Research design:

Security on vehicle communication protocols

Journal 1: Security on in-vehicle communication protocols: Issues, challenges, and future research directions

https://www.sciencedirect.com/science/article/abs/pii/S0140366421003297

Notes:

- Data security necessary for functionality and safety of the one driving the vehicle
- Could ECU's eventually be hacked as they become more and more developed and more connected to the internet?
- These devices only should allow access from authorized users, any outside users could put drivers at risk, or damage the functionality of the vehicle
- Number of code lines in cars now surpass 100 million (exceeding that in regular airplanes)
- Now cars interact with their environment, something that was never done before on vehicles
- It is not highly likely, but it has happened - hackers have manipulated code of vehicles not their own and even controlled automobiles
- Electronic Control Units is the computer/brain of the car; it holds all of the code
    - Nowadays there's more code being stored in these ECUs than ever
    - More than that, there are hundreds of ECUs in individual vehicles now
- Example of extreme cyberattack would be hacking a car that has the ability to autonomously drive - this is a huge danger
- These vehicles are not originally designed to be secure against cyber-attacks, which is why this needs to be developed more so now than ever

Article 2: https://xiphcyber.com/articles/automotive-hacking

- Various methods of hacking
    - Key fobs
    - Mobile apps and remote control
    - Wifi and Bluetooth vulnerabilities
    - GPS
    - Server hacking
- Kevin Mahaffey and Marc Rogers, showcased how they could take control of a Tesla Model S at the 2015 DEF CON hacking conference.

- In 2015, cyber researchers Charlie Miller and Chris Valasek hacked a Jeep Cherokee and took remote control of it from a house around 10 miles away using a laptop.
- “The incidence of automotive hacks increased by about 225% in the last five years, and remote attacks accounted for about 85% of all breaches, according to the latest Upstream Global Automotive Cybersecurity Report.”

Brainstorm on methods to use to battle this:
https://brightsec.com/blog/best-practices-for-secure-coding/

- Input validation
    - As simple as this may seem, if you are getting garbage inputs, the program should be able to recognize it and not computer - so if there were malicious attacks regarding inputted information, the program would simply stop or perform a loop that would get the hacker stuck
        - This really only works for inputs, though, not changing code within the program
- “Sanitation”
    - Cleaning and filtering data entered to stop hackers from adding malicious code
    - For example, a hacker tries to insert code that the program isn’t familiar, so the program simply changes it to something it is, rendering the attack essentially useless.
- Authentication
    - “Verifying identity of the user, system, or entity”