chaturbate.com
[190, 190, 15, 16, 14, 193, 13, 15, 20, 20] Zhihu.com
[15, 16, 17, 16, 28, 14, 21, 15, 15, 14] Qq.com
[17, 18, 17, 34, 27, 14, 16, 31, 21, 14] Taobao.com
[26, 32, 28, 31, 23, 24, 26, 18, 17, 13] Baidu.com

MyDig Resolver

[346, 359, 337, 367, 325, 342, 339, 338, 336, 325] Wikipedia.org
[181, 125, 456, 644, 595, 828, 275, 679, 158, 160] Zoom.us
[92, 78, 75, 84, 79, 82, 69, 85, 84, 105] Live.com
[121, 98, 76, 120, 88, 98, 89, 83, 111, 80] Netflix.com
[800, 6991, 11205, 864, 813, 3941, 880, 3990, 829, 885] 360.cn
[88, 88, 59, 97, 86, 80, 80, 105, 82, 105] Chaturbate.com
[404, 412, 402, 396, 394, 388, 398, 386, 401, 396] Zhihu.com
[153, 147, 149, 146, 163, 155, 153, 148, 145, 165] Qq.com
[156, 148, 162, 391, 145, 129, 370, 401, 417, 143] Taobao.com
[404, 357, 396, 401, 431, 391, 381, 394, 390, 385] Baidu.com



After running the experiment on 10 websites we get some very interesting results. We see that 8.8.8.8 perform very similarly and very quickly while my dig program runs rather slowly. This is to be expected since they both use caching and therefore don't have to go through the traverse through the DNS structure every time a user requests for a name resolution. As a result, their data spreads are also much tighter since they don't have to deal with a lot of network congestion that my dig has to go through. We also see sites like weibo.com and yahoo.co.jp have one large time that the local DNS and 8.8.8.8 takes followed by really quick times. This also makes sense since after we make our first 2 requests, the address of the URLs should've been cached by the DNS servers while in my dig, I never implemented any caching leading to consistently high resolution times. It also makes sense that these sites weren't previously cached since there likely aren't many people in New York City trying to access these Chinese services when they have local ones like Amazon and other American businesses.

There's also an interesting note about the effect of the location of the user and time of day and the effect those two variables have on the resolution times. While I was on campus, www.amazon.com was resolving to an AWS CloudFront server that I believe was located on campus, but at home in the city, it was resolving to an Akamai

server. This is likely due to the CDN giving me a closer server depending on my location. Addresses in China were also taking much longer to resolve at night, sometimes taking over 10 seconds, or even timing out, however, resolution times in the daytime were much shorter and more reasonable. This is likely because most of the people accessing those name servers live in China and they're asleep while it is the daytime in the USA, therefore there's less congestion. With that same logic, when it's the nighttime in the USA, it's daytime in China meaning traffic is increased leading to longer resolution times.