CS 405 Final Reflection
Mark Webster

Throughout this course, I've been challenged quite often on my traditional views of security. While before I had more of a “perimeter-centric” view, where the best way to defend a system was to construct virtual bulwarks like strong firewalls and anti-virus, I have definitely embraced the zero trust model. Our discussion of Triple-A policy, as well as our examination of defense-in-depth and several high-profile security breaches, have led me to gain a larger view of preventative measures. Logins can be stolen, IP addresses spoofed, and code weaknesses exploited. The best approach is to continously verify that users accessing a system are who they say they are and that they have the requisite authorization to do what they are trying to do.

This goes beyond just human users as well. In my other course, CS 470, we designed security policy for various AWS components, completely spelling out what each API endpoint or virtualized function could execute and access. Any system resource can be hijacked for nefarious means. Partitioning the system allows us to mitigate the ease of access that any malicious actor may have even if they gain access.

While planning, designing, and implementing security measures—both physical and digital—are costly in terms of time and capital, the negative consequences of a security breach far outweigh the increased overhead. In fact, integrating security at all levels can actually alleviate work over time: set up those automated tools, and you're set for a while. Ultimately, we need to be paranoid at all times: in the computer age, anyone can say that they are anyone, spoof credentials to that effect, and wreak havoc. Utilizing defense-in-depth and layering our security measures in the only way we can ensure the best protections. All avenues in and out of the system need to be constantly monitored and verified, code needs to be maintained to the highest standard of SEI and CWE standards, and ongoing security testing must always be a priority after a product ships.