Sean Kells

CSEC.380.01

Professor Chaim Sanders

09/07/2021

**Web Server:**

With our provided freedom in choosing the technologies for acts 1, 2 and 3 I decided to choose an Apache Web Server, Varnish reverse caching proxy, and a HAProxy load balancer. My reasoning for choosing an Apache server was I know the general configuration of the server and the set up is all performed all internally for Docker’s httpd image.

**Reverse Caching Proxy:**

My reasoning for choosing a Varnish reverse caching proxy was due to recommendations I found on blogs and articles comparing Varnish versus Nginx or Varnish versus Squid. In particular these websites all seemed to indicate that “Varnish has been noted to serve **content at very high speed**.” Within the comparison between Squid and Varnish it was noted that Varnish is the best thing that you can get if you are not keen about Windows, and you are looking for an HTTP accelerator proxy for Linux and BSD based applications only” which fit my Apache web server well! In addition, I didn’t want any additional web functionality in my image that would be present in Nginx. Rather, a specialized reverse proxy is more desirable. Some of the sources I used for my reverse proxy decision are listed below:

<https://blog.resellerclub.com/varnish-cache-vs-nginx-cache-performance-comparison/>

<https://www.it-consultis.com/blog/how-drastically-improve-website-speed-using-nginx-and-varnish>

**Load Balancer:**

For my load balancer I decided to choose HAProxy. My reasoning for this is very similar to the Varnish, the fact that HAProxy “was designed from the start for high performance load balancing”. In other words, specialization. Why would I choose a service that has additional features that only serve to take up space and are not specialized for my desired functionality (Nginx, Apache)? In addition, there are numerous online resources that utilize and portray different scenarios of HAProxy. Specifically, HAProxy working with a Varnish Reverse Proxy being one of the more popular instances. The HAProxy official website says it the best: “They complete very well together: **Varnish** will make the website faster by offloading static object delivery to itself, while **HAProxy** can ensure a smooth load-balancing with smart persistence and DDOS mitigation.” Yes, HAProxy officially endorses utilizing its product with Varnish despite them being competitors in that they are both reverse proxies. Some of the sources I used for my load balancer decision are listed below:

<https://www.loadbalancer.org/blog/nginx-vs-haproxy/>

<https://www.haproxy.com/blog/haproxy-and-varnish-comparison/>

**Why might these technologies be needed?**

A reverse caching proxy would be significantly useful in a larger consumer environment where a large number of users utilize a web page. The proxy ultimately provides two main benefits, it lightens the load on origin servers and provides some additional security by hiding our web server’s IP address behind it. A load balancer is also significantly useful in a business of any size. I would argue that the load balancer benefit of providing back-up servers and/or reducing the load on a single server (regardless of if it’s web, ftp, etc.) is extremely useful.

In our scenario, the complexity behind our webservers is completely unnecessary as our customer base is merely one individual, my site is not vital to stay up for an extended period of time, and has no need to be highly efficient (why spend the resources if it’s not necessary?).

**Topology:**

