Security can often be confusing in its terminology for those who are unfamiliar with the field, as there is no shortage of terms that betray their simplicity with a complex form, acronyms, and initialisms. CA’s, or Certificate Authorities, are one such entity. For the everyday person, think of a CA as a Notary. Notaries are people who are known and trusted to observe and certify that a signature on a document is authentic, and while you may not know the notary personally, they are held to a standard as an impartial witness that allows you to trust that the signature on the document is indeed authentic. A stamp from a notary along with their signature certifies that the signature on the document is authentic.

This comparison may make it more obvious why we use CA’s, as we are rather uncertain about the people we talk to over the web, and having an impartial party to verify things can help us be certain in our decision-making process. But what if you did not want to shell out the money to pay the notary? Or, if you and someone you already trust are communicating, do you need to keep using the notary? No, not always. If I am communicating with my best friend of over ten years, seeing his signature is certification enough. Neither of us is about to spend the money to go to a notary for each other unless the matter becomes rather serious. This is rather similar to how developers use self-signed certificates, as we are sending them from a trusted source (ourselves) to another trusted source, and so we need not involve a third party and incur an expense on an already secure enough communication. But, if your program handles money or personally identifiable information (PII) or any other sensitive piece of data, it may be prudent to turn to that neutral third party and cough up the coin.

A screenshot of a computer program

Description automatically generated

[[View Certificate on GitHub]](https://github.com/Kubia-Beta/Software-Security-CS305/blob/main/Mod5/server.cer)