



**Leonard Clinton Williams III [REDACTED]@gmail.com>**

Jul 9th, 2023, 7:14 PM

**Re: Email for July 2, 2023**

**To:** "Division, Criminal (CRM)" <Criminal.Division@usdoj.gov>, civil.feedback@usdoj.gov

"if you go online and look for books on how to build ecommerce sites, you get result after result on how to do it with php and MySQL"

-the book I bought on how to make an e-commerce site, I never used. It was like a recipe book for a site with a specific look and specific source code. It showed how to set up stripe payments, and my clients threw a wrench in the system by wanting to use Authorize.Net near the beginning.

I had never written a line of programming in my life, except for one time in 2010 that I had attempted to write a mail script. I copied and pasted most of it, if I remember right, and I couldn't get it to work to save my life. I realized, after I got good, that the reason is that you can send emails from a local server installation on a windows PC. I didn't remember anything about how to write code, when I started the site for my clients in 2017.

I had 4 or 5 static pages and then the e-commerce. I got the static pages done by end of april 2017. The e-commerce was what required programming. I think I was busy with my job at Wells Fargo, or something, because I didn't start the e-commerce until June 2017. I bought some cheap udemy courses and was off to the races.

My process, up to September and October 2018, when I finally finished, was to work on the website, get something working, then get something else working, and so on, until I hit a snag. I would typically hit a snag after very small increments of progress. I would buy books and videos and do google searches, until I got to the next step. It was like I would know what the next question is when I got to it. I'd have to guess as to where the answer would be, and I'd just add to my pile of books and videos.

It was brutal as hell. I was learning at warp speed, compared to the learning track a college student studying computer programming would. What I mean is that I was moving on to more and more difficult problems at a very fast rate, and I was learning the hard way. I would get stuck on problems that would have me ready to scream and to pull my hair out for 20+ hours straight. I'd do this for days in a row.

At one point, I had the store working, except every item that anyone on any computer put in their shopping cart would all go into the same shopping cart. It was like there was a shared shopping cart for every customer. I found this out when I put it on a remote server and tried shopping different devices that I owned. It was murder finding the error, with me not being an experienced developer.

Part of my problem was that at first, I knew no better but to write one big script for everything that I had to do on a given page. I'd have to go through one line at a time, trying to figure out where the bug was. I'd often mess up my mega scripts, editing the wrong thing. I did figure out eventually that the way around this is to write a tiny bit of code and see if it works, then write more, etc. Doing this, I would still end up with a mega script, just one that was written a little at a time. It is very easy to have unintended bugs in your software.

But I did it. Shortly before I finished the version of the site that would go to production, I hired a guy to audit my code for security flaws. He lived in a foreign country that has a lot of programmers. He spoke several languages and had developed award winning software for Fortune 500 companies. He charged me only \$48 an hour. He found some things that would be security flaws, like database queries that were not parameterized, but when I pointed some things out to him, he said that for me this wasn't a problem, but that I should probably fix it anyways. I asked him if someone who was fresh out of college with a 4 year degree could do what I had done, with having this working e-commerce website. He said that they could not.

I would end up going to the [authorize.net](http://authorize.net) developers forum and teach and solve the problems of developers who had been writing software for 10 or 20+ years.

---

Warmest Regards,

Clint Williams

(980)-