1. See screenshot.

A screenshot of a computer

Description automatically generated

1. A screenshot of a computer

   Description automatically generatedA screenshot of a computer

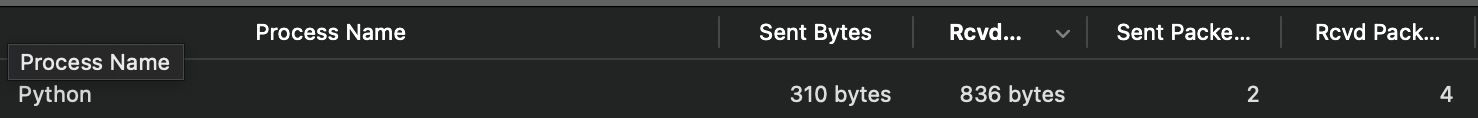
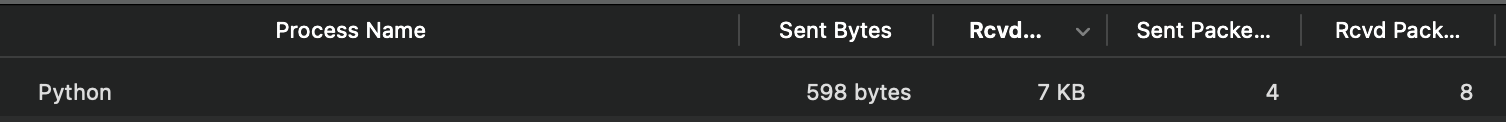
   Description automatically generatedSee screenshots.
2. The GET request encodes the parameters in the GET portion of the URL. The POST request encodes them as part of the HTML form, not the URL.
3. The web server knows that the client is using a Mozilla browser on Linux, Firefox v102.0. It knows that the browser supports webp, xhtml, html, xml, avif, and is English language based. The web server also knows the remote IP of the client. Additionally, the web server knows that the client wants gzip or deflate compressed files.
4. See screenshots.

A screenshot of a computer

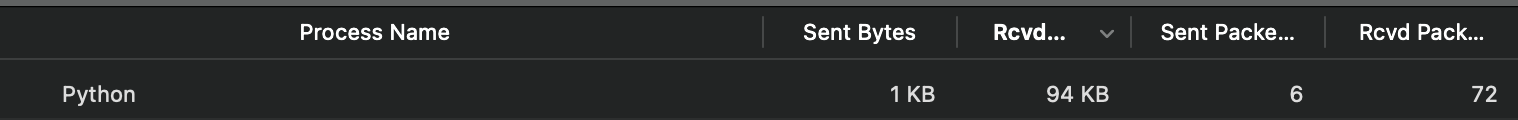
Description automatically generated

A screenshot of a computer

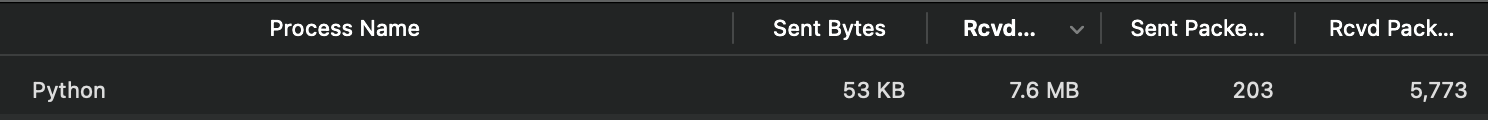
Description automatically generated

1. The only major difference is in the HTML content provided back to the client – one says GET and the other says POST. The clients know that this is an apache 2.4.53 server running on Rocky Linux 9, running PHP 8.0.27. The client also knows the server’s keep-alive TCP connection timers, default character set of utf-8, and that it wants to use chunked responses for large amounts of data.
2. 5.5 KB were exchanged throughout the course of the two transactions. This includes a 404 or two when I messed up an address.
3. There were two requests made to the web server VM. 
4. There was 836 bytes of inbound data from said web server.
5. There is not much information sitting on the web server with no external references and only a couple internal references.
6. There were 4 requests made to the wordpress VM. 
7. There was only 7KB of traffic transferred from the wordpress VM. The amount of data transferred is much larger than the web server because wordpress pages are javascript and php, not just basic HTML.
8. There is no data in my Wordpress web site, so there is almost no data to find except the home page.

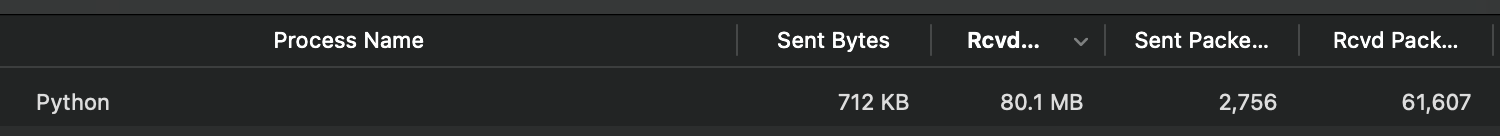
Depth 1:



Depth 2:



Depth 3:



1. Depth 1: 6

Depth 2: 203

Depth 3: 2756

1. Depth 1: ~94 KB

Depth 2: ~7.6 MB

Depth 3: ~80.1 MB

1. I would be concerned about being DDOSed. My internet connection is only 500Mb/s, and that could easily be overwhelmed by a single host sending rapid connections.
2. I would use a DNS provider like Cloudflare to prevent DDOS attacks. Cloudflare acts as a proxy and rate-limits web traffic to a reasonable limit. Additionally, setting up a proxy server in AWS and redirecting all my traffic to pass through that.