Missouri S&T Library and Learning Resources

Scholars' Mine

Masters Theses

Student Theses and Dissertations

Spring 2013

# Security analysis of a cyber physical system : a car example

Jason Madden

Follow this and additional works at: https://scholarsmine.mst.edu/masters_theses

 Part of the Computer Sciences Commons

Department:

Recommended Citation

Madden, Jason, "Security analysis of a cyber physical system : a car example" (2013). *Masters Theses*. 5362.
https://scholarsmine.mst.edu/masters_theses/5362

This thesis is brought to you by Scholars' Mine, a service of the Missouri S&T Library and Learning Resources. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

www.manaraa.com

المنارة للاستشارات
www.manaraa.com

# SECURITY ANALYSIS OF A CYBER PHYSICAL SYSTEM : A CAR EXAMPLE

by

JASON MADDEN

A THESIS

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF COMPUTER SCIENCE IN COMPUTER SCIENCE

2013

Approved by

Dr. Bruce McMillin, Advisor
Dr. Daniel Tauritz
Dr. Mariesa L. Crow

www.manaraa.com

Copyright 2013
Jason Madden
All Rights Reserved

www.manaraa.com

# ABSTRACT

Deeply embedded Cyber Physical Systems (CPS) are infrastructures that have significant cyber and physical components that interact with each other in complex ways. These interactions can violate a system's security policy, leading to the leakage of rights and unintended information flow. This thesis will explore information flow as it uses a public channel. In order to exemplify the use of the public channel, a vehicle being composed of the computer system and its operators will show how information is disclosed to an observer.

The example is made up of a vehicle traveling across some terrain with an observer watching the car. The observer then uses the contextual information, based on the topography and previous knowledge about an automobile, to attempt to learn some of the events taking place in the car's computer system and the actions of the driver. The combination of the observer and the passage of information from the car to the observer forms a public channel.

This model is analyzed for both nondeducibility, noninference, and properties about its information flow. In security, the knowledge that information flow exists is a violation. This is known as leakage. To remedy the weaknesses observed during the analysis, a method to obfuscate the information flow is introduced. The fact that important information can be camouflaged, even while it flows over a public channel, is an important observation of this thesis. This process of obfuscation can be applied to other cyber physical systems to secure the public channel.

www.manaraa.com

## ACKNOWLEDGMENT

Foremost, I would like to express my gratitude to my advisor Dr. Bruce McMillin. He has truly been a great mentor and extremely supportive. Without his guidance, motivation, and immense knowledge I would not have made the progress nor learned as much from this experience. I can only hope that I have the opportunity to help someone in the same way that he has helped me.

Additionally, I would like thank the rest of my committee, Dr. Mariesa Crow and Dr. Daniel Tauritz. Without working with these individuals I would have never had the opportunities that I now have. Dr. Tauritz's enthusiasm and knowledge in the computer sciences helped to fuel my interests and lead me to the the opportunity of working under Dr. McMillin. In additiona, I gained the exposure to a different field of study with my involvement with Dr. Crow and the electrical engineers under her. The experience with these individuals has made me a more well rounded person by permanently changing the way I think about and approach problems.

Finally, I want to express my appreciation to all the people close to me: my parents Lewis and Chris Madden for the life, opportunities, the discipline to work hard, and always gave me the extra encouragement to push on; my sister Rachel Kirkpatrick for putting up with my antics and lending a kind ear when it was needed most; my grandparents Dorothy Madden, Richard and Virginia Drone for all the love throughout my life; and Anna Nisbett for the love, support, and encouragement to stay focused.

www.manaraa.com

# TABLE OF CONTENTS

Page

ABSTRACT ..... iii

ACKNOWLEDGMENT ..... iv

LIST OF ILLUSTRATIONS ..... viii

LIST OF TABLES ..... ix

SECTION

1. INTRODUCTION ..... 1

1.1. CYBER-PHYSICAL SYSTEMS (CPS) ..... 1

1.2. SECURING THE CPS ..... 3

1.2.1. Privacy ..... 4

1.2.2. Data Processing and Analysis ..... 5

1.2.3. Modeling and Metrics ..... 6

1.2.4. Real-Time Guarantees ..... 6

1.2.5. Autonomous System ..... 7

1.3. VEHICLE AS A CPS EXAMPLE ..... 7

2. THE ENGINE MANAGEMENT SYSTEM ..... 10

3. APPLICATION OF TRADITIONAL SECURITY ..... 12

3.1. BACKGROUND ..... 12

www.manaraa.com

3.2. HARRISON-RUZO-ULLMAN MODEL (HRU) ..... 13

3.3. BELL-LAPADULA MODEL (BLP) ..... 14

3.4. INFORMATION FLOW ..... 16

4. PROBLEM STATEMENT ..... 18

5. METHODOLOGY ..... 20

5.1. HARRISON-RUZO-ULLMAN MODEL ..... 20

5.1.1. Overview ..... 20

5.1.2. Remarks ..... 22

5.2. BELL-LAPADULA MODEL ..... 22

5.2.1. Overview ..... 22

5.2.2. Remarks ..... 25

5.3. INFORMATION FLOW ..... 26

5.3.1. Overview ..... 26

5.3.2. Remarks ..... 28

5.4. MODELING DISCUSSION ..... 29

6. SECURITY POLICY EVALUATIONS ..... 31

7. ANALYSIS OF EMS USING SECURITY TECHNIQUES ..... 33

7.1. TERRAIN EXAMPLE ..... 33

7.1.1. Nondeducibility ..... 37

7.1.2. Noninference ..... 41

www.manaraa.com

7.1.3. Noninterference ..... 44

7.2. OBSTRUCTION EXAMPLE ..... 47

8. CONCLUSIONS ..... 55

9. FUTURE WORK ..... 57

BIBLIOGRAPHY ..... 59

VITA ..... 62

www.manaraa.com

# LIST OF ILLUSTRATIONS

Figure Page

2.1 Depiction of ECUs in the Vehicle ..... 11

5.1 Multi-Level Security Model of the Vehicle Example ..... 24

7.1 Comparison of Events on Differing Terrains ..... 34

7.2 Information Flow Diagram of an EMS ..... 34

www.manaraa.com

# LIST OF TABLES


| Table | | Page |
|---|---|---|
| 4.1 | Confidential Information in EMS | 18 |
| 4.2 | Security Levels in EMS | 19 |
| 5.1 | List of Events used in the Terrain Examples | 21 |
| 5.2 | ACM of the Physical World in the Vehicle Example | 21 |
| 5.3 | ACM for the Vehicle Example | 22 |
| 7.1 | Events of the Terrain Examples | 35 |
| 7.2 | Trace list for the Standard Cruise | 35 |
| 7.3 | Trace list for the Perfect Cruise | 36 |
| 7.4 | Trace List for the Random Cruise | 36 |
| 7.5 | Obstruction Example Event List | 48 |
| 7.6 | Trace List for the Controllers During the Hazard Scenario | 49 |


www.manaraa.com

# 1. INTRODUCTION

## 1.1. CYBER-PHYSICAL SYSTEMS (CPS)

According to E. A. Lee, the best way to describe a Cyber-Physical system is stated as "... integrations of computation and physical process" [1]. One defining characteristic of a CPS is that its architecture is heterogeneous. The range of devices that construct a CPS includes anything from the sensors that detect some physical aspect in the world, the most simple of hardware, to the high-end work computers and the cloud that manages data and controls the overall system, the most complex of hardware. To be working as a system, information and commands must move from one piece to another through a range of reliable, unreliable, and compromised networks. This implies the requirement that a CPS must be highly reliable, predictable, and secure.

The hardware and software in CPS have a combination of both strict and relaxed operating rules which can make the system complex. Since a CPS interacts with the physical world, there must be a way to translate the real world information into a computational quantity or value. There are four distinct categories of systems that compose a CPS.

- Sensors - A device that translates the physical world to the digital world.
- Embedded System - A standalone device has hardware, software, and mechanical parts that provide a limited range of functions within a system. It does

www.manaraa.com

not require anything special to operate, but it may not be completely useful by itself. This system can perform advanced logic.

- High Level System - A computer, ranging from a desktop computer to a high-end super computer, that can perform mass quantities of computations, advanced logic, and vast multitude of functions at the same time.
- Physical World - Where the system resides which is confined by physics and real-time constraints.

Adding a sensor and tying it to an Embedded System or a High Level System creates a means for the computing device to understand a little more about the physical world. The sensor can be considered a simple device and requires something reading it to become useful. Through a sensor, the high level processing, computations too complex for anything but the high level system to perform, can continuously interact with the physical world and operate against the input provided by the sensors. Ensuring that the information passing, like with the sensors and high level systems, also becomes a very important problem that a CPS needs to address.

Information passing is another difficult aspect of a CPS. Communication is not restricted to specific type of networks or communication paths and CPS typically require many network protocols in order to operate. Both reliable and unreliable networks are being used in a CPS at the same time. A CPS must be able to tolerate the limitations introduced by the networks and still be able to satisfy its requirements of operation. Because data is being communicated, specifically, the communication channels need to be evaluated and properly secured.[2]

Securing the CPS incorporates all the aspects of analysis involved with computers, the physical part of the system, and finally, the interactions that take place between the physical parts and the computing parts. Currently there are security

www.manaraa.com

analyses for the data and computing aspects of the system and there are security analyses for the physical part of the system, but security analysis that encompasses both aspects at the same time does not exist. Since the system includes the physical domain as inputs and outputs to the system, deeper analysis has to be performed to adequately ensure that it is secure. One vulnerable sub-system of the CPS can impact one or all of the connected sub-systems. Security analysis has to consider the physical behavior and how the components interact together.[3]

Due to the complexity of a CPS, there are many properties that force a considerable amount of design and planning to ensure the capabilities expected of a CPS can meet the operating requirements of a CPS. These systems do exist and are becoming increasingly more important in modern computing systems.

## 1.2. SECURING THE CPS

The security of cyber-physical systems is critical and complex. Information is difficult to protect due to the combination of data, communication, processing, and use of public communication channels. This characteristic architecture of a CPS by design cannot circumvent this. Well known CPS systems are aircraft systems, mobile computing networks like cell phone networks, and sensor networks. These structures include many physical communication channels as well as many different security domains.

The fundamental properties of computer security are (*confidentiality, integrity and availability*) are still applicable for the security of a CPS. Confidentiality is the protection of information. Integrity is the validity and correctness of data.

www.manaraa.com

Availability is the capability of using a resource[4]. Typically when one of these properties are eliminated in a system, the system either does not work or does not work correctly.

Ensuring the security of a CPS is increasingly more difficult than modern day implementations of computer systems. The common method to protect networks from unwanted access is by segregating the access of the network from entities that should not access and those that should. Due to the accessibility of a CPS and its resources to the physical world, it is impossible to apply this security principle to it [5]. A CPS can be more characterized as a peer to peer network than a traditional client/server model. This introduces a lot of emphasis on ensuring that entities can be trusted to do their work and that their work is not compromised. A CPS is often defined by being a real-time system. Most, if not all, devices are expected to operate within a very granular time frame. Some of the device's peers may rely on the peer's functionality for its own functionality to work. Because of this, security on a CPS concentrates on these principles:

- Privacy
- Modeling and Metrics
- Real-Time Guarantees
- Autonomous System

A more in depth look of these topics are included below.

**1.2.1. Privacy.** Data moving about the system can have a lot of identifying information about it or the users of the system. Without adequate protection on this

www.manaraa.com

data there may be no concept of confidential data [6]. The leakage of this information risks disclosing characterization of an entity's habits, settings, or functions. By monitoring the information flow patterns and data available moving in the channel, that information can be associated to the entity. Especially with the use of a public channel, an observer will have access to the information moving around the system, the information presented to the observer, and in some cases, can reveal information about the entity behind the system.

**1.2.2. Data Processing and Analysis.** Data processing and analysis includes everything from communication between entities, encrypting the data, and using the data in the system. To elaborate on what was stated before, a CPS can use a varying network structure in order for the devices to communicate between one another. Each network must be able to tolerate the conditions it is in. Furthermore, since a CPS uses public channels, there is a wide array of attacks and interception that can happen to the data as it is being passed.

Not only does the network have to be safe, but since data passes into a public channel, there must be the assumption that the data will be captured while traveling between entities. The communication between entities should account for this. Additionally, once the data is passed, it should be evaluated to ensure that it has not been altered by anything but the source of the data.

The issue is not that most of these problems have been adequately solved, it is that they all have to be solved at the same time. In a CPS there is a vast quantity of data that is being passed, stored, and used which makes the problem quite astronomical. Every device does not support all levels of security, and every layer

www.manaraa.com

of prevention or alteration to the data causes extra work for the processing and the storage of the data that is being passed. These measures have to be accounted for and also determine how the loss or compromise on any piece of the system might impact the system as a whole.

**1.2.3. Modeling and Metrics.** In security, models are the backbone of validating the architecture and protection of a system. These measures ensure that a system will do what it is designed to do. Currently, no model exists that encompass all that is included in the CPS domain. Without the ability to model a CPS, there is no means to abstract the system in its entirety. Currently, a combination of traditional methods and new developing methods are being applied. They work, but it's not the ideal solution [7][8]. It is important to quantify and measure how events can influence the system in order to better understand the class of system can be characterized.

**1.2.4. Real-Time Guarantees.** Adding the notion of time and honoring time in a computer based system makes time the most valuable resource of the system. Tasks that exist throughout a CPS must meet their deadlines, in addition, missing the deadlines can invalidate the correctness of it. The environment of the CPS is also a critical component of its real-time constraints[9]. Adding time into the system also adds the ability to measure predictability in the system. The level of predictability throughout the system may vary, but with the addition of the physical world it is highly unlikely that any predictions can be completely accurate. Because of the physical part of the system, there is no way to quantify or accurately predict a CPS probabilistically yet.

www.manaraa.com

**1.2.5. Autonomous System.** A CPS needs to be self-sustaining and fault tolerant. There are several reasons that this is important including[9][10]:

- One compromised or lost device should not impact the system greatly.
- Not all entities can be guaranteed to work 100% of the time.
- Ability to recover or at least safely shutdown from faults.
- Communication is a huge expense on the system and is not 100% reliable.

The likelihood of one of the events above is highly certain to happen eventually. If the system cannot handle a failure then the CPS is not really useful. Designing and planning of the system should be done when the system is incepted rather than after it is already built. These methodologies need to be developed, implemented, and used to help guarantee the success of these systems.

## 1.3. VEHICLE AS A CPS EXAMPLE

A specific system chosen as a model problem in this paper, representative of these cyber-physical systems, is the automotive Engine Management System (EMS). EMSs are confidential systems comprised of all the mechanical parts, the computer chips that control them, and the driver. Proper system function ensures that a driver arrives safely from destination to destination by preventing failures of key components, avoiding failures, and providing convenience to the driver. Distributed computing forms a key part of the EMS[11]. The information that is available to any outside observer can disclose characteristics of the vehicle controllers and can manipulate the low level inputs to the EMS. When driving becomes increasingly more automated,

www.manaraa.com

the visual information becomes critical in determining what can be guaranteed, what is in control, or what can be done to force the driver or vehicle into a state that compromises information or safety that should be protected.

Due to the transformation of a vehicle from being a closed system to an open system through the use of wireless technology, and the advancement of the computer systems in a vehicle, there are more vulnerabilities involved with the security to the system. Revealing any information to an outside observer can be a threat to the system, including the driver. Through wireless technology the information can be applied instantaneously to alter settings on the control systems of the vehicle. Thus the vehicle is a system where events internal to the vehicle are high level, and the terrain and observer are low level.

There are several ways that a driver could be attacked once the controller of the vehicle is determined. For one, if it is known that the driver is in control of the vehicle, a road hazard that would cause damage to the car or driver could be set in place to force the driver to react. By contrast, having the knowledge of which systems in the EMS are in control of the car, the vehicle can be forcibly stopped. When the EMS is in control of the vehicle the controller acts predictably in certain situations. Additionally, the EMS controls can be hijacked to cause unwanted vehicle response. These possible attacks make it important to investigate these types of security issues.

This thesis uses different methods of modeling the security of the system and addresses confidentiality of the information and evaluates the availability and integrity of the system. Knowing the state of the EMS exposes the existence of critical system control of the vehicle. Determining these controls and preventing their proper

www.manaraa.com

execution can reveal confidential information about the driver and impact the availability of the system to the driver. The disclosure of confidential information reveals the vulnerabilities to the system. This thesis shows how the confidential information can be compromised.

The main contribution of this paper is to address the confidentiality, availability and integrity of an EMS by analyzing the information flow between the levels of security entities in the system using security models [12] [13] [14] [15] [16]. This is a complementary analysis to the traditional security mechanism of applying access control to provide confidentiality. The vehicle itself does not reveal what its operation is to the outside observer, but velocity, terrain, and obstructions send information to the outside observer. Coupled with the knowledge of the world, this information is adequate to gather enough additional data to reveal knowledge about the driver. Similar analogies can be made with oil/gas (observing flow in a pipe), aircraft control (observing a physical motion change), or power flow along power lines.

www.manaraa.com

# 2. THE ENGINE MANAGEMENT SYSTEM

An EMS is composed of a set of microprocessors that take real time input from sensors distributed around different components of a vehicle. Each of the individual components are known as electronic control units (ECU). A specific electric control unit houses a printed circuit board, micro controller, and EPROMS or flash memory. These units can be reprogrammed and drastically change the function of the particular part of the control system that it is managing. The most common of these units include transmission control units, traction control, brake control, etc. A Controller Area Network (CAN) is used to communicate between the devices. See Figure 2.1 to get a visualization of how each of the components are represented in the system. In composing this system of controllers with a connection to the CAN device, all the components work together to provide the driver with enhanced safety, ease of control, and timely response of the vehicle [17]. There are dependencies that many components have over another and should have predictable responses resulting from the actions of the driver. This causes a complex interconnectedness of the components in the system that must be maintained. When a change occurs on one component, it has a need to be reflected to other components of the EMS. The change relies on the communication to be available and reliable, which takes place on the CAN.

www.manaraa.com

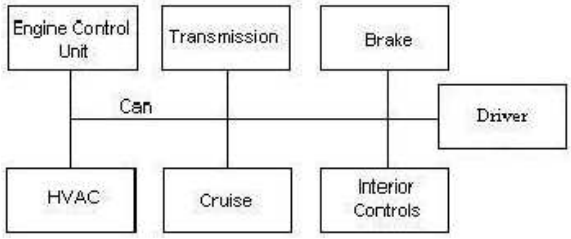


Figure 2.1. Depiction of ECUs in the Vehicle

www.manaraa.com

# 3. APPLICATION OF TRADITIONAL SECURITY

## 3.1. BACKGROUND

Maintaining a secure environment in the midst of nondeterminism becomes one of the most difficult topics of information and computing. The major goals of security include:

- Integrity - Rules that will guarantee data to be correct and complete, including its relations, sources, definitions, and lineage. If the integrity of fundamental components of a system cannot be guaranteed, then the integrity slowly degrades as the amount that is unknown grows.[18]
- Confidentiality - Not disclosing information except to authorized entities.
- Availability - The amount of use that one entity provides to another entity.

In summary, these goals try to achieve a useful secure service or information that can be trusted and easily verifiable. There are various modeling techniques that look at how different aspects protect each of these properties.

In the remainder of this thesis, there are a set of generic actors that hold specific properties. These actors are similar with respect to holding rights, but what sets them apart from each other are which rights each actor is able to hold over another. These terms are defined as followed:

- Subject - An entity that can request operations on other subjects or objects that it holds the proper rights over.
- Object - A related set of functionalities that is classified as an entity, but has no control over another entity.

www.manaraa.com

- Rights - A property a subject can use to operate on, control, read, write, or other defined properties on another subject or object.
- *-Property (read as star property) - A rule that states that a subject can write to an object if and only if the object's clearance is the same or above the clearance level of the subject[19].
- Trusted Subjects - Subjects that are not bound to the *-property.

## 3.2. HARRISON-RUZO-ULLMAN MODEL (HRU)

The HRU model is one of the first attempts to conceptualize a security system at the operating system level. To achieve this there are several basic structures that are used to model a system. The foundation of the Harrison-Ruzo-Ullman Model is built around a matrix of rights. The columns represent the subjects and objects of the system and the rows represent the subjects of the system. Subjects are aspects of a system that should have the ability to interact and change the settings or structure of the system and in some cases act the same as an object. Objects are entities that exist in the system and provide a means to quantify something about the system. For each intersection point there are a list of rights that a subject holds over an object. This matrix is known as the Access Control Matrix (ACM)[20]. An ACM is included in this analysis and will be presented in a later section. The next components of the HRU provides the means to manipulate the ACM. It is necessary for there be a slight difference between the rules that apply to subjects and objects to maintain their differences. In consequence, these rules are also responsible for the leakage of rights. An overview of the rules follow:

www.manaraa.com

- Create Subject - If the subject does not already exist in the ACM then a new subject will be created with no rights for any pairing of subjects and objects. This rule adds another row and column to the matrix.[20]
- Create Object - If the subject does not already exist in the ACM, a new object will be created with no rights with any paring of a subject. This adds a column to the matrix.[20]
- Delete Subject - Removes the subject's footprint from the ACM if the subject exists and does nothing if it does not.[20]
- Delete Object - Removes the row from the ACM if the object exists but does nothing if it does not.[20]
- Enter Right - Enters a right (r) if the subject and object do exist.[20]
- Remove Right - Removes the right (r) from the subject, object pair if they exist but does nothing if that combination does not have that right.[20]

The final component of the HRU model is the formulation of commands. Commands are the series of basic rules executed on the ACM. To give more meaning to the commands, HRU will allow for conditionals to only check if a right is contained in a specific pairing. It is NOT allowed to check for the absence of a right. A Turing machine can be directly mapped into a HRU model. Doing so reveals that as long as the create primitive is not removed, the safety model is undecidable [21]. In summary, the strength of this model is that the commands give a good visualization of the transfer of rights as they are executed in the model. The weaknesses are that it is both undecidable and, thus, cannot be computationally verified.

### 3.3. BELL-LAPADULA MODEL (BLP)

The Bell-LaPadula Model is a modeling strategy that enforces access control. It takes a different approach of modeling the system by implementing clearance levels on subjects. Each time the system wants to change, properties of the system must be

www.manaraa.com

satisfied before a change of the state is allowed. BLP uses the Subjects and Objects paradigm, but defines them a little differently than the previously mentioned models. A subject can only have read access at or below its level or write access over other subjects/objects at or above it's level. Again, objects can have no rights over other objects/subjects.[19]

The two important rules that build the structure for this model are:

- Simple Security Condition - A subject can read an object if and only if the security clearance of the object is either below or at the same security clearance.[19]
- Discretionary Security Property - An access matrix can be used to enforce other rights to trusted objects.[19]

These rules are formed to maintain the confidentiality of the system. In order for the system to remain secure from the initial state, a theorem was developed. The theorem states, that if each transition from a secure initial graph maintains the simple security condition and the *-property, then the system will always remain secure; transitions occur when a request is made to get some sort of rights over another subject or object. When this request is made, a decision occurs, and the result of this decision can be yes, no, illegal, or error. The only decision that will allow for a change in the system is yes. To maintain the security throughout, the three fundamental security properties (confidentiality, availablility, integrity) must be held in the system at all times.

As mentioned before, the simple security theorem must be maintained; if this fails the decision is no; the parameters of the request must also be true, if this fails then the decision is illegal.[19]

www.manaraa.com

For a subject to write to a lower classified subject/object it must drop down to the same level which has a negative impact to the system. At the point that a subject lowers itself, it can pass the information that it knew when it held the higher classification level. This information can then be passed, causing leakage in the system.[19]

### 3.4. INFORMATION FLOW

Information flow conceptualizes how information moves from one point to the other. There are two major components that have to be maintained in order to keep data secure. The first goal is to maintain confidentiality and also to prevent data from flowing to an unauthorized entity. Analyzing data as it passes around a system shows how data can change and impact a system. This flow can reveal characteristics about it that may not be apparent in the other models and also displays how information can be leaked without violating any rules or how data can influence state of the system.

Certain systems have a probabilistic state determination that can reveal what state the system can be in with a given set of information. Other systems can have deterministic state transitions that make it very easy to know what is occurring in the system, even knowing the states of individual components.

Information flow is defined with several rules. An entity $\alpha$ is said to transfer information to another entity $\beta$ with the following conditions[22]:

- $\alpha$'s current state is in a state that allows it to transfer information to $\beta$.
- $\beta$'s current state can accept information coming from $\alpha$.

www.manaraa.com

- $\alpha$'s security level is at most $\beta$'s least secure level.
- $\beta$'s security level is at least the same security level as $\alpha$'s highest security level.

The first two rules are defining what it means to transfer information, and the last two define what it means to be secure with respect to the transfer of data.

www.manaraa.com

# 4. PROBLEM STATEMENT

In the EMS, the control is maintained by both the driver and ECUs controllers. The distinction of the two control types should be kept confidential. This paper follows their definition of confidential information (shown in Table 4.1) to analyze the information flow in the EMS.

Table 4.1. Confidential Information in EMS

| **Data** | **Type** | **source** | **Function** |
|---|---|---|---|
| Driver | | Chief operator of the car | |
| Cruise Control | ECU | Automatic control | |
| Crash Avoidance System | ECU | Automatic control | |
| **Control** | **Type** | **source** | **Function** |
| Sensor Information | Digital (CAN) | Sensor Network | Sensor information giving input into the EMS |

The EMS is made up of two confidentiality levels (shown in Table 4.2). In the high-level security domain, communication is done through a CAN bus and is responsible for the control of the vehicle.

At the low level, the speed information causes implicit communication due to visual queues. The unobstructed view of the car reveals information about the car such as speed, which makes this the implicit communication. The failure of confidentiality occurs when an observer from the low level security domain observes or deduces information from higher security levels. In order to demonstrate the problem clearly, the following assumptions are made:

www.manaraa.com

Table 4.2. Security Levels in EMS

| Security Level | Security Entities | Reasons |
|---|---|---|
| High-Level | EMS | Responsible for aiding the control of the vehicle for the driver |
| | Driver | Relays input to the EMS by the use of actuators Levers and buttons |
| Low level | Physical | Everything exterior to the vehicle |
| | Observer | The entity watching information flow |

**Assumption 1:** The messages sent by the one device to another is legitimate and correct. (The security of each component is outside the scope of this paper.)

**Assumption 2:** The CAN is secure.

**Assumption 3:** Automatic control systems all operate according to real world design specifications.

**Assumption 4:** Automatic control's purpose is only to maintain speed or keep the driver safe.

Assumptions 1, 2, 3, and 4 are trying to define the problem scope of this paper, which is confined to investigate the security of the CPS.

www.manaraa.com

# 5. METHODOLOGY

## 5.1. HARRISON-RUZO-ULLMAN MODEL

### 5.1.1. Overview.

- Read (R) - Read messages from a subject.
- Write (W)- Ability to write messages to a subject.
- Own (O)- Denotes that a subject has ownership over another subject.
- Control (C)- Has commands that can be executed on a subject.
- Execute (E) - Can request that some functionality be performed.

At the vehicle level, the Access Control Matrix (ACM) that was developed for this model can be seen at Table 5.1. Each component that needs to communicate to another component has read and write access over the CAN bus, since the CAN bus is the connecting piece for the communication of the various components. This model exemplifies the rights that each component should have over another for proper operation of the vehicle. There are some cases where the state in one subsystem should notify another subsystem of its state (W). In most cases, there is a need for one controller to have influence over another controller (R,W).m level, the ACM developed for the vehicle example can be seen in Table 5.2. This depicts how the high-level components of the system are able to interact with each other. At the conceptual level, it does not make sense to simply add the observer into the ACM generated in Table 5.3. The vehicle itself is the combination of all the components

www.manaraa.com

that make up Table 5.3, excluding the driver. Having the observer able to interact with the specific components that compose the vehicle does not capture the logical jump of the observer's interaction to the system. To the observer the vehicle does not exist as a combination of the components, but as a complete unit of these components. This leaves the vehicle example with three distinct subjects and objects.

Table 5.1. List of Events used in the Terrain Examples

| **List of Rights $\alpha$ has over $\beta$** | **Description** |
|---|---|
| O,R,W | $\alpha$ owns and can notify $\beta$ but can not change any execution properties of $\beta$. |
| R,W | $\alpha$ can read and write messages to $\beta$. |
| R, W, E | $\alpha$ can read and write information to $\beta$ as well as execute commands on $\beta$. |
| W | $\alpha$ can only write information to $\beta$. |
| E | $\alpha$ can execute actions on $\beta$, but can't read any additional state information over $\beta$. |
| C,E | $\alpha$ can control and execute changes on $\beta$. |
| O, C, R, W, E | $\alpha$ has all possible rights over $\beta$, component has full control possible over $\beta$. |
| C, R, W | $\alpha$ can control $\beta$ and read and write information to and from $\beta$. |
| C, R, W, E | $\alpha$ has full control over the system $\beta$ except for owning the system. Relies on the service of $\beta$ for proper operation of $\alpha$. |
| O, C, E | $\beta$ reacts to the commands given to it by $\alpha$, but doesn't need to be aware of the state of $\beta$ to operate. |

Table 5.2. ACM of the Physical World in the Vehicle Example

| HRU Applied to the System of the Car Example | | | |
|---|---|---|---|
| | Driver | Observer | Vehicle |
| Driver | R, E, C | R, W, E, O | W |
| Observer | R, W | | R, W, O, E |

www.manaraa.com

Table 5.3. ACM for the Vehicle Example

| HRU Applied to the Vehicle of the Car Example | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CAN Bus | ABS Control | Body Control | HVAC Control | Transmi | Engine Control | Gas | Luxury Control | External Lights | Driver | Cruise Control | Observer |
| Can Bus | O, R, W | R, W | R, W | R, W | R, W | R, W | R, W | R, W | R, W | R, W | R, W | |
| ABS Cont. | R, W | O, R, W, E | | | C, R, W | W | C, R, W | | | R, W | | R, W |
| Body Cont. | R, W | | O, R, W, E | | | R, W | R, W | R, W | | | | R |
| HVAC Cont. | R, W | | | O, C, R, W, E | R, W | | | R, W | | R, W | | |
| Transmission | R, W | R, W, E | | | O, C, R, W, E | R, W, E | R, W | | | R, W | R, W | |
| Engine Control | R, W | R, W, E | R, W, E | C, R, W, E | C, R, W | O, C, R, W, E | C, R, W | R, W, E | C, R, W | R, W | C, R, W | R |
| Gas | R, W | | W | W | C, R, W | R, W | O, C, E | | | | R, W | R, W |
| Luxury Control | | | | R, W, E | | | | O, C | | R, W | | R |
| External Control | R, W | | | | | | | | | R, W | | R |
| Driver | | E | E | E | E | E | C, E | E | E | O, C, E | C, E | R, W |
| Cruise Control | | | | | | R, W | | | | | O, C, R, W, E | R, W |
| Observer | R | R | R | | | R, W | R | | R | R, W | R, W | R, W, O, E, C |

**5.1.2. Remarks.** While the HRU model and the resulting ACM are invaluable for depicting the rights that one entity can have over another entity, it is easy to be overwhelmed by the quantity of information that is provided by the matrix. Because of this, the ACM is not the most meaningful way to approach the analysis of the system, but it does provide the fine details of the rights. Visually, a hierarchy of how the entities in the system interact with each other can provide a better a portrayal of those interactions. The Bell-LaPadula model captures a more abstract depiction of the system, but captures much of the same details as the ACM.

## 5.2. BELL-LAPADULA MODEL

**5.2.1. Overview.** The model in Figure 5.1 model depicts what access is allowed between the different classification levels defined for the car example. The

www.manaraa.com

three are:

- Critical Components - Devices that are critical for the operation of the car. The severity of any of these failings will increase the occurrence of an event that causes a negative outcome to the operation of the vehicle. These components are embedded hardware.
- Operational Components - Objects that greatly impact the operation of the car. Failure of one of these systems will negatively impact the car, but will not always be destructive to the car's operation. This level is where the high-level devices reside. In the car example, the driver acts as a high level system since he has the knowledge of the system's goal, while the critical components do not.
- Optional Components -Do not impact the operation of the car. Failure of one of these objects is just an inconvenience.
- Observable Components - The entities of the system that utilize the public channel in order to interact with the system.

The Critical section contains the most important devices, which are the devices that must not fail. If any of the components that make up the Critical section fail, the system will be inoperable. Since the subjects that are contained in this field can read, write, and execute each other, in order to maintain the integrity these components need to be included at the same integrity level. The Critical objects will be able to write to all the subjects/objects below their integrity level while those below can only read up which is also the desired system functionality.

The Operational Components are important for the execution of the system. Indirectly, in the car example, the Operational Components have the capability to influence the Critical level components. With this model, the operational components can request to set the Critical Components at a specific state. The Critical Components can either act on them or not, but when a car does not act on the requests of

www.manaraa.com

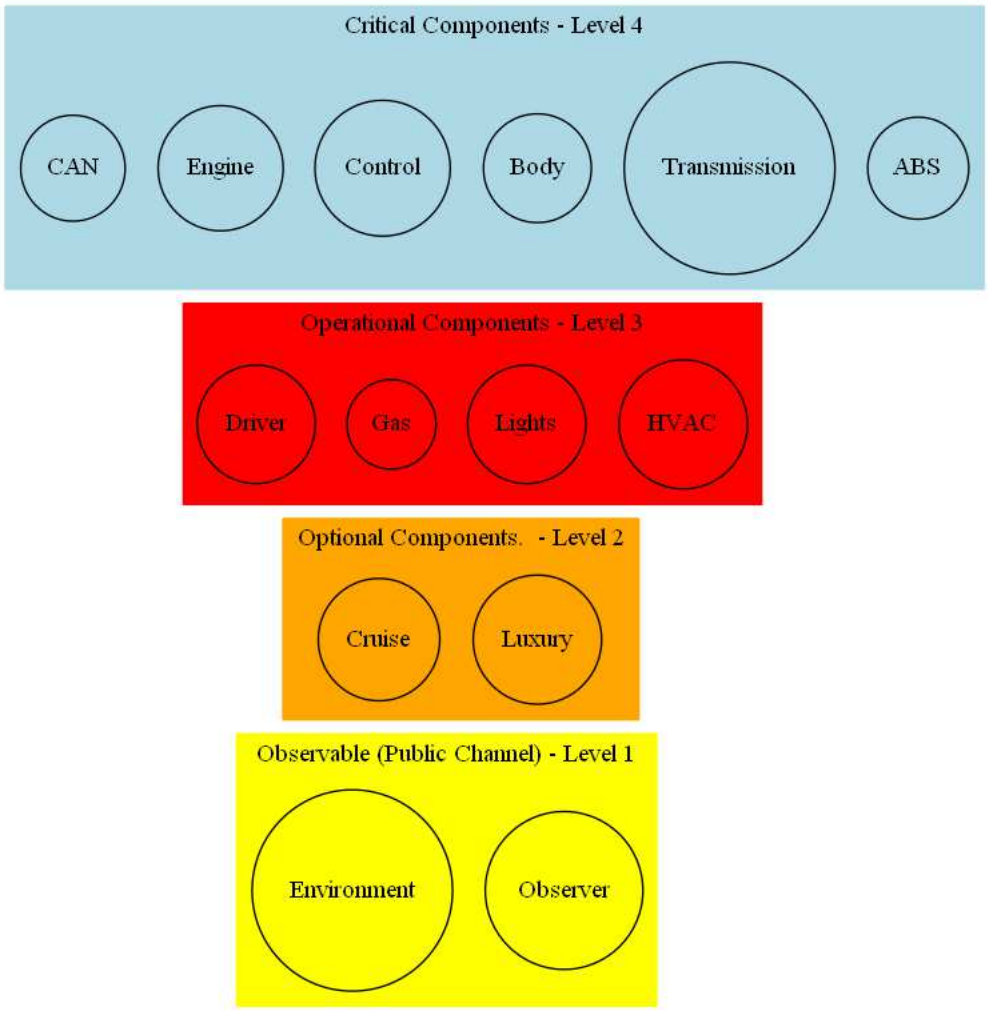


Figure 5.1. Multi-Level Security Model of the Vehicle Example
The Multi-Level Security Model organizing the components based on their integrity and confidentiality categories.

the operational equipment the driver's safety is risked. This creates a visible representation of what the components are doing even though they should be hidden from the Operational Components. In the case of the driver, because the entity knows additional knowledge about what the components of the vehicle must do, understands how the car is interacting with the world.

www.manaraa.com

The Optional components in the car also have impact on how the car functions. If a driver sets the cruise control to maintain speed, the cruise control manipulates the amount of the gas that flows into the engine.

The Observeable Components make up the entities that exist in the public channel. The main importance of the components at this level is that they are able to read the freely available information that the vehicle portrays by its interaction with they physical world. In some cases the observer, which resides in the Observable Components, can interact with the system by forcing the vehicle to react to the physical world.

The car's design must enforce responsiveness and predictability to the Operational Components and the Optional Components of the car. Since the car's operation cannot be hidden to the driver, it works against the enforcement of a car as a secure system. The vehicle cannot prevent the passage of information to the Observable Components to the system.

**5.2.2. Remarks.** The BLP model gives a great visualization of how the system should ideally be composed. Using the ACM that was generated in Table 5.1, the entities that reside in the Critical Components level generally have R, W, E and optionally C. The grouping of rights that are common in the ACM provide a part of what is needed to build the BLP model. In addition to those rights, the functionality of each component and its importance to the system needs to be captured in this model. Since some of the rights that exist in the ACM and what is presented in the BLP can show violations of the foundation that the BLP is trying to build and is why the discretionary security property exists. Refer to the Figure 5.1 to see this model.

www.manaraa.com

The disagreement between the rights in the ACM and the rules that the system needs to follow to maintain its multi-level hierarchy needs deeper analysis. Information flow can provide answerers to some of these inconsistencies.

## 5.3. INFORMATION FLOW

**5.3.1. Overview.** Information flow in a car is largely a translation of one piece of information into a more detailed expansion of information. The actions that a driver takes eventually reach the lowest level (most critical) of the information hierarchy, and happens without the driver knowing exactly what is going on at that level.

The Bell-LaPadula model sets the organizational ground work of how the components are separated into their repsective classification levels. A breakdown of the entities with their relation to the BLP are are listed below:

- Public channel (Observable Components) - includes the observer and anything external to the car. While this item is not included on the multi-level security model in Figure 5.1, it would reside at the bottom of the hierarchy.
- Driver (Operational Component) - the operator of the vehicle
- Core (Critical Components) - Includes major components of the system such as the brake, transmission, engine
- Support (Critical Components) - Components that compose some core functionality.

A vehicle is made up of several rules that enforces the way that information passes through the system. These rules have been defined as follows:

www.manaraa.com

- Information can flow from the car to the driver.
- The driver can send information to the core components by interacting with the available actuators.
- Core components federate messages to the supporting components to fulfill the driver's request.
- Support responds to the Core about the conditions of the components composing the support elements.
- Core sends information to the car about the happenings in the Core.
- Car gives a display of information to the driver about the state of the car.
- The car can notify the public channel of happenings within the car.

An example of information flow in a vehicle could consist of the action of the driver applying pressure to the accelerator. This action sends a relay of information to all entities in the system. The message the driver sends to the core components is to increase the car's speed. The core components react by keeping the car in the proper gear and the engine to fire at its optimal operating condition. The engine ensures the parts of the supporting components about their functionality. The transmission keeps the drive at the proper gear to ensure engine safety and translation of the power coming from the engine to the road. A final message is then sent out to the external part of the car which is speed. This basic knowledge of how a car operates would tell the observers external to the car that the driver pushed the gas pedal.

Even while each component operates in a safe, secure manner, information flow as depicted in the vehicle example shows how confidentiality can be lost in a system. These topics will be further discussed in later sections. A car is very hard to protect because safety and expectations of its operation have strict requirements. Those requirements are to protect the driver, the other drivers, be reliable, and move a

www.manaraa.com

person from one place to the other in a timely manner. To ensure all the requirements are met, the car becomes predictable due to the the information that can be deduced from the observable properties.

Information flow shows the value of investigating how outside knowledge influences the knowledge of known information of the system. If the observer had no idea how a car operated, then seeing a car speed up would not have much meaning to them. This leads to the next question, how much previous knowledge does one need in order to make meaning out of observable properties in a system? This is discussed in the next section.

**5.3.2. Remarks.** Including the analysis of the information flow creates the foundation of a CPS's security when approaching and looking at the system. This analysis not only shows how information travels, but also helps to determine at what level the information should be protected to satisfy the requirements of the system. Even though information can pass securely from the correct entity and to the correct target, the path that it takes may give the opportunity of learning something about the information or the entities as it passes. In the car example, it is apparent that information is getting passed from the hardware of the car back to the driver. While this movement is intentional, some of the unintentional data movement is from the hardware of the car to an outside observer.

www.manaraa.com

To continue this analysis, the remainder of the thesis will focus on analyzing what the observer can learn about the information passing between the vehicle and the physical world.

## 5.4. MODELING DISCUSSION

The modeling concepts, HRU, BLP, and Information flow, define the rules of how the entities can interact with each other and how specific information can pass between one component to the other. These models are well-suited for defining a way to see how components interact with each other, the path that the information flows, and develop a hierarchy of the entities in the system.

HRU specifically lays out the rules that define exactly what type of rights one entity can have over another. This model is the most granular of the three used thus far. The leakage of rights occurs due to the chain of commands that one entity can pass to another, which sometimes results with a right ending up on an entity that should never have been there.

BLP helps to keep the hierarchy of how the information flow should be organized. This maintains the integrity of the system by regulating the process in which the components and read and write to each other. Violations are easily identifiable by using this model to define the group of read and write rules that exist in the system. This model also has some flexibility by allowing a specific matrix of rights to exist between entities to allow for exceptions.

Information Flow keeps the passage of information. This model ensures that the information passes from the critical components to the operational components

www.manaraa.com

and the optional components are only the data that each of the entities need to see. This model has the ability to show where the entities that should not see the data are able to observe it.

What these models do not cover is the implication of distinguishing the events of a higher level with the information that can be freely accessed by an entity. In the vehicle example, because the observer is able to piece knowledge about vehicle operation and the information that the vehicle gives the observer, there is information that should be protected that is not. This will be covered in the remainder of the thesis.

www.manaraa.com

# 6. SECURITY POLICY EVALUATIONS

This section is an expansion of a previously published paper[23]. The inference of confidential information from the observable information flow has high potential to cause critical information leakage; therefore, the vehicle's information flow should be carefully analyzed. Various security models that analyze multi-level security system behavior from the access control or execution sequence perspective have been discussed for decades to address the information flow problems in the defense community [14]. However, most of the related publications [12] [13] [14] [15] [16] have not been directly applied to a CPS due to the complexity. It is worthwhile to apply these models due to the significance to the critical infrastructure. The following models will be used to show these properties.

Each of the models deal with the principles of low level and high level events. Low level events are low priority events that are visible to low level users. In a secure system, low level events will never have direct control over higher priority events. High level events are events that have a high priority of execution. Depending on a particular model, high level events may or may not have direct interaction with a lower level entity. When evaluating traces, which are a set of events that contain both high-level and low level events, a purge function may be used as a systematic way to create the visible trace at a specific security level.

**Nondeducible Model**: a system is considered nondeducible secure if it is impossible for a low level user, through observing visible low level events, to deduce anything

www.manaraa.com

about the sequence of inputs made by a high-level user. In other words, system is nondeducible secure if the low level observation is compatible with any of the high-level inputs [12] [14].

**Noninference Model**: a system is considered noninference secure if and only if for any legal trace of system events, a trace resulting from a high-level purge is still considered a legal trace. [14]

**Noninterference Model**: a system is considered noninterference secure if and only if for any legal trace of system events, a purge at the high-level does not leave a clue to which high-level events are taking place as low level events occur. [14]

The EMS system is a multi-level security structure. To give a clear analysis of the information flow of EMS, the security models defined above are used in this thesis. However, the noninference model is applied for the purpose of analyzing the information flow of a vehicle interacting with the world and in this case the road. This is done to illustrate the cases where low level inputs result with some high-level response in the controller of the computer system. Nondeducible security models are used to analyze the system where high-level outputs are observable. According to [12], if an entire system is nondeducible secure, then no low level user of that system will ever learn any high-level information through the system.

www.manaraa.com

# 7. ANALYSIS OF EMS USING SECURITY TECHNIQUES

## 7.1. TERRAIN EXAMPLE

Using the appropriate security models will show how the EMS can divulge information to the low security level. A set of examples will be used to illustrate the information flow in the system. The analysis takes a look at the vehicle as a composition of both the driver (manual inputs) and the automated control provided by the EMS that includes units such as cruise control. In one example, a vehicle is traveling along a flat land which is a selective case from a typical system. The second view uses a general case by incorporating a terrain that is hilly. The scenarios will match in all snapshots restricting the view of the car in the system. See Figure 7.1 for clarification of how the events are portrayed and Figure 7.2 for a depiction of how information flows from the vehicle to the observer. The comparison of the two scenarios' information flow will allow a further investigation of the security of the system.

In this example the components that make up the scenario consist of a vehicle that includes a driver, automatic control, and the physical world that the vehicle interacts with. Refer to Figure 2.1 in a previous section for the depiction of the components. Each of these are grouped into a set of security entities. See Table 7.1 for the events used in the scenario traces. In Tables 7.2, 7.3, and 7.4 are traces for the multiple cases of the same scenarios, but interestingly reveal different properties.

www.manaraa.com

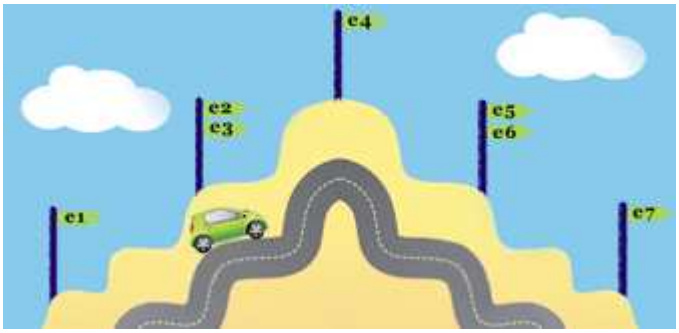


Figure 7.1. Comparison of Events on Differing Terrains

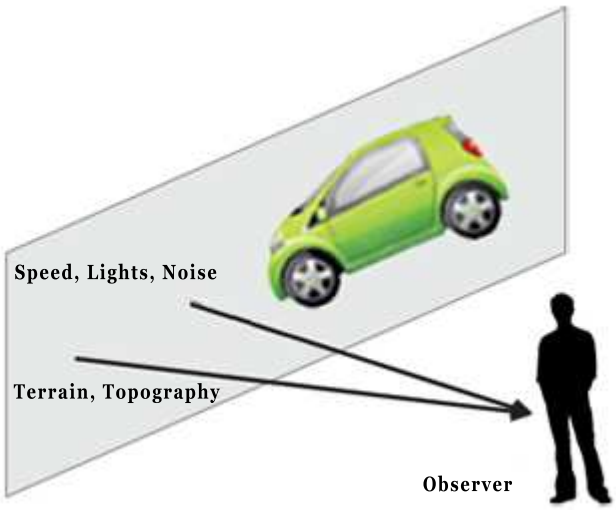


Figure 7.2. Information Flow Diagram of an EMS

www.manaraa.com

Table 7.1. Events of the Terrain Examples

| Notation | Description |
|---|---|
| l1 | Initial speed |
| l2 | Speed reduction |
| l3 | Speed Increase |
| h1 | Driver maintains gas flow |
| h2 | Cruise control maintains speed |
| h3 | Driver increases gas flow |
| h4 | Cruise control increases speed |
| h5 | Driver slows current gas flow |
| h6 | Cruise control decreases speed |
| h7 | Driver resumes original gas flow |

Table 7.2. Trace list for the Standard Cruise

| Scenario | Trace |
|---|---|
| Traveling Up Hill | |
| Maintain Speed | $\{l_1h_3l_1\}$, $\{l_1h_4l_1\}$ |
| Speed Reduction | $\{l_1h_1l_2\}$, $\{l_1h_2l_2\}$, $\{l_1h_5l_2\}$,$\{l_1h_6l_2\}$ |
| Traveling Down Hill | |
| Maintains speed | $\{l_1h_5l_1\}$, $\{l_1h_6l_1\}$ |
| Increases speed | $\{l_1h_1l_3\}$, $\{l_1h_2l_3\}$, $\{l_1h_3l_3\}$, $\{l_1h_4l_3\}$ |
| Traveling on Flat Surface | |
| Maintains speed | $\{l_1h_1l_1\}$, $\{l_1h_2l_1\}$ |
| Increases speed | $\{l_1h_3l_3\}$ |
| Decreases speed | $\{l_1h_5l_2\}\}$ |

www.manaraa.com

Table 7.3. Trace list for the Perfect Cruise

| **Scenario** | **Trace** |
|---|---|
| Traveling Up Hill | |
| Maintain Speed | $\{l_1h_3l_1\}$ |
| Speed Reduction | $\{l_1h_1l_2\}$, $\{l_1h_5l_2\}$ |
| Traveling Down Hill | |
| Maintains speed | $\{l_1h_5l_1\}$ |
| Increases speed | $\{l_1h_1l_3\}$, $\{l_1h_3l_3\}$ |
| Traveling on Flat Surface | |
| Maintains speed | $\{l_1h_1l_1\}$, $\{l_1h_2l_1\}$ |
| Increases speed | $\{l_1h_3l_3\}$ |
| Decreases speed | $\{l_1h_5l_2\}\}$ |

Table 7.4. Trace List for the Random Cruise

| **Scenario** | **Trace** |
|---|---|
| Traveling Up Hill | |
| Maintain Speed | $\{l_1h_3l_1\}$, $\{l_1h_4l_1\}$ |
| Increases speed | $\{l_1h_3l_3\}$, $\{l_1h_4l_3\}$, |
| Speed Reduction | $\{l_1h_1l_2\}$, $\{l_1h_2l_2\}$, $\{l_1h_5l_2\}$,$\{l_1h_6l_2\}$ |
| Traveling Down Hill | |
| Maintains speed | $\{l_1h_5l_1\}$, $\{l_1h_6l_1\}$ |
| Increases speed | $\{l_1h_1l_3\}$, $\{l_1h_2l_3\}$, $\{l_1h_3l_3\}$, $\{l_1h_4l_3\}$ |
| Traveling on Flat Surface | |
| Maintains speed | $\{l_1h_1l_1\}$, $\{l_1h_2l_1\}$ |
| Increases speed | $\{l_1h_3l_3\}$, $\{l_1h_2l_1\}$ |
| Decreases speed | $\{l_1h_5l_2\}$, $\{l_1h_2l_1\}$ |

For each of the following proofs the event that the car changes or maintains velocity is treated as an independent set of traces. If one of the traces fails to hold the property, for standard cruise control and the driver then it fails for the entire scenario.

www.manaraa.com

### 7.1.1. Nondeducibility.

**Theorem 7.1** *Vehicle operation is nondeducible secure when traveling up hill for standard cruise control and a driver or the random cruise control and a driver.*

**Proof** The EMS is a nondeterministic system which is built from the traces in Table 7.4: For each trace the first and third event are low level events and the second event is a high level input into the system.

**Standard Cruise:** This scenario is built from the traces in Table 7.2. The vehicle as it travels up the hill is nondeducible secure because for any trace the low level outputs are identical regardless of the controller of the vehicle. The traces are either $l_1 l_1$ or $l_1 l_2$. Temporarily ignoring all the high level actions it can be seen that there are multiple cases that show that the vehicle either speeds up, slows down, or maintains speed. Because of this fact, there is no event trace that leads to a unique set of outputs to the observer.

**Random Cruise:** This scenario is built from the traces in Table 7.4. Traveling up hill the random cruise is able to perform the same actions as a driver, so for each case of the car with a change in velocity or maintenance of the velocity the controller is nondeducible secure. The random perturbations of the velocity are able to hide the specifics of the controller since the random cruise exhibits the same behavior as the driver.

**Theorem 7.2** *Vehicle operation is not nondeducible secure when traveling up hill for a Perfect Cruise and a driver.*

www.manaraa.com

**Proof** This scenario is built from the traces in Table 7.3. Because the perfect cruise does not allow for any deviation from the set speed, it is more revealing of the controller of the vehicle. The driver has the ability to change speed or due to a lack of awareness changes speed, this leads to a divulgence of information and violates the nondeducibility property.

**Theorem 7.3** *Vehicle operation is nondeducible secure when traveling down hill for standard cruise control and a driver or the random cruise control and a driver.*

**Proof** The EMS is a nondeterministic system and for a given low level inputs and controllers there are several representative traces. For each trace the first and third event are low level events and the second event is a high level input into the system.

**Standard Cruise:** This scenario is built from the traces in Table 7.2. The vehicle as it travels up the hill is nondeducible secure because for any trace the low level outputs are identical regardless of the controller of the vehicle. Ignoring all the high level actions it can be seen that there are multiple cases that show that the vehicle either speeds up, slows down, or maintains speed. The traces are either $l_1 l_1$ or $l_1 l_3$. Because of this fact, there is no event trace that leads to a unique set of outputs to the observer.

**Random Cruise** This scenario is built from the traces in Table 7.4. The random cruise will vary the speed in such a way that it will average out at the target speed. This is feasible if the cruise has as much control over the car as the perfect cruise but alters the velocity enough to make it reasonable at different discrete events that the velocity can vary.

www.manaraa.com

**Theorem 7.4** *Vehicle operation is not nondeducible secure when traveling down hill for a Perfect Cruise and a driver.*

**Proof** This scenario is built from the traces in Table 7.3. Traveling down hill reveals the controller much in the same way as going up hill. Since it is unreasonable for the car to change its speed while on a perfect cruise control it reveals that the driver is in control of the vehicle. There is a unique trace for any sort of speed variation, so this violates the nondeducibillity property.

**Theorem 7.5** *Vehicle operation is not nondeducible secure when traveling across a flat terrain for the standard cruise or random cruise and a driver, but it its nondeducible secure for a random cruise and a driver.*

**Proof** The EMS is a nondeterministic system represented by a trace of events. For each trace the first and third event are low level events and the second event is a high level input into the system.

**Standard Cruise:** Refer to the traces in Table 7.2. The vehicle as it travels across a flat surface is not nondeducible secure because for any trace the low level outputs are unique. In the case that the vehicle speeds up ($\{l_1 h_3 l_3\}$) or slows down ($\{l_1 h_5 l_2\}$), there is no other way to explain these actions than to say that the driver is operating the vehicle in these cases. A cruise control would not act in this manner.

**Perfect Cruise:** Refer to the traces in Table 7.3. Traveling across the flat terrain with a perfect cruise control is much the same as a standard cruise control. Since the standard has no problem just maintaining the speed, the driver as a controller is revealed.

www.manaraa.com

**Random Cruise** Refer to the traces in Table 7.4. The random cruise has the most substantial effect on the case of the flat terrain. For both the standard cruise and the perfect cruise, it is not nondeducible secure for the reason that it is unreasonable to have any differences of speed from the set velocity. Adding a cruise that will randomly increase or decrease speed in this scenario adds traces into the system that match with traces of the driver. If the cruise were random in this scenario, it would be impossible to distinguish whether the driver is in control of the vehicle or the cruise control.

**Discussion** The results from these examples reveal a powerful source of obfuscation even in the public channel. When the low level outputs correlate with a low level input at the physical level, this being the type of terrain, the controller of the high level inputs are hidden from the observer. This comes down to being able to explain the characterization of events as it corresponds with a cause and effect in the physical world. For example, as the vehicle travels up the hill, the event of the car slowing down from its initial velocity can be explainable in two ways. The driver did not maintain the proper amount of gas flow to give the car enough force to make it up the hill or the cruise control could not compensate for the steep grade of the hill. It follows that there is a trace for both the driver and the cruise control that allow the car to slow down while traveling up hill.

Adding a perfect cruise in the system shows how any deviation in speed from the target speed reveals the driver in most cases. Having the variation of speeds, as added by the random cruise, is beneficial to the system as a whole since the cruise

www.manaraa.com

control exhibits similar behavior as a human controller. Using a randomized cruise ensures the traces of the cruise control matches the traces of the driver. For most scenarios the randomized cruise keeps the driver from being distinguishable from the cruise control. Generally a driver does not maintain a constant speed over all terrains while a perfect cruise control has the ability to do this.

### 7.1.2. Noninference.

**Theorem 7.6** *Vehicle operation is not noninference secure for a driver and standard cruise control when maintaining speed and traveling down hill or traveling up hill, but is noninference secure for a driver and standard cruise control when slowing down and traveling up hill or speeding up and traveling down hill.*

**Proof** : The EMS is a nondeterministic system which is built from the following traces: For each trace the first and third event are low level events and the second event is a high level input into the system.

**Standard Cruise:** The vehicle as it travels down the hill is noninference secure because for any trace the low level traces are identical regardless of the controller of the vehicle. See Table 7.2 for a reference of the possible traces. After a purge of the high level events, in the case of maintaining speed the resulting trace is $\{l_1 l_3\}$, while increasing speed the resulting trace is $\{l_1 l_3\}$. In terms of traveling up hill the resulting traces is $\{l_1 l_2\}$, while reducing speed the resulting trace is $\{l_1 l_2\}$. It is not noninference secure with respect to there being a controller when the vehicle is maintaining speed, but this does not reveal who is in control of the vehicle. It is noninference secure in the case that the car is increasing speed while traveling down

www.manaraa.com

hill and decreasing speed while traveling up hill when the high level input is purged from the traces.

**Perfect Cruise:** See Table 7.3 for a reference of the possible traces. After purging the high level events with respect to the perfect cruise, it follows much in the same way as the standard cruise. Once the high level events are purged, the low level trace would result with $\{l_1 l_3\}$ in the down hill trace and $\{l_1 l_2\}$ for the up hill trace. Having a controller violates the noninference property.

**Random Cruise** Including a random cruise in this environment keeps the vehicle controller noninference secure. See the table 7.4 for a reference of the possible traces. The fact that there is a controller is still visible to the observer, but the controller of the vehicle cannot be determined. This is caused by there being no unique mappings to an individual controller after the removal of the high level events.

**Theorem 7.7** *Vehicle operation is not noninference secure for a driver and standard cruise control when traveling across a flat surface and increases or decreases speed, but is noninference secure for a driver and standard cruise control when the speed is maintained.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and third event are low level events and the second event is a high level input into the system. When the vehicle maintains speed the observer cannot tell whether it is the driver of the cruise control that is in control of the vehicle. In any case that the vehicle changes speed, it is obvious that it is the driver that is in control.

www.manaraa.com

**Standard Cruise:** The vehicle as it travels along the road will never increase or decrease speed while the standard cruise is operating. After a purge of high level events, the resulting trace for the car while maintaining speed is $\{l_1 l_1\}$.

**Perfect Cruise:** In this case the perfect cruise acts the same as the standard cruise.

**Random Cruise** The random cruise is less revealing about regarding the non-inference property. Applying the purge of the high level events it remains that the resulting trace is either $\{l_1 l_3\}$, $\{l_1 l_1\}$, or $\{l_1 l_2\}$. It is noninference secure in terms of the controller, because it has just as much validity of not maintaining speed as the driver does.

**Discussion** The outcome of this analysis shows another natural obfuscation of high level events when the low level events correlate with the physical design of the system. For example, if the car traveling down the hill has the high level action removed from the trace, it is reasonable for the car to slow down or remain a constant speed. In order to determine the controller, it must be deduced from the system. Although, on the flat terrain the controller of the vehicle can be inferred because if the vehicle either speeds up or slows down then it is not the cruise control that is managing the operation of the vehicle.

The perfect cruise does little to improve the results as is expected and the random cruise does improve the ability to infer which controller is active in the system. With its addition, it masks the controller by adding additional traces into the system that guarantee that there is no unique mapping of traces at the low level.

www.manaraa.com

### 7.1.3. Noninterference.

**Theorem 7.8** *Vehicle operation is noninterference secure for a standard cruise and a driver when traveling across a flat surface and decreases speed.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and third event are low level events and the second event is a high level input into the system. Looking at only the low level events, the events of the vehicle slowing down are $\{l_1 l_2\}$. Because the observer sees the vehicle slowing down, there are no indicators that there is a controller of the vehicle, which satisfies the property of noninterference.

**Driver:** The driver can sometimes fail to keep the target speed and causes the speed to decrease.

**Standard Cruise:** Having a standard cruise in operation would not satisfy the noninterference property, but since it does not allow the vehicle to slow down on a flat surface, it is obvious it is not in control of the vehicle.

**Perfect Cruise:** In this case the perfect cruise acts the same as the standard cruise.

**Random Cruise:** Adding a random cruise into the system also maintains the noninterference property when the random cruise reduces the speed of the vehicle on a flat surface. If the random cruise allows the vehicle to speed up or slow down, it would violate the noninterference property.

**Theorem 7.9** *Vehicle operation is noninterference secure for a standard cruise and a driver when traveling up hill and decreases speed.*

www.manaraa.com

**Proof** : The EMS is a nondeterministic system where for each trace the first and third event are low level events and the second event is a high level input into the system. The low level trace of the vehicle slowing down is $\{l_1 l_2\}$. Because the observer sees the vehicle slowing down, there is no indicators that there is a controller of the vehicle, which satisfies the property of noninterference.

**Driver:** The driver can sometimes not compensate enough to maintain speed going up a hill and can end up slowing down.

**Standard Cruise:** The standard cruise does not always compensate enough for going up hill, which results with the vehicle slowing down.

**Perfect Cruise:** The perfect cruise would not satisfy the noninterference property because it would maintain speed. Since it does not allow the vehicle to changes speed, it is obvious it is not in control.

**Random Cruise:** Adding a random cruise into the system also maintains the noninterference property when the random cruise allows the vehicle to reduce speed. For any reason the random cruise allows the vehicle to speed up or maintains speed, it would violate the noninterference property.

**Theorem 7.10** *Vehicle operation is noninterference secure for a standard cruise and a driver when traveling down hill and increases speed.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and third event are low level events and the second event is a high level input into the system. The low level trace of the vehicle speeding up is $\{l_1 l_3\}$. Because the observer sees the vehicle speeding up while going down hill, there is no indicators that there

www.manaraa.com

is a controller of the vehicle, which satisfies the property of noninterference. Gravity causes the vehicle to speed up regardless to there being a controller of the vehicle.

**Driver:** The driver can sometimes not compensate enough to maintain speed going down a hill and can end up speeding up.

**Standard Cruise:** The standard cruise does not always compensate enough for going down hill, which results with the vehicle speeding up.

**Perfect Cruise:** The perfect cruise would not satisfy the noninterference property because it would maintain speed. Since it does not allow the vehicle to changes speed, it is obvious it is not in control.

**Random Cruise:** Adding a random cruise into the system also maintains the noninterference property when the random cruise allows the vehicle to increase speed. For any reason the random cruise allows the vehicle to slow down or maintains speed, it would violate the noninterference property.

**Discussion** Similar to the noninference property, when the vehicle's low level events correlate with the events expected in the physical design of the system. Because the vehicle naturally speeds up going down hill and slows down when going up hill or traveling on a flat surface, having a controller that allows exhibits these events in the system make it noninterference secure. Applying a perfect cruise or a random cruise into the system does not aide in the obfuscation of the controller.

Applying a random cruise in the system does not result with the same level of obfuscation that it did is the noninference or nondeducibility properties because it can also maintain or an event that violates the noninterference property. Since the

www.manaraa.com

perfect cruise only maintains speed, it is obvious at all times that there is a controller of the system. As a result the noninterference property is stronger when there is an imperfection in the controller that allows for the high level events to hide in the physical design of the scenario.

## 7.2. OBSTRUCTION EXAMPLE

In the previous section, the results showed that depending on the topography of the terrain the control of the system can be obfuscated by the the context of the outside environment. There is also the case that events that also happen externally to the system and it is clear that there is some sort of automatic control within the vehicle.

There are some subtle differences in control by the driver versus control by the car EMS. Obstructions offer one such example. By allowing some entity to intentionally inject low level inputs into the system and monitor the results of the low level inputs, it reveals different characteristics of the system to the observer. Since the previous scenarios only involve an observer watching the vehicle, a new scenario was built to analyze how an interactive observer effects the system. In this scenario, instead of the observer being passive and only observing the vehicle, the observer interacts with the driver by using an object to obstruct the path of the car. This obstruction is severe enough that if it is not avoided, it can cause great harm to the driver. Defining an obstruction in this way allows it to directly interact with the CPS in order to show whether or not information about the system can be exposed. A major determining factor of the satisfaction of this policy is the response time of

www.manaraa.com

the controller; by planting an obstruction on the road that intentionally causes the driver or automatic control to react in a predictable way, it can reveal the controller of the vehicle. The list of events for the traces are listed in Table 7.5. The following scenarios' traces are depicted in Table 7.6. The scenarios in the obstruction example cannot be compared directly to the previous terrain examples because of the events introduced by the obstruction.

Table 7.5. Obstruction Example Event List

| Notation | Input or Output | Description |
|---|---|---|
| $l_{o0}$ | output | Fast Initial speed |
| $l_{o1}$ | output | Slow Initial speed |
| $l_{if}$ | input | Obstruction raised a long distance away. |
| $l_{im}$ | input | Obstruction raised a few car lengths away. |
| $l_{ic}$ | input | Obstruction raised meters away. |
| $h_1$ | | Driver tries to stop car. |
| $h_2$ | | Automatic Control tries to stop car. |
| $l_{o2}$ | output | Obstruction Avoided |
| $l_{o3}$ | output | Obstruction Hit |

**Theorem 7.11** *Vehicle operation is nondeducible secure when an obstruction is raised from a very far distance at high speeds and low speeds.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and fourth events are low level output events and the third event is a high level action performed by a controller of the vehicle. The second event is an low level input into the system. In this case, the results of the obstruction being raised at a great distant results with the driver and automatic control being able to avoid a collision with the object. In regards to the controller, the raised obstruction occurs at a distance that

www.manaraa.com

Table 7.6. Trace List for the Controllers During the Hazard Scenario

| **Scenario** | **Trace** |
|---|---|
| Obstruction Raised at High Speed | |
| Obstruction raised in distance | $\{l_{o0}l_1h_1l_{o2}\}$ |
| Obstruction raised in distance | $\{l_{o0}l_{if}h_2l_{o2}\}$ |
| Obstruction raised a few car lengths away | $\{l_{o0}l_2h_1l_{o3}\}$ |
| Obstruction raised a few car lengths away | $\{l_{o0}l_{im}h_2l_{o2}\}$ |
| Obstruction raised meters away | $\{l_{o0}l_3h_1l_{o3}\}$ |
| Obstruction raised meters away | $\{l_{o0}l_{ic}h_2l_{o3}\}$ |
| Obstruction Raised at Slow Speed | |
| Obstruction raised in distance | $\{l_{o1}l_{if}h_1l_{o2}\}$ |
| Obstruction raised in distance | $\{l_{o1}l_{if}h_2l_{o2}\}$ |
| Obstruction raised a few car lengths away | $\{l_{o1}l_{im}h_1l_{o2}\}$ |
| Obstruction raised a few car lengths away | $\{l_{o1}l_{im}h_2l_{o2}\}$ |
| Obstruction raised meters away | $\{l_{o1}l_{ic}h_1l_{o3}\}$ |
| Obstruction raised meters away | $\{l_{o1}l_{ic}h_2l_{o2}\}$ |

is out of the range of the controller to detect. By the time the automatic control can detect the object, there is still enough time for the car to slow down due to the speed that it can react to the input. Since the driver has a better view distance, the driver can make better decisions for obstructions further than the detection range of the automatic controller. The traces for this scenario are $\{l_{o0}l_{if}h_1l_{o2}\}$ and $\{l_{o0}l_{if}h_2l_{o2}\}$. Looking at only the low level events both controllers, they both have the events $\{l_{o0}l_{if}l_{o2}\}$. Since the obstruction is avoided by both controllers, there is no way to distinguish the controller of the vehicle.

**Driver:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events of the driver, the trace results with the the trace list $\{l_{o0}l_{if}l_{o2}\}$ for the high speed trace and $\{l_{o1}l_{if}l_{o2}\}$ for the low speed trace. In this scenario, it makes sense that the driver has the capability to stop the car in order to avoid the collision with the object. Ultimately having the object at a long distance there is adequate time for a driver to react to the obstruction.

www.manaraa.com

**Automatic Control:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events of the automatic control, the trace results with the the trace list $\{l_{o0}l_{if}l_{o2}\}$ for the high speed trace and $\{l_{o1}l_{if}l_{o2}\}$ for the low speed trace. In this scenario, the vehicle also has the capability to stop by the time it detects the obstruction.

**Theorem 7.12** *Vehicle operation is not nondeducible secure when an obstruction is raised from a few car lengths away and the car is traveling at a high speed.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and fourth events are low level output events and the third event is a high level action performed by a controller of the vehicle. The second event is an low level input into the system. The observer plants a more intelligent event into the system by knowing that driver is probably slower to react than the automatic control. By raising an obstruction only a few car lengths in front of the car, the observer is limiting the amount of reaction time that can be taken to avoid it. The resulting traces for this scenario are $\{l_{o0}l_{im}h_1l_{o3}\}$ and $\{l_{o0}l_{im}h_2l_{o2}\}$. Since the low level events do not match between the traces, this scenario is not nondeducible secure as the controller is uniquely identified.

The obstruction being raised in this scenario gives a much smaller tolerance of time between when the controller sees the obstruction and when the controller must react in order to avoid hitting the object. Because a driver needs some additional time to react (seeing the obstruction, realizing the obstruction needs to be avoided, and moving the body to make the vehicle react), the time it takes to stop the vehicle

www.manaraa.com

is longer for the driver than the automatic controller. The automatic controller has an immediate response to sensing the obstruction and still has time to react to avoid the obstruction. Based on the response of the vehicle, the observer knows who the controller is.

**Driver:** Refer to Table 7.6 for the trace list of this scenario. Purging the high level input from the trace results with the trace list $\{l_{o0}l_{im}l_{o3}\}$. In this scenario, the time it takes the driver to react to the obstruction takes longer than what needs to be done in order to avoid the object. **Automatic Control:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events, the trace results with the the trace list $\{l_{o0}l_{im}l_{o2}\}$. In this scenario, the vehicle has capability to avoid the obstruction.

**Theorem 7.13** *Vehicle operation is nondeducible secure when an obstruction is raised from a few car lengths away and the vehicle is traveling slow.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and fourth events are low level output events and the third event is a high level action performed by a controller of the vehicle. The second event is an low level input into the system. In regards to the controller, the raised obstruction causes the vehicle to stop for both the driver and the automatic control. To the observer, the controller is indistinguishable because the high level event has no impact on the car's low level events. In this case, the results of the obstruction being raised at a moderate distant while the vehicle is moving slow are equivalent for both the high speed and slow speed scenarios with the low level events being $\{l_{o1}l_{im}l_{o2}\}$ for both controllers. The traces

www.manaraa.com

for this scenario are $\{l_{o1}l_{im}h_1l_{o2}\}$ and $\{l_{o1}l_{im}h_2l_{o2}\}$. Because there is no unique set of low level events in this scenario, it is nondeducible secure.

**Driver:** Refer to Table 7.6 for the trace list of this scenario. Purging the high level input from the trace results with the trace list $\{l_{o1}l_{im}l_{o2}\}$. In this scenario, it makes sense that the driver has the capability to stop the car in order to avoid the collision with the object. Ultimately having the object at a moderate distance in a slow moving vehicle there is adequate time for a driver to react to the obstruction.

**Automatic Control:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events, the trace results with the the trace list $\{l_{o1}l_{im}l_{o2}\}$. In this scenario, the vehicle has the same capability to avoid the obstruction as the driver does.

**Theorem 7.14** *Vehicle operation is noninterference secure when an obstruction is raised meters away and the vehicle is traveling at a high speed.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and fourth events are low level output events and the third event is a high level action performed by a controller of the vehicle. The second event is an low level input into the system. In this case, the results of the obstruction being raised at a close distance are equivalent. Because the obstruction is raised at such a close proximity to the vehicle, neither the driver or the automatic control can react in time to give the car enough space to stop.

**Driver:** Refer to Table 7.6 for the trace list of this scenario. Purging the high level input from the trace results with the trace list $\{l_{o0}l_{ic}l_{o3}\}$. In this scenario, the

www.manaraa.com

vehicle is moving too fast to stop, even if the reaction was instantaneous.

**Automatic Control:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events, the trace results with the the trace list $\{l_{o0}l_{ic}l_{o3}\}$. In this scenario, the vehicle is moving too fast to stop, even if the reaction is instantaneous.

**Theorem 7.15** *Vehicle operation is nondeducible secure when an obstruction is raised meters in front of the vehicle and the vehicle is traveling slow.*

**Proof** : The EMS is a nondeterministic system where for each trace the first and fourth events are low level output events and the third event is a high level action performed by a controller of the vehicle. The second event is an low level input into the system. In this case, the results of the obstruction being raised at a close distance while the vehicle is moving slow are equivalent for both the high speed and slow speed scenarios. In regards to the controller, the raised obstruction causes the vehicle to stop for both the driver and the automatic control. The traces for this scenario are $\{l_{o1}l_{ic}h_1l_{o2}\}$ and $\{l_{o1}l_{ic}h_2l_{o2}\}$. The low level events are $\{l_{o1}l_{ic}l_{o2}\}$ for all the traces. To the observer, the controller is indistinguishable because the high level events of both the driver or automatic control causes no unique low level events trace, making the scenario nondeducible secure.

**Driver:** Refer to Table 7.6 for the trace list of this scenario. Purging the high level input from the trace results with the trace list $\{l_{o1}l_{ic}l_{o2}\}$. In this scenario, it makes sense that the driver has the capability to stop the car in order to avoid the

www.manaraa.com

collision with the object. Ultimately having the object at a close distance in a slow moving vehicle there is adequate time for a driver to react to the obstruction.

**Automatic Control:** Refer to Table 7.6 for the trace list of this scenario. Looking at only the low level events, the trace results with the the trace list $\{l_{o1}l_{ic}l_{o2}\}$. In this scenario, the vehicle has the same capability to avoid the obstruction as the driver does.

**Discussion** The results for the obstruction example is quite different from the terrain example. The major difference with this scenario compared to the scenarios of the previous policies is instead of the observer being a passive entity in the system, it now becomes interactive with it. In this scenario, the driver is using what it knows about how a vehicle will react with an obstruction to learn information pertaining to the controller of the vehicle. The controller in the various scenarios is mostly secure. The results expose two major pieces of knowledge in certain conditions. The other information exposed by this scenario is that a obstruction can be raised that only the automatic controller has time to react to, revealing that there is a difference in reaction time between the driver and the automatic control. This is a perfect example to illustrate how using a set of low level events, a low level user (violating no security rules) can learn about the high level events in the system.

www.manaraa.com

# 8. CONCLUSIONS

Security comes from a need to protect information that has to be revealed to provide a service and a role. The leakage of data is a risk of any system. The amount of information that is released effects how much can be inferred about the system.

Applying security concepts to a car reveals many characteristics that are both positive and negative with respect security in both the cyber-physical domains and the information security domains.

There are several properties that make up the example vehicle system. A vehicle requires that it provides reasonable predictability, otherwise it is unreasonable for a person to operate a car. Core functions provided by the car can be considered: speeding up, slowing down, protection, and visibility. In order for a car to be predictable, its core functions are not noninference secure, noninterference secure or nondeducible secure as shown by the proofs in this thesis. The inner components of the car can be known or unknown for the proper operation of the vehicle.

The most revealing aspects of the car, was in the scenario that an observer is interacting with the system, like in the obstruction example. During this scenario, since the predictability of the vehicle is satisfied, the observer can easily determine information about the controller of the vehicle. This is due to the vehicle being a well-known system and the problems with there being any variation to the reaction due to certain stimuli on the controller.

www.manaraa.com

Because of those two facts, it is easy to generate low level inputs to influence the high level events in the system, resulting with the expected low level output to the observer.

The functions that the critical components' composition make reveals information to the driver and reveals information to the people external to the car, but more information about these components are required to understand how they work. The information that is being revealed to the outside, though, still takes some previous knowledge of the system. The car is a well-known system so the information required to understand how a car operates is often taken for granted. A system that is not well known can appear much more secure.

The design of any system must always be considered carefully. If a car should have been designed to not reveal information about what the car was doing, it would fail miserably. The question to ask is: "If it is required to reveal this information, how much will it take to fit the pieces together to understand the information that is being revealed? How bad is it to reveal it?"

www.manaraa.com

# 9. FUTURE WORK

The process of securing the cyber physical systems is still in its infancy. Referring back to Sections 1.2.1, 1.2.2, 1.2.3, 1.2.4, and 1.2.5, each of these sections have not completely had their problems solved.

While this thesis touched on most of these concepts, it focused on privacy and modeling. Currently models that are being utilized have the capacity for modeling specific aspects of the CPS [7], but modeling it as a whole does not exist. A model that can encapsulate the different levels of computation, real time constraints and define the overall rules of each of the subparts must satisfy on a single view.

Another important piece missing from CPS analysis is the means to measure aspects of security [8]. For instance, if an outside observer can accurately read data outside of the system, what is the probability that it can determine how the high level system is using that value. Another good measuring system would be a component fail rating. This rating would represent the impact to the rest of the system, in the case that it fails. Ideally all components would have no impact on failure, but this is really difficult to satisfy, but strategies to meet this criteria can be incorporated into the designs to prevent future problems.

Another future research topic can focus on the generalization of the CPS. There are specific requirements for each CPS, but all CPS have the same composition. Refer to Section1.1.

www.manaraa.com

The difference between each CPS is the what part of the physical world that the CPS interacts with, so a strategy to generalize the physical world goal into a generic CPS semantic needs to be created.

This thesis only is the start to the real work that has to be completed in the CPS domain. This simple example clearly shows the complexity and difficulties of looking at all the facets of a system and illustrates the need to push further into researching this domain.

www.manaraa.com

# BIBLIOGRAPHY

[1] E. A. Lee, "Computing foundations and practice for cyber-physical systems: A preliminary report," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2007-72, May 2007. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-72.html

[2] K. Wan, D. Hughes, K. L. Man, and T. Krilavicius, "Composition challenges and approaches for cyber physical systems," in *Proceedings of the 1st IEEE International Conference on Networked Embedded Systems for Enterprise Applications, NESEA 2010, November 25-26, 2010, Suzhou, China*. IEEE, 2010, pp. 1–7.

[3] R. Akella, H. Tang, and B. M. McMillin, "Analysis of information flow security in cyberphysical systems," *International Journal of Critical Infrastructure Protection*, vol. 3, no. 34, pp. 157 – 173, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874548210000405

[4] M. Bishop, *Computer Security: Art and Science*. Addison-Wesley, Dec. 2002. [Online]. Available: http://nob.cs.ucdavis.edu/book/book-aands/

[5] M. Surridge and C. Upstill, "Grid security: lessons for peer-to-peer systems," in *Peer-to-Peer Computing, 2003. (P2P 2003). Proceedings. Third International Conference on*, sept. 2003, pp. 2 – 6.

[6] D. Work, A. Bayen, and Q. Jacobson, "Automotive cyber physical systems in the context of human mobility," in *National Workshop on High-Confidence Automotive Cyber-Physical Systems, Troy, MI, April 3-4, 2008*. IEEE, 2008, pp. 1–3.

[7] E. A. Lee, "Cyber-physical systems - are computing foundations adequate?" in *Position Paper for NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*, October 16 - 17, 2006.

[8] R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proceedings of the 47th Design Automation Conference*, ser. DAC '10. New York, NY, USA: ACM, 2010, pp. 731–736. [Online]. Available: http://doi.acm.org/10.1145/1837274.1837461

www.manaraa.com

[9] K. Shin and P. Ramanathan, "Real-time computing: a new discipline of computer science and engineering," *Proceedings of the IEEE*, vol. 82, no. 1, pp. 6 –24, jan 1994.

[10] P. Pal, R. Schantz, K. Rohloff, and J. Loyall, "Cyber-physical systems security-challenges and research ideas," pp. 157 – 173, 2009.

[11] D. K. Nilsson, U. E. Larson, and E. Jonsson, "Creating a secure infrastructure for wireless diagnostics and software updates in vehicles," in *SAFECOMP '08: Proceedings of the 27th international conference on Computer Safety, Reliability, and Security*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 207–220.

[12] D. McCullough, "A hookup theorem for multilevel security," *IEEE Trans. on Software Engineering*, pp. 1–3, 1996.

[13] J. McLean, "Security models and information flow," in *Proceedings of the 1990 IEEE Computer Society Press*, 1990.

[14] ——, "Security models," in *Encyclopedia of Software Engineering*, 1994.

[15] ——, "A general theory of composition for a class of 'possibilistic' security properties," in *IEEE Trans. on Software Engineering*, 1996, pp. 53–67.

[16] A. Zakinthinos and E. Lee, "A general theory of security properties," in *Procs. of the 18th IEEE Computer Society Symposium on Research in Security and Privacy*, 1997, pp. 94–102.

[17] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive can networks - practical examples and selected short-term countermeasures," in *SAFECOMP*, 2008, pp. 235–248.

[18] K. J. Biba, "Integrity considerations for secure computer systems," p. 66, 1977.

[19] D. E. Bell and L. J. Lapadula, "Secure computer system: Unified exposition and MULTICS interpretation," The MITRE Corporation, Tech. Rep. ESD-TR-75-306, 1976. [Online]. Available: http://csrc.nist.gov/publications/history/bell76.pdf

[20] M. A. Harrison, W. L. Ruzzo, and J. D. Ullman, "Protection in operating systems," *Commun. ACM*, vol. 19, pp. 461–471, August 1976. [Online]. Available: http://doi.acm.org/10.1145/360303.360333

www.manaraa.com

[21] ——, "Protection in operating systems," *Commun. ACM*, vol. 19, pp. 461–471, August 1976. [Online]. Available: http://doi.acm.org/10.1145/360303.360333

[22] A. Myers and B. Liskov, "Complete, safe information flow with decentralized labels," in *Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on*, may 1998, pp. 186 –197.

[23] J. Madden, B. McMillin, and A. Sinha, "Environmental obfuscation of a cyber physical system - vehicle example," in *Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual*, july 2010, pp. 176 –181.

www.manaraa.com

# VITA

Jason Lewie Madden's received his high school diploma from Hannibal Senior High School. He was also involved in the Boy Scouts of America where he achieved Eagle Scout and a Brotherhood member in the Order of the Arrow. In 2004, Jason completed his Associate of Science with an emphasis in Computer Science at Moberly Area Community College located in Moberly, MO. Because of his merits and reputation among his professors, he was awarded a scholarship to continue at a four year school in the science and engineering fields.

In the fall of 2004, Jason enrolled into Missouri University of Science and Technology to continue his studies that combined a mixture of artificial intelligence (A.I.) and operating system theory. Later, Jason began working on various software projects as a Undergraduate Assistant, under Dr. Bruce McMillin, while he completed his B.S. in Computer Science. He continued his graduate studies under Dr. McMillin as a Graduate Research Assistant. His graduate studies focused on advanced Operating System Theory and Security.

Before completing his Masters, Jason went to work in the software industry as a Software Developer while he continued to complete his graduate studies. During this time, Jason was an co-author to a published conference paper which is included as a reference in some of the research. In May 2013, he completed his Master's Degree in Computer Science from Missouri University of Science and Technology.

www.manaraa.com

المنارة للاستشارات
www.manaraa.com