Mobile Vehicle Cybersecurity Using a Secure Bridge

Design Document

Team Number: sdmay23-14
Client: John Potter
Advisers: Joseph Zambreno

Team Members/Roles
Josue Torres - Coding/Testing
Ryan Campbell - Cryptography
Cody Stricker - Coding
Levi Jansen - Coding
Drake Ridgeway - Software - coding/testing
Riley Lawson - Coding
Ryan Scehovic - Coding/Testing

Team Email: sdmay23-14@iastate.edu
Team Website: sdmay23-14.sd.ece.iastate.edu

Revised: 11/29/2022 Ver. 0.1

Executive Summary

Development Standards & Practices Used

Hardware - Personal Computers

Software - Linux Red Hat, Emacs Client, OpenSSL, Linux Terminal, C, Vehicle ECU, CANSniffer, socketCAN

Engineering Standards - J1939, AES-128

Summary of Requirements

- Devise an encryption program/algorithm
- Simulate the ECU communicating with each other

Applicable Courses from Iowa State University Curriculum

- CPRE 288 - Reading Datasheets (so we can determine bitfields of CAN/ CAN FD frames)
- COM S 311 - Creating algorithms
- S E 421 - Basic cybersecurity tactics
- S E/CPR E 185 - Introduction and practice with C
- Cyb E/CPR E Cryptography - for analyzing openssl and understanding how crypto can be used within the CAN FD network.

New Skills/Knowledge acquired that was not taught in courses

- Working with CAN / CAN FD
- Simulating a CAN MITM attack on a virtual machine
- Simulation tools to mimic a CAN/CAN FD network
- Cryptography Libraries

Table of Contents

Definition(s):

- CAN (FD)
  - Controller Area Network (Flexible Data)
- Nonce
  - Number used once
- ECU
  - Electronic Control Unit
- AES
  - Advanced Encryption Standard
- OpenSSL
  - Open Secure Sockets Layer

1 Team

1.1 Team Members

Josue Torres

Ryan Campbell

Cody Stricker

Levi Jansen

Drake Ridgeway

Riley Lawson

Ryan Scehovic

1.2 Required Skill Sets for Your Project

Cryptography

CAN Bus knowledge

C Programming Skill Set

Determination to learn new skills

1.3 Skill Sets covered by the Team

Josue Torres - Coding/Testing

Ryan Campbell - Cryptography

Cody Stricker - Coding

Levi Jansen - Coding

Drake Ridgeway - Software - Coding/Testing

Riley Lawson - Coding

Ryan Scehovic - Coding/Testing

1.4 Project Management Style Adopted by the team

We adopted the waterfall model of management. This is because while meeting with our client we already discussed a method of implementation. The project has a pretty clear goal in mind and therefore waterfall worked best.

1.5 Initial Project Management Roles

(Enumerate which team member plays what role)

Ryan Scehovic - Coordinator / Developer
Ryan Cambell - Coordinator / Developer
Josue Torres - Developer / Researcher
Cody Stricker - Developer / Tester
Levi Jansen - Developer / Tester
Drake Ridgeway - Developer / Researcher
Riley Lawson - Developer / Tester

2 Introduction

2.1 Problem Statement

The problem that our project is intended to solve is the lack of security in mobile vehicle networks that run CAN: controller area network, rather than ethernet. CAN is an older implementation of how Electronic Control Units (ECUs) in cars, such as windshield wipers, transmission, brakes, radio, all work. When this type of vehicle network was created, they originally were just happy that they got something to work, and security wasn't even an afterthought. Now, it is Team 14's duty to implement a security device that will prevent the electronic signals being sent from the main control unit in the vehicle from being tampered with by malicious instructions. Some of these malicious things could be but not limited to: causing the car to drive into the ditch, blasting the radio, disengaging the brakes from use, locking the car in a specific gear on the transmission, or even unlocking the car and stealing it.

2.2 Intended Users and Uses

Car Manufacturers:

Characteristics

This is a broad group of users involving anyone who builds and manufactures vehicles for a company. The demographic for car manufacturers are generally going to be older (21+), of any race and origin, both male and female, varying education, various living situations and various families. For example, some manufacturers may have a mechanical engineering degree, some may have an industrial engineering degree, or some may have gone to UTI and went through their mechanic program there. The personality of those that are care manufacturers could vary widely. For example, someone who is the CEO of a company like Ford, Chevy, Dodge etc.... May be very strict when it comes to rolling out new vehicles, where they might be the most passionate about the project, as well as the engineers (which also fall under this category) who design the actual car and

features. On the other hand, we will have the line workers / blue collar workers who might not care as much about the product as they (most likely) don't get paid as much as the engineers or CEOs do. The values that those who fall under this category may have are, but not limited to: creating a valuable career to support their lives and their families' lives if applicable. To create a product that will be valued by the people purchasing it without the worry of it being hacked.

Needs

Many car manufacturers today are currently looking for a solution that prevents their vehicles from being hacked. Manufacturers know that an unsafe car means unsafe customers. This in turn results in bad press for the company well as loss of money due to lawsuits and damages to products. If a car manufacturer isn't looking for a solution to this problem they likely will be as soon as an incident involving one of their vehicles happens.

Benefits

These manufacturers will gain an increase of profit knowing that the solution provided will work in a backwards compatible manner. It will also allow them to apply this solution to many other cars that use the CAN FD system since it can be used universally and isn't necessarily locked to one specific branding.

Distribution Companies:

Characteristics

These companies would be able to send out their semi-truck drivers across the country and back for thousands of miles, knowing and confident that their trucks have the latest security on them. If there is one thing that is important in the United States in the transmission of goods, a vast majority of them get transmitted by semi-trucks. These trucks having the latest security would prevent lives being in danger, companies losing a lot of money, and companies not receiving their products etc.

Needs

A distribution company wants to be able to send drivers and products out with the peace of mind that both will arrive safely and on time. Currently without security on their trucks, these companies are at a large risk of their product being damaged or stolen in a hacking related attack. Even a single hacking related attack on a distribution company would result in a major loss of money. Knowing vehicles aren't safe to drive would slow the transmission of goods, so distribution companies want the latest and best security on their trucks. Throughout the world trucks are the most common way of transporting goods. Keeping this supply chain in good health is vital.

Benefits

With our product distribution companies will have peace of mind knowing their vehicles are not susceptible to CAN network attacks.

Drivers

Characteristics

These are everyday people that do pretty everything and anything. So any user can use one particular thing more than another. However, all of them use the whole vehicle even if they aren't thinking about it. However, safety is often a concern of most vehicle users. These users often explore every possibility of a vehicle problem and can even be the people who are trying to break into other vehicles. Overall, this problem will affect all of the users in some fashion ranging from utilization to driving.

Needs

Fully electric cars are being rolled out from companies and are expanding rapidly to other companies. These cars need to have the latest security on them as well. Companies like Tesla are a step ahead with this, however this year there have been a few cases of Teslas being hacked.

Kay, Grace. "A 19-Year-Old Security Researcher Describes How He Remotely Hacked into over 25 Teslas." Business Insider, Business Insider, http://bit.ly/3XJlTWn

This document shows how a 19-Year-Old was able to hack into 25 Teslas just this year. This is not only a problem for Tesla, but for any company producing these electric cars. So the security on these cars needs to be improved. Preventing the endangerment of lives for many people driving these cars is absolutely vital to us in this project.

Benefits

An everyday commuter will likely not even know that our product is installed in their vehicle unless specifically told. There is an expectation that their vehicle will be safe from remote hacking. Car Enthusiasts: Characteristics These Enthusiasts like their cars to be excellent in all categories including the body, interior, and engine. The cars that they drive are extremely nice looking whether that would be an old vintage car or a brand new off the line vehicle. The securities of these vehicles are of the utmost importance to these people due to the fact that they are made bigger targets. These people more than likely don't want to worry about their car being taken advantage of by some security risk. Needs The needs of a enthusiast are relatively the same as a regular driver, except they are expecting a slightly higher amount of security due the amount of time and money they are placing in their vehicles. This will prevent their expensive, custom, and unique cars from being taken control of without proper authentication with the increased cryptography. Benefits There are an infinite number of hobbies out there, and a car

enthusiast is nothing short of one. Having a vehicle that is all-electric and claimed "unhackable" would be a very nice car to have, show off, or sell.

Original Equipment Manufacturers (OEMs)

Characteristics

Currently, OEMs that produce CAN bus networks for heavy-duty vehicles use the J1939 standard, which defines a network which has been touted as open and unsecure. These OEM companies are in the industry of manufacturing a system in which various devices within a vehicle are able to communicate with each other. They are focused on the reliability and robustness of the CANbus network, ensuring their resilience in a harsh environment. Their main characteristic would be developing a lucrative device which is desirable to car manufacturers.

Needs

The needs of OEMs include a standard, or spec to which to base their CANbus products from. Also, with security being of bigger importance in the present day, they need to develop a system which is insusceptible to threats. There are now many examples of the CANbus networks being compromised. With this in mind this new system needs to be inexpensive and easy to integrate.

Benefits

This is where the idea for a secure bridge comes in. As mentioned in the problem statement, the bridge will secure the data being sent between the CANbus networks. This will fulfill the need of having a more secure network, which will bring about many benefits. The most notable being the robustness of the network. Also, OEMs integrating security into their products will make it more desirable for the vehicle manufacturers to integrate them into their own products. At the end of the chain will be the customers of the vehicles who will benefit by having technology that is less susceptible to intrusions.

2.3 Requirements & Constraints

The Mobile Vehicle Security Bridge will be able to detect instructions sent by a malicious user, whether they be replayed instructions that a hacker reads while sniffing on the local CAN network, or artificial. These fraudulent messages will be ignored which will allow the vehicle to keep functioning normally.

We will need access to cryptography libraries to implement the AES encryption system, and socket CAN libraries.

The product is a box that needs to be durable, able to withstand any situation, holding the technology inside. The box is roughly 8x5x2 inches. We think approximately between 5-10 pounds max is suitable for this project.

It is just a simple monocolored box. It is not something a user physically sees inside of their vehicle; it's a hidden component. The product is comparable to any other part for a car and will most likely remain untouched unless the user intends to find it.

The user shouldn't have to worry about their car being hacked due to this device and have one less stress while driving. The user doesn't need to activate anything. The product passively sits, defending any cyber attack against the user's vehicle that may come its way. If the user attempts to hack their own car, this device will prevent that.

The product is mainly electronic, so it will have to be in an internal part of the car and immune to any weather conditions. Additionally, it will have to be secured snuggly so any bumps in the road or a minor accident wouldn't dislodge it from its installed location. It will have to be securely screwed onto the vehicle in such a way to ensure that it stays in place, but can still be unscrewed so any replacement parts or physical updates required by the product can be applied with relative ease.

2.4 Engineering Standards

AES-128 (Advanced Encryption Standard for 128 bits)

- Justification: Our project leader (John Potter) introduced us to AES128 and thought it could be good for us to use it due to it being very secure and efficient, so it should fit within any time constraints we have for communication to happen.
- Within AES-128 we are currently looking at these specific modes:
    - AES-128 ECB (Electronic Codebook)
    - AES-128 CTR (Counter)
    - AES-128 CMAC (Cipher Message Authentication Code)

J1939

- Standard developed by Society of Automotive Engineers (SAE)
- Designed for Controller Area Network (CAN) for quick data communication between Electronic Control Units (ECUs)
    - Commonly used in heavy duty tractors, cars, and buses

3 Project Plan

3.1 Project Management/Tracking Procedures

For this project, our team is planning on adopting the waterfall model rather than the agile method since this way we can help each other with implementation processes. The reason we are going with this approach as compared to agile, is that this group is not cyber oriented, so working together on new concepts will work best. How we will track our progress as a team is by using Git, Github, and Git Milestones, so we can track who is contributing what, when, and what people are currently working on. We summarize the accomplished work in a weekly report every Friday.

3.2 Task Decomposition

The tasks required for this project are the following:

- Choose programing language
- Find a crypto library that works best with AES-128 and C.
- Setup virtual testing environment
- Figure out how to simulate CAN data for testing.
- Distribute workload amongst team members, finding strengths to ensure everyone is comfortable with their own goals
- Figure out how to upgrade a CAN frame into CAN FD for an expanded byte-size frame for security implementations.
- Individually familiarize with chosen language syntax and development practices.
- Understand how to encrypt and decrypt CAN frames using chosen language
- Ensure that data travels through encryption software at expected speeds and that malicious messages are caught.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

All tools have been researched and decided (crypto library, programming language, etc.)

All members of the team feel comfortable developing functions and logic in chosen programming language.

All members of the team have a simulated environment of the CAN bus system up and running on their individual machines.

Method for upgrading CAN frames to implement our improved security solution has been discovered.

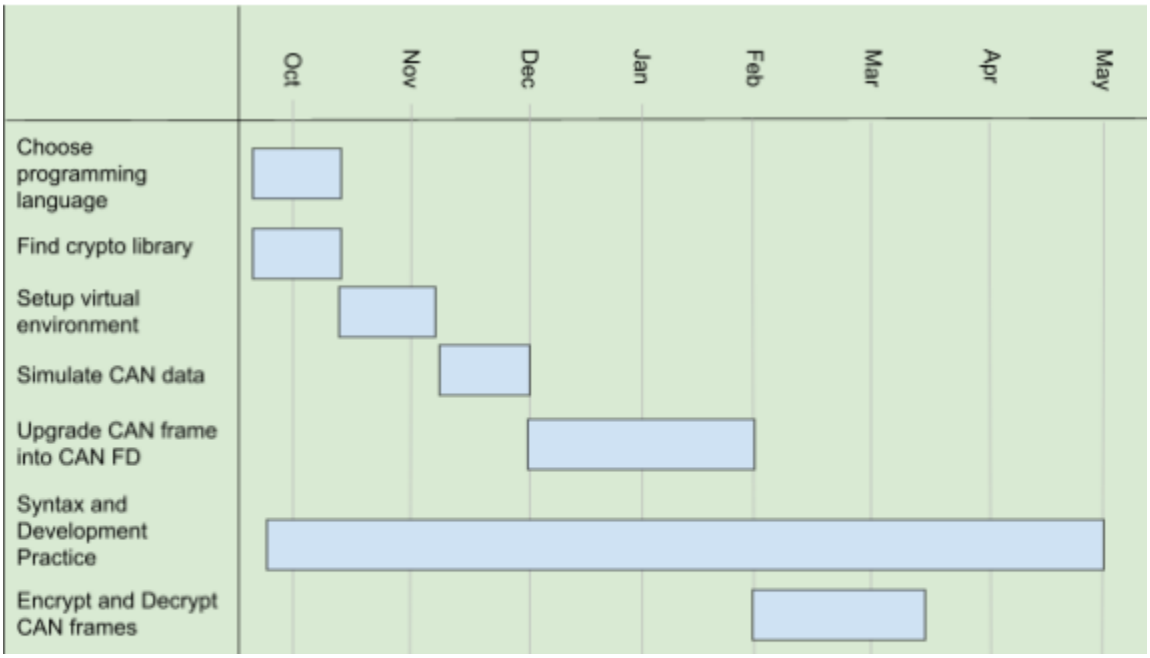
CAN frame encryption and decryption 10% solved, code structured and rough understanding.

CAN frame encryption and decryption 25% solved, able to encrypt data with some implementation

CAN frame encryption and decryption 50% solved, data is running through encryption and decryption cycle with hiccups.

CAN frame encryption and decryption 100% solved, all CAN frames are running through the encryption and being read correctly on the other end.

3.4 Project Timeline/Schedule



3.5 Risks And Risk Management/Mitigation

Task	Risk
Choose programing language	• Not everyone may know or be efficient with the language we decide on as a group. • We may find out later that X language isn't the most efficient with what our project is.
Find a crypto library that works best with AES-128 and Python.	• Make sure nobody gets left behind in the learning phase so we all understand how the encryption libraries work. We will need to thoroughly comment on the code we write.
Figure out how to simulate CAN data for testing.	• No risk involved, this is a much needed step. A risk would be not doing this early enough.
Distribute workload amongst team members, finding strengths to ensure everyone is comfortable with their own goals	• Burning out team members of what they are good at if they are doing the same repetitive thing.
Figure out how to upgrade a CAN frame into CAN FD for an expanded byte-size frame for security implementations.	• Misunderstanding the bit fields of CAN FD, or overwriting data into important bit fields.
Individually familiarize with chosen language syntax and development practices.	• No risk, this is a much needed step.
Understand how to encrypt and decrypt CAN frames using chosen language	• No risk, this is a much needed step.

3.6 Personnel Effort Requirements

To estimate these numbers we took into consideration that we have 7 people on our team so a 1 hour long team meeting would equal 7 person-hours.

Task	Time
Choose programing language	7 hours
Find a crypto library that works best with AES-128 and C	14 hours

Figure out how to simulate CAN data for testing.	35 hours
Distribute workload amongst team members, finding strengths to ensure everyone is comfortable with their own goals	14 hours
Figure out how to upgrade a CAN frame into CAN FD for an expanded byte-size frame for security implementations.	35 hours
Individually familiarize with chosen language syntax and development practices	21 hours
Understand how to encrypt and decrypt CAN frames using chosen language	28 hours
Finalize software and do extensive testing	30 hours

3.7 Other Resource Requirements

Outside of financial resources, other resources would include our own computers (laptop or pc) to make the project from scratch on an IDE with various libraries, test, and simulate the project. We will also use the CAN network for our Security Bridge. We have our client who has provided us with plenty of information on our project. And finally, as a potencial bonus we may be able to test everything using a tractor provided by our client.

4 Design

4.1 Design Context

4.1.1 Broader Context

The communities this project is being designed for encapsulates many communities across the globe. However, most importantly, this will directly impact the daily commuters, public transportation, and any commonly used heavy duty machinery that has a CAN-based vehicle network. People across the globe, rich or poor, use these vehicles. so there aren't exactly specific demographics we can point at. What we can look at in a defined matter, however, is what these various groups of communities will benefit from with this project; a well-curated module that will solve security risks involved with their everyday lives.

Area	Description	Examples
Public health, safety, and welfare	Creating a secure Mobile Vehicle Bus network will disallow malicious users across the grobe from creating threats in communities that are vulnerable to traffic incidents	They won't be able to control the steering of the vehicle to hit pedestrians, harm the driver, or other vehicles on the road
Global, cultural, and social	Every community and every culture deserves the right to feel safe, especially when it comes to mobile vehicle safety. Modern cars universally are pushing new and extensive safety measures, and this is one further step to ensure no bad actors can steal away that feeling of safety. The publicizing of this technology will also, in theory, boost car sales if the economy is doing well. A few of the older generations don't necessarily trust modern technology advancements	By making a public statement that car companies and other vehicle manufacturer's are pushing for public safety, regulation and trust, more people will trust the technology and be more willing to use it. Some people drive older cars out of fear of fear for technology being abused and against their safety
Environmental	However, the chips used in the Mobile Bus device will cause harm to the environment once the vehicle goes to a junkyard. (Very minorly as compared to everything else around it.)	Decomposition over time will pollute the environment that the vehicle's final resting place is in.
Economic	The economic impact that our project will have on the vehicle market, we can see impacting machinery in different ways. Companies will have to be careful how they let this adjust their profit margins. We can see companies being greedy with this technology, but that would also ruin their reputation on whether they're doing it for the money, or building a reputation in the community for a promising product.	While this is very important technology that can save lives, we don't believe it should cost thousands of dollars of an upcharge just for the general safety that a consumer, or the general public should have when it comes to mobile vehicles.

4.1.2 Prior Work/Solutions

We have a unique situation with our project in which our goal is to essentially peer review a solution which has been discovered. Our advisor, John Potter, has developed a potential solution to the J1939 security issue already, but due to limitations in his ability to have his work peer reviewed, he is unsure if there are shortcomings in his solution. The advantages we have in our project are thanks to John Potter's knowledge in this area. Rather than going into the project entirely blind, in a way we are able to follow in his footsteps while still remaining in the dark about his overall solution. This method will hopefully allow us to develop a different solution than his that's close enough that we are able to compare the two to find strengths and weaknesses in both.

In terms of solutions that allow the protection of J1939 CAN networks there are very few, and none like ours. The most common approach are devices that allow the monitoring of data on the CAN network, but these devices actually make the network MORE vulnerable (Arilou, NNG Group). The main benefit of our device is that it will go unnoticed and isn't intended to replace anything, but

rather introduce the concept of cryptography into a CAN network. A direct pros and cons list of currently available solutions can be found below

Our Solution	Arilou's Solution
PRO - Offers high vehicle safety	PRO - Offers moderate vehicle safety
PRO - Uses encryption to make CAN data secure	PRO - Allows users to read their CAN data
PRO - Installed by knowledgeable manufacturer	PRO - Tested and released to the market
CON - Makes CAN data more complex	CON - Requires users to learn and install on their own
CON - Currently not fully developed or tested	CON - Potentially opens up vehicle to vulnerabilities
	CON - Doesn't use encryption

Group, NNG. "How to Secure Commercial Vehicles: SAE J1939 Cybersecurity." Arilou, NNG Group, 3 May 2021, https://ariloutech.com/news/heavy-duty-vehicles-sae-j1939-cybersecurity/.

4.1.3 Technical Complexity

The design consists of multiple components because we will have to handle encrypting/decrypting messages sent in the CAN system. Part of this will be ways of verifying freshness of messages, if messages were tampered with, and the messages still getting where they need in sufficient time.

Currently in the industry there is no standard for encryption of messages in the CAN systemes, and many are vulnerable to attacks so this project looks to address that. Our client explained to us that there was a past project they did to come up with a solution to this, but that he's trying to see what we can come up with and if there's anything they didn't think of or consider that we will.

4.2 Design Exploration

4.2.1 Design Decisions

1. Choosing to use AES-128 for encryption because it fits the time constraints to encrypt/decrypt blocks of messages. This is important to the project success because our whole project has to do with how we are doing to secure messages on the CAN system, so the standard we use for encryption/decryption will be very important in how we do.
2. Using C - found lots of good resources and libraries for encryption/decryption. This is important to the project success because if we choose a language with good resources/libraries then it could have created unnecessary struggles for our group. Now we know we choose a language that won't hold us back.
3. laptop simulation - saves time so we don't have to test on the actual system and plug into the CAN network in the lab every time we want to test. We'll be able to test on our own laptops whenever we want. This is important to the project success because we would have wasted a lot more time testing than we had to.

4.2.2 Ideation

For picking C as the language we are using we considered a lot of different things. Using the lotus blossom technique we considered the following:

1. Python - language the group is second most familiar with, has a lot of resources/libraries for what we're doing
2. Rust - no one in the group is familiar with it, but we'd be interested in learning it, it's known for performance and memory safety
3. C - group is most familiar with due to past programming classes, it's a lower level language which means we would be responsible for a lot more memory management which could make the code more complicated
4. C++ - a potential alternative to C if we found that we needed the object oriented aspect of C++ over C's solely procedural oriented style.
5. A potentially unknown language - We were open to learning new things if the project required it, and we considered that there could be languages we hadn't heard of or worked with before that would end up being the best fit during the research phase.

4.2.3 Decision-Making and Trade-Off

	Pros	Cons
Python	Python is a simple language that has many libraries involving cybersecurity and encryption.	Python is a language that not all of us know in our group. So it would be a learning curve.
Rust	Rust is known for its memory safety and overall performance, recommended by our client.	None of us know this language, so it would be an even bigger learning curve than Python.
C	This language is the most familiar with everyone in our group as it is the most common language in the classes we have taken.	C is a lower level language that is not known for its memory management. This would make coding more complicated than we think it needs to be.
C++	Good alternative to C if we believe that Object-Orientation is needed for our project.	Not a familiar language to everyone, so a learning curve would ensue.
Unknown Language	The pros of an unknown language could be having a major benefit to encryption or other stuff that would fit our needs for this project.	An unknown language could cause a huge learning curve for our group, could have bad memory management, a low level language without any Object-Orientation, etc.

After the pros and cons, we decided on writing our project in C. This language is something we're all familiar with, so no one is left behind learning the syntax and potentially confused, while

everyone else is writing the code. Debugging in a language you know can be difficult, just imagining trying to debug with a language we're not familiar with helped us solidify our choice of C.

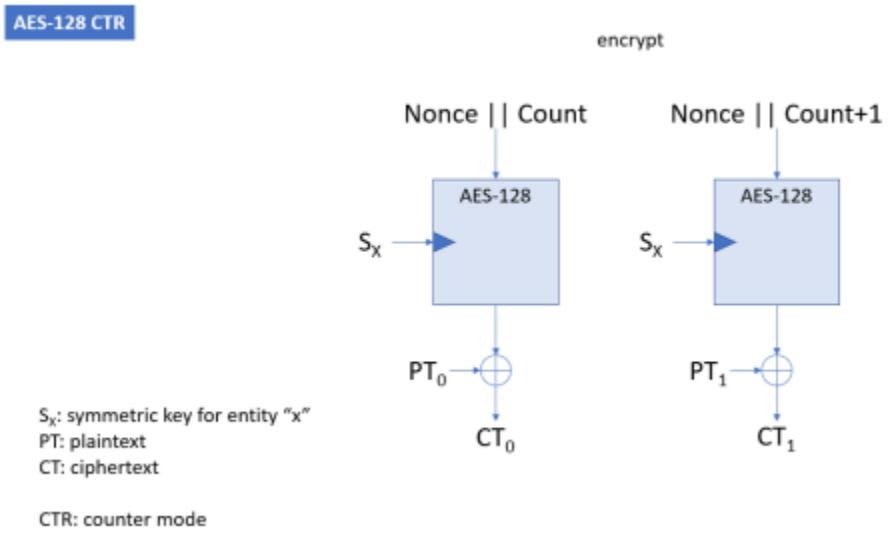
4.3 Proposed Design

4.3.1 Overview

Our current design is to improve the security of vehicles by protecting the data they use to communicate. The vehicle data in question includes things like breaks, transmission, etc. This will be accomplished by adding a small device which reads data traveling through the vehicle, and validates it accordingly. Any malicious or incorrect messages will be caught and thrown out with software that we write for this device.

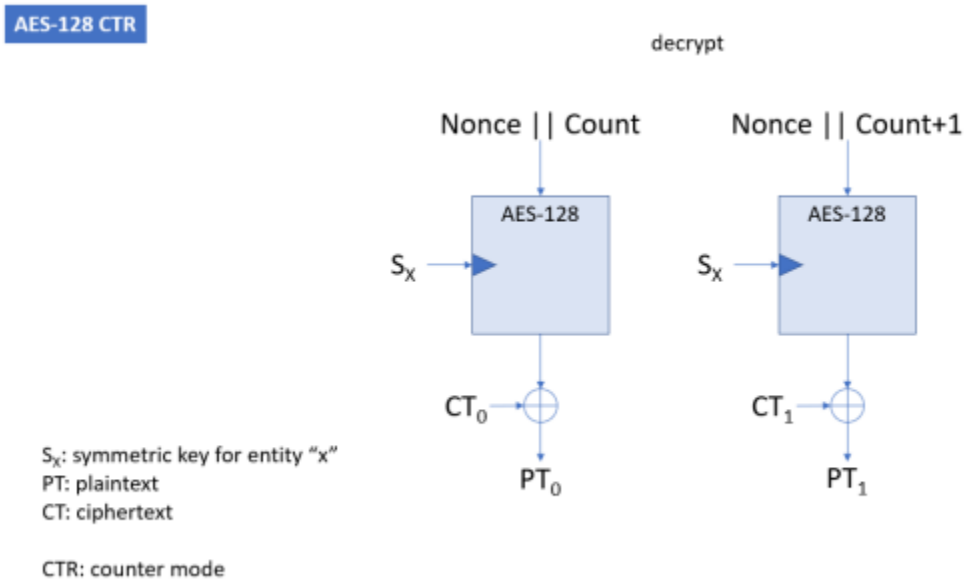
4.3.2 Detailed Design and Visual(s)

Our current design plan is developing a program for encrypting and decrypting a single CAN frame. Once we have it working for a single CAN frame, we will look to test it on larger inputs of multiple CAN frames and make sure the code consistently performs as expected. Here is a diagram of how AES-128 CTR encryption will work:

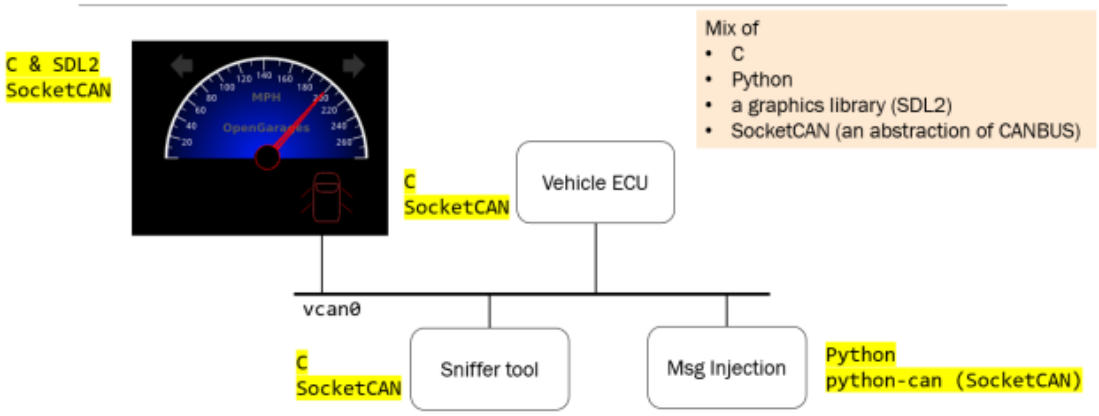


The encryption process works by sending in the Nonce, Count, and Symmetric Key to the program to encrypt using AES-128. Then you take the output of that and XOR it with the Plaintext to produce the Ciphertext.

The decryption portion of the program will work very similarly, except for the ciphertext being XOR'ed where the plaintext was in the encryption program to then output the plaintext:



This is a visual of the SocketCAN network and how it communicates with the vehicle. The messages sent can be related to any variety of things like vehicle acceleration, steering, and other functions of a vehicle. The main reason behind our project is that the current network is not very secure and leaves vehicles vulnerable to being hacked, so we are trying to increase the safety of the system.



4.3.3 Functionality

Our design is intended to operate in the manner that a bad actor won't be able to send repeat data on the CAN bus network, nor their own fabricated instructions. For example, the user will be driving their car, controlling the MPH, steering, brakes, radio. All of these generate signals for the CAN bus to process and execute, but it should do so in a manner that consists of integrity and availability. This means that a bad actor can't send their own instructions to the vehicle CAN network via intrusion because they won't have the "rights" to send data, so instead, it will be rejected by the network and continue operations as usual. Additionally, the CAN network will also not accept repeat data. For example, a person could be snooping on the network traffic being sent

on the internal CAN network of a car and grabbing CAN data frames and repeating them into the system. How our device will prevent this is by using a mixture of a counter and freshness value in each data frame packet that is sent to ensure that old packets can't be sent again.

4.3.4 Areas of Concern and Development

The current design that we have implemented is a good starting point for what our finished product will represent. Currently, we are using the AES-128 encryption standard. We believe that this is exactly what we need to satisfy the requirements and the user needs. Our client specifically told us that this is the encryption standard we will be using as the lower bit size allows us to ensure the availability of the data, as this is a much faster encryption speed than 192, 256.. etc. What we are unsure of currently is how we will prevent repeat data right now. We are figuring that we can use the expanded CAN FD frame to allocate certain bits to be "time stamps" or "freshness values" that will help prevent repeated instructions. The immediate plans for developing the solution to this concern is researching more into the exact bit fields of the CAN FD frame and verifying this, or identifying if some other solution will arise out of the provided documentation for the bit fields. We don't have any questions as of right now.

4.4 Technology Considerations

The technologies that we are currently using include, the C programming language, the emacs text editor, and a CAN BUS simulation program. The advantage of using C is that there are a vast number of libraries for our use case. For example, because we are using AES-128 bit encryption, we found a library in which to build this program. Also, being that C is a lower-level language, we have more control over the lower-level details for our program. Next is the emacs text editor, which has the advantage of being lightweight, and easy to use. Finally is the CAN BUS simulation program, which we are using in order to delve into how an actual attack into a CAN BUS network might happen. One of the outstanding disadvantages to using C is the learning curve associated with it. Being that it is a lower-level language means it is harder to adjust the mindset needed when using it. As for the text editor, although it is lightweight, that means there aren't as many features associated with it. One of the more noticeable downfalls is the lack of a debugging tool. Even though these weaknesses exist, the advantages outweigh said disadvantages.

4.5 Design Analysis

Currently we have found a potential candidate for our cryptography. The library is called OpenSSL and will allow us to use AES-128 with C. It's a very vast tool kit with a lot of documentation online which will hopefully mean that many questions that may come up while we get a closer look at its uses will already be answered. As of now one member of our team has experimented with the toolkit and has reported its uses. We have also individually set up our virtual machines that we intend to use for virtual testing of our code. We intend to begin experimenting with code on these virtual machines soon, but as of now there is nothing to report in terms of issues with getting set up and ready to go. We are still keeping in touch with our advisor, John Potter, and receiving helpful information from him regarding our project about every week. Currently our proposed design from 4.3 is going smoothly and there are no glaring problems that we have had trouble solving.

5 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, power system, or software.

The testing plan should connect the requirements and the design to the adopted test strategy and instruments. In this overarching introduction, given an overview of the testing strategy and your team's overall testing philosophy. Emphasize any unique challenges to testing for your system/design.

In the sections below, describe specific methods for testing. You may include additional types of testing, if applicable to your design. If a particular type of testing is not applicable to your project, you must justify why you are not including it.

When writing your testing planning consider a few guidelines:

- Is our testing plan unique to our project? (It should be)
- Are you testing related to all requirements? For requirements you're not testing (e.g., cost related requirements) can you justify their exclusion?
- Is your testing plan comprehensive?
- When should you be testing? (In most cases, it's early and often, not at the end of the project)

5.1 Unit Testing

The mobile security CAN bus has two functions: encryption and decryption. How we have tested both of these is by creating a C program that uses AES encrypt and decrypt functions using the OpenSSL library. What we are specifically designing this algorithm to encrypt and decrypt is the CAN data being sent between two networks, so we have a CAN data log file that we are iterating through as a test subject and ensuring we can encrypt that data line by line, and decrypt it as well. The tools we are using for this are Red Hat Linux, Emacs, C, and Terminal.

5.2 Interface Testing

The main interfaces of our design are, the vehicle, the vehicle's can system, our CAN bridge device, our code within that device, and the OpenSSL library used within our code. Between our code and the OpenSSL library significant testing and research will be done to ensure that we are using the OpenSSL library correctly. We will be using Red Hat Linux and Emacs to write our code and perform our tests on our code. To test our code within the CAN bridge device we will attach it to a physical system, or a simulated one to validate that we are able to read and output the correct data with the device. Tools for creating a simulated environment will be done through matlab and simulink.

5.3 Integration Testing

Some critical integration paths include our encryption and decryption methods, and printing them out correctly. Receiving and distributing the keys within the encryption and decryption methods is critical for it to be correct. Some of the physical components that need to work for our integration paths to work would be physical parts of our vehicle (e.g. speedometer, engine, steering wheel, etc.). They'll be tested by checking if the key is correct on each end from our program in a simulation from the software socketCAN. Other tools we are using are OpenGarages, Vehicle ECU, CANSniffer, Linux RedHat, and programming language C.

5.4 System Testing

The most significant form of full system testing we will have is running our product on a physical vehicle (tractor). The most obvious way that these tests will be conducted is to confirm that the vehicle still operates as normal while our device is connected. We will also be reading the CAN FD frames to ensure the data is progressing through the system as normal while we also attempt to perform man in the middle attacks on the CAN bus. Other forms of system testing include simulating vehicles through matlab and simulink as well as using basic text files containing CAN data.

5.5 Regression Testing

The list of features that our product needs to have is encryption, decryption, not allow repeats of data, and to deny any Man in The Middle attacks. Right now, we have the encryption and decryption algorithm set, so moving forward, we are going to have to look at the CAN FD frame and see if we can "verify" what is a "good" packet and what is a "bad" packet by utilizing the bit fields of the frame. To ensure that we don't break our code, we are going to frequently compile and ensure it still successfully encrypts and decrypts the data without a buffer overflow, as we are working with C. The list of tools we are using for regression testing are Linux Red Hat, C, Emacs, and Terminal.

5.6 Acceptance Testing

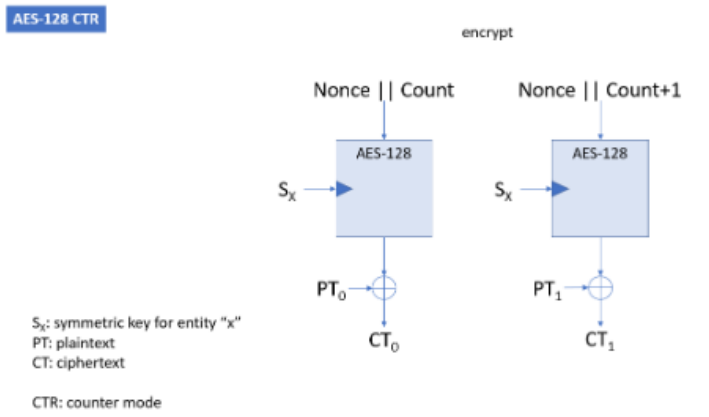
First, we will demo our algorithm with a laptop simulation using socket can, and a demo our client showed us early on in the semester. This demo will prove whether or not our program can survive a Man in The Middle attack, or detect duplicates of data being sent based on freshness values in the CAN FD frame. Once this is a success, our client has mentioned that we will be using our algorithm on a real tractor he will get on campus for us to see how successful we were.

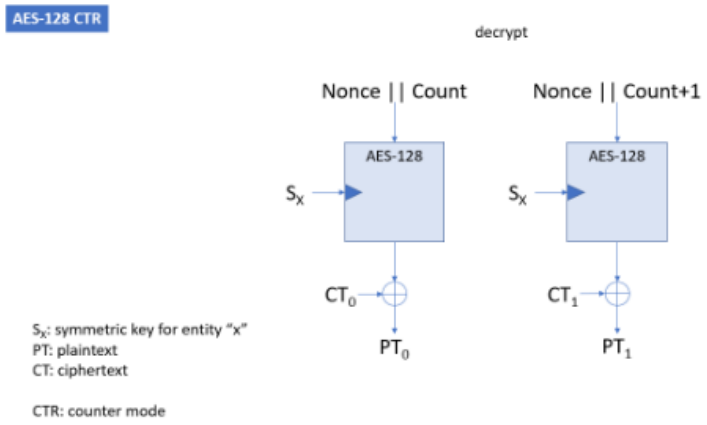
5.7 Security Testing (if applicable)

The main testing for the security of our application is going to ensure that the vehicles that are continuously running this code on their CAN busses won't be susceptible to Man in the Middle attacks, nor will a malicious user be able to send repeats of data sniffed on the can bus network. Right now, we have the encryption and decryption algorithm set, so moving forward, we are going to have to look at the CAN FD frame and see if we can verify what is a "good" packet and what is a "bad" packet by utilizing the bit fields of the frame.

5.8 Results

The main result we'll be testing for is that we can send encrypted messages on the CAN network and that they are able to be decrypted when they arrive at their endpoint. For testing this, we can make sure a decrypted message is the same as the original message before encryption. We are planning on using AES128 CTR (counter mode) encryption, which can be seen in the following 2 diagrams:





6 Implementation

The plan for next semester is going to be working on our encryption algorithm some more before testing it with the laptop simulation to see if it stops a man in the middle attack with just virtual components. If all goes well in this, then we will bring in a tractor and get to physically upload our code to it, and ensure it stops a man in the middle attack or duplicate CAN data sets sent by a malicious user. This would be as close as we get to real world testing and implementation.

7 Professional Responsibility

This discussion is with respect to the paper titled " Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", *International Journal of Engineering Education* Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

Code	In our own words and Difference compared to IEEE CoE
Work Competence	Ensure that the work you are doing is high quality, robust, and on time. IEEE focuses on the quality when growing software as well as peer review of software
Financial Responsibility	Ensure that you are creating products at affordable costs. IEEE does not have a specific mention of financial responsibility as I understood their code of ethics.
Communication Honesty	Ensure that when you share information to stakeholders that you aren't withholding information and being honest about your work. IEEE covers conflict of interest as well as peer reviewing and calling out violations.
Health Safety, Wellbeing	Ensure that stakeholders are kept safe and their well being is looked after. IEEE chooses a more broad approach which involves stakeholder as well as public safety overall.
Property Ownership	Ensure that you are treating others property, information, and ideas with respect. IEEE has more specific coverage in this area including the respect of individuals along with their ideas and information.
Sustainability	Ensure that you are not putting the environment of any scale at risk. IEEE covers the environment briefly inside their safety section, but I believe it is still conveyed well.

Social Responsibility	Ensure that the work you are performing is beneficial to society. IEEE has this spread throughout as it touches on peer review and upholding others to produce quality.

7.2 Project Specific Professional Responsibility Areas

Code	Our team
Work Competence	Low- We have room to improve on our motivation towards the project, but the work we have done has been high quality.
Financial Responsibility	N/A - We currently have used free resources to perform our work and no physical items have been purchased
Communication Honesty	High - We have maintained a high level of honest communication amongst ourselves and our advisor.
Health Safety, Wellbeing	High - Our team understands the seriousness of the problem we are attempting to solve and we all intend to ensure safety is not put at risk during this.
Property Ownership	High - We all have maintained a level of respect of each other and our advisor and have maintained active listening for each member.
Sustainability	N/A - The environment has not been impacted in any meaningful way during our progress thus far.
Social Responsibility	Med - We believe our effort towards this project has been lower than intended, which results in less benefit, but as the semester closes we are all striving to finish strong.

7.3 Most Applicable Professional Responsibility Area

The most important professional responsibility area to us has to be communication honesty. By remaining open and respectful to our client by showcasing transparency with our communication with where we are in our project, we open ourselves and this project to a greater magnitude of

tracking and progress. Even if we get stuck in a spot, it's still vital to the lifeline of this project to be open and honest about our faults and mistakes as well so we can learn from them and stay in a forward motion.

8 Closing Material

8.1 Discussion

Discuss the main results of your project – for a product, discuss if the requirements are met, for experiments oriented project – what are the results of the experiment, if you were validating a hypothesis – did it work?

8.2 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals. What constrained you from achieving these goals (if something did)? What could be done differently in a future design/implementation iteration to achieve these goals?

8.3 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

8.4 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.

8.4.1 Team Contract

Team Members:

1) Ryan Campbell	2) Josue Torres
3) Ryan Scehovic	4) Cody Stricker
5) Levi Jansen	6) Riley Lawson

7) Drake Ridgeway

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:

NAME/DAY	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Campbell	10am+	4pm+	10am+	X	10am+	all	all
Scehovic	2-5pm, ** 7:15pm+	7:15pm+	2-5pm, ** 7:15pm+	2-5pm, 7:15pm+	2pm+	all	8am-5pm, 7:15pm+
Riley	8pm+	8:30pm+	5pm+	5pm+	5pm+	all	all
Josue	4pm+	4pm+	4pm+	12pm+	4pm+	all	all
Drake	4pm+	5:45pm+	4pm - 6:30pm	5:45pm+	4pm+	all	all
Levi	5pm+	4:30pm+	5:30pm+	Before 6pm	X	all*	all*
Cody	2pm+	12:30pm+	2pm+	12:30pm+	X/Depends	X	X/Depends
Total	**8pm+**	**8:30pm+**	**5:30pm+**	**X**	**5pm+**	**All/Online**	**8am-5pm, 7:15pm+**
key							
** = could meet during time marked off if we plan it ahead of time							
X = no times that day							

2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):

   In person or through the team Discord server.

3. Decision-making policy (e.g., consensus, majority vote):

   Democratic majority vote

4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):

   Cycle through each member for keeping meeting minutes; create a shared drive for senior design cloud resources (sheets, docs, links, etc).

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:

   Contribute as much as you can, make an effort to show up, let people know ahead of time if you can't make it / have something else going on 2.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:

   Do your part, if you need help, ask for it

3. Expected level of communication with other team members:

   6-12 hour response time, 24 hours if long / complex thing to respond to

4. Expected level of commitment to team decisions and tasks:

   Be punctual, be open with what you're struggling with, we are a team and open to helping each other where we can

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction,

individual component design, testing, etc.):

Shared leadership, since we are a new group of people working together.

2. Strategies for supporting and guiding the work of all team members:

Weekly or more check-ins with where people are standing on their deadlines / goals / milestones. Gitlab?

3. Strategies for recognizing the contributions of all team members:

Gitlab shows who's committing what, working on what branch, etc.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the

team.

Josue: Computer engineering major. Experience working on UI and frontend software in a team based environment.

Levi: SE Major; experience in databases and interfacing between frontend and backend software. Experience working with a professional team in a 2-week sprint system.

Campbell: Cyber security major, outside of class blue team work with CDCs, taken multiple cyber classes, 288, 381, 309

Drake: SE Major, experience in embedded systems and testing, from CPRE 288 and SE 317 respectively. Unique Perspectives; moral and positive attitude, and tries to make his team have the same mentality. No internship experience.

Cody: SE Major, have experience with vehicle CAN bus communication as well as working in a professional team with a 1-week sprint system. Experience with frontend to backend communication.

Ryan Scehovic: SE Major & Data Science Minor - have had classes on databases, web dev, app dev. Internship experience working with building data flows and delivering features on a biweekly basis.

Riley Lawson: SE Major & Cyber Security Minor - Done UI and Web Development along with some experience with a team in a professional environment (internship), along with a cyber security course, and numerous classes in embedded systems.

2. Strategies for encouraging and support contributions and ideas from all team members:

Listen to all ideas and give it some good insight before making a final judgment, and give a good reason as to why the final judgment was made.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

Bring it up to the team, and the team will listen respectfully and make changes accordingly.

Goal-Setting, Planning, and Execution

1. Team goals for this semester:

Levi: I want to gain experience and see the perspectives on a team of people from more than just one field. I want to learn how my skills fit into a large team project.

Josue: Put the skills I've obtained to use and challenge myself on a rewarding project. Learn how to better handle team conflict. Also learn more about the CAN communication bus.

Campbell: I'd like to learn a lot about more cyber security concepts / RSA encryption / private/public key cryptography, and developing interpersonal skills such as team building, bonding, communication etc.

Drake: I want to learn what it is like to work on a full team with a client, and learning about some cybersecurity will be an added bonus as well.

Cody: I want to create something that I'm able to share with others to say: "This is what 4 years of studying software engineering has done for me." Interested in operating in a professional, fully software focused team, and how I can contribute.

Ryan Scehovic: I want to apply skills I've learned so far through other classes and also learn new things, like cyber security related stuff, to help make for a more well rounded set of skills. I also want to be able to work with the team to build a successful project.

Riley Lawson: Using the skills that I learn from this semester and applying them to a professional environment. I hope to develop communication skills with my teammates and learn how to develop a proper use of the coding knowledge that I will learn over the next semester.

2. Strategies for planning and assigning individual and team work:

Assign work to people who are most comfortable on the subject, if possible.

3. Strategies for keeping on task:

Do your part, if things get difficult, bring it up to the team for help, and the team must be willing to help as much as they can.

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?

Check in on the person, ask how they are doing rather than getting mad since they might be going through a hard time, try and help where we can.

2. What will your team do if the infractions continue?

Bring it to the attention of the advisor, if serious enough, decided by group, deduct points from that person from the final grade.

****

a) I participated in formulating the standards, roles, and procedures as stated in this contract.

b) I understand that I am obligated to abide by these terms and conditions.

c) I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.

1) Ryan Campbell DATE 09-11-22 2)Drake Ridgeway DATE 09-11-22

3) Josue Torres DATE 09-11-22 4) Cody Stricker DATE 9-11-22

5) Levi Jansen DATE 09-11-22 6) Riley Lawson DATE 09-11-22

7) Ryan Scehovic DATE 09-11-22