2013 IEEE 37th Annual Computer Software and Applications Conference

# A Multiple Security Domain Model of a Drive-by-Wire System

Gerry Howser and Bruce McMillin
*Department of Computer Science*
*Missouri University of Science & Technology*
*500 West 15th Street, Rolla, Missouri - 65409*
{*gwhrkb, ff*}*@mst.edu*

***Abstract*—Traditional security models partition the security universe into two distinct and completely separate worlds: us and them. This partition is absolute and complete. More complex situations are most commonly treated as sets of increasingly more secure domains.**

**This view is too simplistic for cyber-physical systems. Absolute divisions are conceptually clean, but they do not reflect the real world. Security partitions often overlap, frequently provide for the *high* level to have complete access to the *low* level, and are more complex than an impervious wall. We present a model that handles situations where the security domains are complex or the threat space is ill defined.**

**To demonstrate our method, we examine a "drive by wire" system from both the traditional view and in light of the modern reality. This paper examines the system from the viewpoint of the driver with special emphasis on the driver's inability to determine who, or what, is actually in control of the automobile during critical situations.**

***Keywords*-information flow security, cyber-physical systems, drive-by-wire systems,security models, nondeducibility, modal logic**

## I. Introduction

Compared with information and computer systems, cyber-physical systems (CPS) present an interesting perspective of security from three standpoints. First, cyber-physical systems have at their heart a control mechanism that is either a computer system or something similar. This requires us to examine the challenges of securing a computer system. Secondly, the physical side of the system makes it easy to understand how observing a system can leak information thus facilitating adversarial attacks on the system. Third, cyber-physical systems are prone to collateral damage when compromised. For the same reasons, it is difficult to secure CPS because you must secure both data and information flow in a system that can be watched.

It is natural to reduce the concept of security to "walling the bad guys out." From primitive forts to sophisticated medieval castles to modern computer security systems, this model has held up well. Unfortunately, as situations become more complex, and the "bad guys" more astute, these models became less effective.

Information flow security in cyber-physical systems (CPS) leads to particularly challenging and complex security domains. Most security models are composed of "secure" and "not secure". Unfortunately, this focus leaves these models open to attacks that do not steal information but simply disrupt critical information flow.

Nondeducibility (ND) was introduced by Sutherland [1] as an attempt to use modal techniques to model data in a partitioned security system. The possible worlds (e.g.,state collections) of this model are partitioned into disjoint sets and information is restricted to *one side of the partition or the other* [2]. Information that could not be inferred from the other side of the partition was determined to be Nondeducibility secure. Overlapping security domains break Sutherland's Nondeducibility as do information flows we simply cannot evaluate.

We introduce a new modal technique to model complex security domains, Multiple Security Domain Model Nondeducibility (MSDND). We show that MSDND can model any system where Sutherland Nondeducibility holds and complex systems where Nondeducibility cannot be determined. MSDND models CPS well, even when the security domains overlap or the boundaries are not ideal and leak.

In Section V we use both traditional Nondeducibility and Multiple Security Domain Nondeducibility to model a "drive by wire" automobile connected to a roadside assistance network such as General Motors OnStar or ToyotaConnect.

> **Of prime concern is the simple question: can the driver determine when the car is under his/her control, the control of the on-board computer, or under the control of something outside the car? [3]**

Section IV outlines the modal techniques and theory behind both security models. In Section V we model in detail: normal operations, hazardous road conditions (a.k.a traction control), and corporate remote control of the car.

## II. Problem Statement

Computer security tools work well for computers, but cyber physical systems leak information because the physical part of the system can be watched for changes. By their very nature, CPS are messy from a security domain view point. Domains overlap, the boundaries are not clean (ideal boundaries cannot leak information), and outside threats can leak into domains thought to be secure.

Computer security tools work best when secure domains are cleanly nested inside less secure domains like a Medieval

0730-3157/13 $26.00 © 2013 IEEE
DOI 10.1109/COMPSAC.2013.62

castle with its outer walls and interior keep, see Fig. 1. This model serves us well for most uses but breaks down when applied to CPS. Because CPS typically need to secure both data and information flows, the security domain picture gets complicated, see Fig. 2. We need tools that can model the cyber and physical components of CPS.

## III. Manufacturer/Automotive System Functional Model

An example of complex security domains is a drive-by-wire car equipped with remote assistance such as OnStar or Toyota Connect. The model consists of a corporation($corp$) that provides service to its drivers($driver$) in the form of remote assistance (navigation, remote unlock, remote shutdown, etc.) and an automobile($car$) with on-board drive-by-wire functionality. We will examine three modes of operation.

***Specific modes of operations**:*

*Normal Operations:* The driver can operate the vehicle as one would normally expect. From this, the driver knows he/she controls the car.

*Hazardous Road Conditions:* Most current automobiles are equipped with varying degrees of traction control systems to automatically correct for a loss of traction which can be thought of as a super set of the Automatic Braking System (ABS) and are very effective. While a traction control is operating, the car will attempt to counter a skid and counter anything the driver does that would make the skid worse.

*Corporate Remote Operations:* If the car is equipped with a service similar to OnStar, the corporation can issue commands to the car. The driver must trust the corporation to act in his best interests [3] [4].

Depending on which mode the car is in, the driver may not be able to distinguish who or what is actually in control. Of particular interest is remote operation by *corp* which exists in one security domain v.s. operation by *driver* in another security domain. What the driver can and cannot ascertain is governed by the information flow that exists among domains, both in the cyber, and in the physical.

The ensuing discussion shows how classic models of information flow and deducibility break down in the cyber-physical environment. The paper develops a multiple security domain model and applies it to the car model.

## IV. Two Modal Logic Based Models

We will examine the classical Sutherland Nondeducibility Model and the Multiple Security Domain Nondeducibility Model as modal models. Informally, each possible combination of binary state variables defines a world, $w$. Changes in any state variable cause a transition to a different world, much like a labeled transition state machine. We can build a model of how the system operates upon a framework of the possible combinations of states and the transitions between those combinations. This model should behave like our physical car. We will look at a series of events, an Event System(ES), to understand what the car reveals to the driver and what the car hides from the driver.

***Modal Logic Model**:* More formally, we define a set of worlds, $\{W\}$ consisting of distinct worlds, $w_0, w_1, \ldots, w_n$ where, if we have $m$ state variables, $s_1, s_2, \ldots, s_m$ then we can have $2^m$ distinct worlds.

The worlds are connected by a set of transitions, $\{wRw^{'}\}$. Changing any state variable causes a transition from the current world, $w$, to another world, $w^{'}$ where all other state variables retain their values. Together, the set of worlds and transitions define a frame, $\mathfrak{F} = \{W, R\}$.

We define a set of valuation functions, $\{V\}$, such that $V^i_{s_x}(w)$ returns the value of state variable $s_x$ as seen by an entity $i$ in world $w$. **NOTE:** If no valuation function exists to return the value of a state variable, say $s_i$, then our model can never determine the value of that state variable nor the value of any logical expression dependent upon that state variable. We can now define a model $M = \{\mathfrak{F}, V\}$ or $M = \{W, R, V\}$.

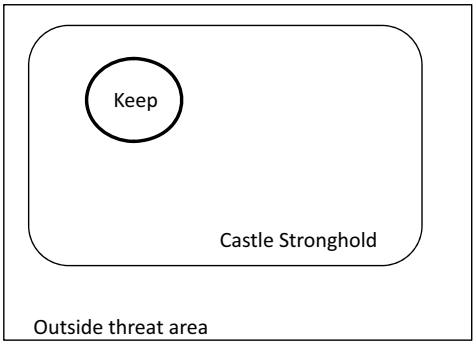


Figure 1. A Medieval Castle Model of Security

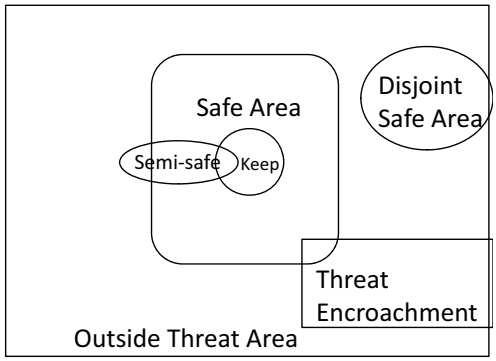


Figure 2. Problematic Overlapping Security Domains

We can informally define some modal logic symbols (see Table I):

| | |
|---|---|
| $\Box\varphi$ | $\varphi$ is always true here |
| $\Diamond\varphi$ | $\varphi$ **might** be true here |
| $w \vdash \varphi$ | in $w, \varphi$ is the conclusion of a valid proof |
| $w \vDash \varphi$ | in $w$, states are such that $\varphi$ is true |

Suppose we have a set $\varphi, \psi \in \phi_0$ of atomic propositions. Questions such as "is the brake pedal pushed" can be asked by evaluating well-formed formulas built from these atomic propositions. The set of such formulas is closed under the following rules:

- if $\varphi$ is a wff, so are $\neg\varphi, \Box\varphi$, and $\Diamond\varphi$
- if $\varphi$ and $\psi$ are wff, then so is $\varphi \vee \psi$
- if $\varphi$ and $\psi$ are wff, then so is $\varphi \wedge \psi$

We define the modal operator, $\Box\varphi$, as an abbreviation for $\neg \Diamond \neg\varphi$. The axiomatic system is given in Table I.

*Sutherland Nondeducibility Model:* In the Sutherland model, the valuation functions are the same for all entities in the same security domain. Typically, some evaluations are restricted to one domain. We will look at a generic case with two valuations, $V_1(w)$ and $V_2(w)$. We can define the Sutherland Nondeducibility [5] with respect to the valuation, $V_2$ for our model as:

$$\begin{aligned} ND(ES) = \quad & (\forall w \in W : V_2(w), V_1(w) \neq \varnothing) \\ & \exists w^{'} : [V_1(w) = V_1(w^{'})] \wedge [V_2(w) = V_2(w^{'})] \end{aligned}$$

*Multiple Security Domains Nondeducibility Model:* Extending existing models to multiple security domains is problematic. We offer a different approach, the Multiple Security Domains Nondeducibility Model. We define an entity $i$ as any part of the system capable of independent observation or action. We can divide our Event System, $(ES)$, into multiple security domains, $SD^i$, as viewed by each entity $i$ in the model. These domains may, or may not, overlap. These multiple security domains conform to the following rules:

$$\bigcup_{i \in I} SD^i = (ES).$$

**Definition.** *Multiple Security Domains Nondeducibility*
*There exists some world with a pair of states where one must be true and the other false (exclusive OR), but an entity $i$ has no valuation function for those states. In security domain $SD^i$, we simply cannot know which state true and which is false.*

$$\begin{aligned} MSDND(ES) = \quad & \exists w \in W : w \vdash \Box\big[(s_x \vee s_y) \wedge \neg(s_x \wedge s_y)\big] \\ & \wedge \big[w \vDash (\nexists\, V_x^i(w) \wedge \nexists\, V_y^i(w))\big] \end{aligned}$$

*An equivalent formulation would be:*

$$\begin{aligned} MSDND(ES) = \quad & \exists w \in W : w \vdash \Box\big[s_x \oplus s_y\big] \\ & \wedge \big[w \vDash (\nexists\, V_x^i(w) \wedge \nexists\, V_y^i(w))\big] \end{aligned}$$

*Reduction of the Sutherland Model to MSDND:*

**Theorem IV.1.** *Any arbitrary case where $ND(ES)$ holds can be shown to be a special case of $MSDND(ES)$*

*Proof:* Given: A system with two security domains, *left* and *right*, and two distinct worlds $w^{'}, w^{''} \in W$ where $ND(ES)$ holds. NOTE: The use of *left* and *right* as designations is to emphasize that MSDND is not a high/low hierarchy model, but is instead a partitioning model.
With no loss of generality, we can easily frame this valuation as a binary decision value because in the current world, $w$, either the *right* event has occurred $(w^{'})$ or it has not $(w^{''})$. We will assign two state variables such that $st \Rightarrow (w = w')$ and $sf \Rightarrow (w = w'')$. Because this case is $ND(ES)$, it follows that the *left* domain cannot evaluate either $st$ or $sf$ because to do so would break $ND(ES)$. It is now easy to construct the conditions for $MSDND(ES)$.

| | |
|---|---|
| $w^{'}, w^{''} \in W \Rightarrow w \in W$ | By construction |
| $w \vdash \Box(st \oplus sf)$ | By construction |
| $\nexists\, V_{st}^{left}(w)$ | ND(ES) |
| $\nexists\, V_{sf}^{left}(w)$ | ND(ES) |
| $w \vDash \big[\nexists\, V_{st}^{left}(w) \wedge \nexists\, V_{sf}^{left}(w)\big]$ | ND(ES) |

Since we constructed the first clause as a tautology, by *conjunction* we can construct the conditions for $MSDND(ES)$

$$\begin{aligned} MSDND(ES) = \quad & \exists w \in W : w^{''} \vdash \Box(st \oplus sf) \\ & w \vDash (\nexists\, V_{st}^{left}(w) \wedge \nexists\, V_{sf}^{left}(w)) \end{aligned}$$

■

*Remarks about the reduction from Sutherland Model to Multiple Security Domains Model:* It is possible to reduce any system that meets ND(ES) to one that meets MSDND(ES) by defining decision variables for each variable that is ND(ES). However, a reduction in the other direction, MSDND(ES) to ND(ES) is not always possible. We show MSDND(ES) works even in the case where the model under examination does not contain a valuation function capable of returning the value of $\varphi$ in all worlds. Sutherland's ND(ES) does not address this situation.

## V. Specific Example of the Drive-by-Wire Prius

*Structure of the model:* We will limit our discussion to the state variables given in Table II and III.

We will now define a set of logical conditions, $\varphi_i, d, t, c, f$, that we can evaluate to determine how the car is responding to commands, see Table III.

Similarly, we can define valuation functions for some of the state variables in the frame as given in Table IV. On any given world, these valuation functions will return the value of the corresponding state variable as seen by the entity in control $i \in \{d, t, c, f\}$. Either the driver $d$, traction control $t$, or corporation $c$ is in control or the car is faulty $f$ and nothing is in control.

Table I
THE AXIOMATIC SYSTEM

1. Definition of logical and modal operators (abbreviations)
   D1: $\varphi \oplus \psi \equiv (\varphi \vee \psi) \wedge \neg(\varphi \wedge \psi)$ (Exclusive OR)
   D2: $\Diamond\varphi \equiv \exists w \in W : w \vdash \varphi$ ($\varphi$ might be true here)
   D3: $\Box\varphi \equiv \neg \Diamond \neg\varphi$ ($\varphi$ true in all worlds)

2. Axioms
   P: all the tautologies from the propositional calculus
   K: $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$

3. Rules of Inference
   R1: from $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$ infer $\psi$ (Modus Ponens)
   R2: $\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$ (DeMorgan's)
   R3: from $\vdash \varphi$ infer $\vdash \Box\varphi$ (Generalization)

Table II
DEFINITION OF STATE VARIABLES

| Variable | |
|---|---|
| $s_0$ | Car is behaving normally($\top$) |
| $s_1$ | $driver$ is aware of car's behavior |
| $s_2$ | $car$ is accepting commands from $driver$ |
| $s_3$ | $car$ is accepting commands from $tc$ |
| $s_4$ | $car$ is accepting commands from $corp$ |
| $s_5$ | $car$ is faulty and not accepting commands |

From observing the actual operation of the car, there is an obvious constraint.

**Constraint** (The $car$ can allow only one source of commands, $control_i$ at a time)**.** *For some arbitrary world,* $w \in W$*, this can be expressed by the following set of conditions:*

$$w \vDash d \quad \Leftrightarrow \quad w \vdash \Box\neg(t \vee c \vee f)$$
$$w \vDash t \quad \Leftrightarrow \quad w \vdash \Box\neg(c \vee f \vee d)$$
$$w \vDash c \quad \Leftrightarrow \quad w \vdash \Box\neg(f \vee d \vee t)$$
$$w \vDash f \quad \Leftrightarrow \quad w \vdash \Box\neg(d \vee t \vee c).$$

*This constraint can be expressed as the predicate which evaluate to* 1 *if that entity is in control and* 0 *otherwise:*

$$w \vDash d \quad \Leftrightarrow \quad control_d = control_1 = 1$$
$$w \vDash t \quad \Leftrightarrow \quad control_t = control_2 = 1$$
$$w \vDash c \quad \Leftrightarrow \quad control_c = control_3 = 1$$
$$w \vDash f \quad \Leftrightarrow \quad control_f = control_4 = 1$$
$$w \vdash \Box\left(\sum_{i=1}^{4} control_i = 1\right).$$

***The Sutherland Nondeducibility Model****:* We will now examine how the car behaves under Sutherland Model with respect to the $driver$. To make the analysis clearer, we can define some common worlds in Table V and common evaluation functions to be used for the entire discussion of the Sutherland Nondeducibility Model. The evaluation functions for *right* domain elements $tc$ and $corp$ are identical but the evaluation function for the *left* element $driver$ must reflect the lack of access to *right* level entities. A valid set of evaluations is given in Figure 3.

Table III
LOGICAL STATEMENTS OF INTEREST

| $\varphi_i$ | state | |
|---|---|---|
| $\varphi_0$ | $s_0$ | The car is behaving normally |
| $\varphi_1$ | $s_1$ | $driver$ is aware of car's behavior |
| $\varphi_2$ | $s_2$ | The $driver$ is in command |
| $\varphi_3$ | $s_3$ | Traction Control is in command |
| $\varphi_4$ | $s_4$ | The $corp$ is in command |
| $\varphi_5$ | $s_5$ | The $car$ is not working correctly |
| $s_d$ | $d = \top$ | $d = \varphi_2 \wedge \neg\varphi_3 \wedge \neg\varphi_4 \wedge \neg\varphi_5$ |
| $s_t$ | $t = \top$ | $t = \neg\varphi_2 \wedge \varphi_3 \wedge \neg\varphi_4 \wedge \neg\varphi_5$ |
| $s_c$ | $c = \top$ | $c = \neg\varphi_2 \wedge \neg\varphi_3 \wedge \varphi_4 \wedge \neg\varphi_5$ |
| $s_f$ | $f = \top$ | $f = \neg\varphi_2 \wedge \neg\varphi_3 \wedge \neg\varphi_4 \wedge \varphi_5$ |

Table IV
VALUATION FUNCTIONS OF OUR MODEL

| Valuation | Result |
|---|---|
| $V_0^i(w) = s_0 \wedge \top$ | "true" $\Leftrightarrow$ $car$ is behaving normally |
| $V_1^i(w) = s_1 \wedge \top$ | "true" $\Leftrightarrow$ $driver$ knows he is in control |
| $V_2^i(w) = s_2 \wedge \top$ | "true" $\Leftrightarrow$ $driver$ is in control of $car$ |
| $V_3^i(w) = s_3 \wedge \top$ | "true" $\Leftrightarrow$ $tc$ is in control of $car$ |
| $V_4^i(w) = s_3 \wedge \top$ | "true" $\Leftrightarrow$ $corp$ is in control of $car$ |
| $V_5^i(w) = s_3 \wedge \top$ | "true" $\Leftrightarrow$ $car$ is in a failure state |

*Normal Operations:*

**Theorem V.1.** *The Sutherland model permits information flow to the* driver *under normal operations.*

Informally, $car$ does what $driver$ asks and this allows $driver$ to know what controls $car$. Under Normal Operations, ND(ES) does not hold as information has leaked from the security domain of the car to the security domain of the driver. This security failure is desirable from the viewpoint of the driver.

*Hazardous Road Conditions:*

**Theorem V.2.** *The Sutherland model prevents information flow to the* driver *under hazardous road conditions.*

*Proof:* When the car senses hazardous road conditions, control is automatically transferred from $driver$ to $tc$. The driver, and passengers, can still sense the actions of the car due to the cyber-physical nature of the entire system but cannot evaluate what is causing the car to do what the driver senses. Using the worlds, states, and evaluation functions we have previously defined (see Tables II and V and Figure 3) we see:

$$V_2^d(w_3) = V_2^d(w_5) = (s_2 = \bot)$$
$$V_3^d(w_3) = V_3^d(w_5) = \top$$
$$V_5^d(w_3) = V_5^d(w_5) = \top$$
$$V_3^t(w_3) \neq V_3^t(w_5)$$
$$V_5^t(w_3) \neq V_5^t(w_5)$$

From the viewpoint of the $tc$(*right*):

$$V_i^t(w) = s_i$$
$$V_i^c(w) = s_i$$
$$V_i^d(w) = \begin{cases} s_i & i < 3 \\ (s_3 \vee s_4 \vee s_5) & \text{otherwise} \end{cases}$$

Figure 3. Evaluation Functions for our Drive-by-Wire Car

Table V
POSSIBLE WORLDS $w_i$ FOR OUR DRIVE-BY-WIRE CAR

| world | in control | $s_2$ | $s_3$ | $s_4$ | $s_5$ |
|---|---|---|---|---|---|
| $w_2$ | $d$ | $\top$ | $\bot$ | $\bot$ | $\bot$ |
| $w_3$ | $t$ | $\bot$ | $\top$ | $\bot$ | $\bot$ |
| $w_4$ | $c$ | $\bot$ | $\bot$ | $\top$ | $\bot$ |
| $w_5$ | $f$ | $\bot$ | $\bot$ | $\bot$ | $\top$ |

$V_2^t(w_3) = V_2^t(w_5) \wedge (V_3^t(w_3) \neq V_3^t(w_5))$
From the viewpoint of the $driver$(*left*):
$V_2^d(w_3) = V_2^d(w_4) = V_2^d(w_5)$
$\wedge(\nexists\, V_3^d(w)) \wedge (\nexists\, V_4^d(w))$
$\wedge(\nexists\, V_5^d(w))$

The fact that $tc$ is in control is ND(ES) secure from the driver because the driver lacks valuations $V_3^d(w), V_4^d(w)$, and $V_5^d(w)$. From the view of the driver, Sutherland ND(ES) fails. The driver is confused. ■

*Corporate Remote Operations:*

**Theorem V.3.** *The Sutherland model prevents information flow to the driver under remote operations.*

*Proof:* When the car receives a command to begin corporate remote operations, control is automatically transferred from $driver$ to $corp$. The driver, and passengers, can still sense the actions of the car due to the cyber-physical nature of the entire system but cannot evaluate what is causing the car to do what the driver senses. Using the worlds, states, and evaluation functions we have previously defined (see Tables II and V and Figure 3) we see:

$V_2^d(w_4) = V_2^d(w_5) = (s_2 = \bot)$
$V_4^d(w_4) = V_4^d(w_5) = \top$
$V_5^d(w_4) = V_5^d(w_5) = \top$
$V_4^c(w_4) \neq V_4^c(w_5)$
$V_5^c(w_4) \neq V_5^c(w_5)$

From the viewpoint of the $corp$(*right*):
$V_2^c(w_4) = V_2^c(w_5) \wedge (V_4^c(w_4) \neq V_4^c(w_5))$
From the viewpoint of the $driver$(*left*):
$V_2^d(w_3) = V_2^d(w_4) = V_2^d(w_5)$
$\wedge(\nexists\, V_3^d(w)) \wedge (\nexists\, V_4^d(w))$
$\wedge(\nexists\, V_5^d(w))$

ND(ES) allows the corporation to know it is in control, and once again ND(ES) cannot be evaluated from the viewpoint of the confused driver. ■

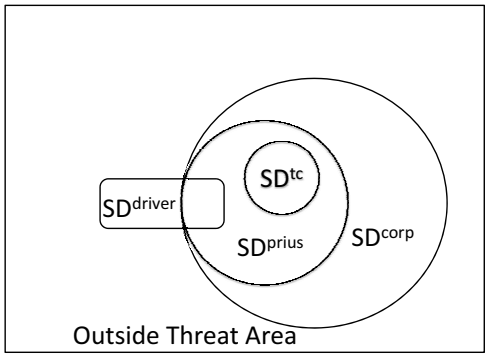


Figure 4. Security Domains in Our Model

*Remarks about applying the Sutherland Nondeducibility Model:* Under Normal Operations, Nondeducibility does not hold and information leaks that the driver is in control. Indeed, the driver needs to know that he or she is in control of the car. The driver lacks the valuation functions to know what is in control in the case of hazardous conditions or remote operations.

***Multiple Security Domains Nondeducibility Model:*** We will now examine how the model behaves when we take overlapping security domains into account, see Figure 4.

*Hazardous Road Conditions:*

**Theorem V.4.** *The MSDND model yields Nondeducibility, thereby stopping critical information flow to the driver under hazardous road conditions.*

Under hazardous conditions, the $car$ acts exactly as in the Sutherland Model theorem V.2. Using the worlds, states, and evaluation functions we have previously defined (see Tables II and V and Figure 3) we see:

*Proof:* Given: The $driver$ knows something else is controlling the car and constraint V still holds.

1. $\exists w \in W : w \vDash \neg d$ — $driver$ is not in control here
2. $w \vdash \Box\left(\sum_{i=1}^{4} control_i = 1\right)$ — something must be in control
3. $w \vdash \Box\left(\sum_{i=2}^{4} control_i = 1\right)$ — $tc$, $corp.$, or broken
4. $V_2^d(w) = (s_2 = \bot)$ — $driver$ sees car's actions
5. $w \vDash \nexists\, V_3^d(w)$ — $driver$ can't tell it's $tc$
6. $w \vDash \nexists\, V_4^d(w)$ — is it $corp.$?
7. $w \vDash \nexists\, V_5^d(w)$ — is it broken?
8. Combining statements 3, 5, and 7 we obtain

$$MSDND(ES) = \exists w \in W : \left[w \vdash \Box\left(\sum_{i=2}^{4} control_i = 1\right)\right] \wedge \left[w \vDash (\nexists\, V_3^d(w) \wedge \nexists\, V_5^d(w))\right]$$

■

The $driver$ has a problem. In the domain $SD^d$ the physical actions of the car can be deduced, but the only deduction $driver$ can make is that he or she is not in control of the car. Strictly speaking, $driver$ does not have all the needed valuation functions and cannot even evaluate Sutherland ND(ES). Using the MSDND(ES) definition, the driver can correctly determine Nondeducibility. The driver can correctly determine he is not in control, but cannot determine exactly what is in control.

*Corporate Remote Operations:*

**Theorem V.5.** *The MSDND model yields Nondeducibility, thereby stopping critical information flow to the driver during remote operations.*

Under corporate remote operations, $car$ behaves as before, see theorem V.3. Using the worlds, states, and evaluation functions we have previously defined (see Tables II and V and Figure 3) we see:

*Proof:* Given: The $driver$ knows something else is controlling the car and constraint V still holds.

1. $\exists w \in W : w \vdash \neg d$ — $driver$ is not in control here
2. $w \vdash \Box\left(\sum_{i=1}^{4} control_i = 1\right)$ — something must be in control
3. $w \vdash \Box\left(\sum_{i=2}^{4} control_i = 1\right)$ — $tc$, $corp.$, or broken
4. $V_2^d(w) = (s_2 = \perp)$ — $driver$ sees car's actions
5. $w \models\!\!\!\not\; V_3^d(w)$ — $driver$ can't tell it's $tc$
6. $w \models\!\!\!\not\; V_4^d(w)$ — is it $corp.$?
7. $w \models\!\!\!\not\; V_5^d(w)$ — is it broken?
8. Combining statements 3, 6, and 7 we obtain

$$MSDND(ES) = \exists w \in W : \left[w \vdash \Box\left(\sum_{i=2}^{4} control_i = 1\right)\right] \wedge \left[w \models (\not\models V_4^d(w) \wedge \not\models V_5^d(w))\right]$$

■

*Remarks about applying the Multiple Security Domains Model:* From the physical actions of the car, it is correct to deduce that the driver is not in control. What is in control is MSDND(ES) secure from the driver. Hazardous Conditions (traction control), Remote Corporate Operations, and possible mechanical failure all present the same way to the driver and passengers. The longer this situation continues the more likely it is that something bad will happen.

## VI. Conclusion

The traditional view of security, the idea of "walling the bad guys out", is too simplistic. Viewing security domains as wholly contained within a threat space or within a less secure domain is inadequate as are the tools available. Restricting models to idealized partitions does not work well with cyber physical systems.

We have shown multiple security domains, without the necessity of ideal partitions, is a more realistic model. We have shown that in CPS information leaks throughout the model by observation of the physical actions of the system. Our new definition of MSDND(ES) can model traditional Nondeducibility as well as provide a definition of Nondeducibility that holds in CPS. Specifically, MSDND(ES) can easily model situations where critical information flow from one security domain to another is disrupted or denied altogether as in the Stuxnet worm attack.

We applied our model to a specific cyber-physical system, a drive-by-wire automobile, under real world conditions. Our model fits the CPS better than traditional Nondeducibility because it does not require us to partition the system into idealized domains that do not allow information flow between domains. Indeed, our model does not even need to address how the security domains interact once they have been properly defined. We have shown that we can relax the requirements of absolute domain partitioning and still model the system.

Furthermore, we have shown that since MSDND(ES) does not depend upon the ability to evaluate information flow between distinct and absolute partitions, our model does not require building complicated decision variables nor does it require access to the total input/output of the model. By relaxing the boundary conditions of the model, results are obtained by modal methods.

## Acknowledgments

**This work was supported in part by the Future Renewable Electric Energy Distribution Management Center; a National Science Foundation supported Engineering Research Center, under grant NSF EEC-0812121, and in part by the Missouri S&T Intelligent Systems Center.**

## References

[1] D. Sutherland, "A model of information," in *Proceedings of the 9th National Computer Security Conference*. DTIC Document, 1986, pp. 175–183.

[2] J. McLean, "Security models and information flow," in *Research in Security and Privacy, 1990. Proceedings., 1990 IEEE Computer Society Symposium on*. IEEE Computer Society Press, May 1990, pp. 180 –187.

[3] G. Howser and B. McMillin, "Modeling and reasoning about the security of drive-by-wire automobile systems," in *International Journal of Critical Infrastructure Protection*. Elsivier, 2012.

[4] K. Poulsen, "Hacker Disables More Than 100 Cars Remotely," http://www.wired.com/threatlevel/2010/03/hacker-bricks-cars/, 2010, [Online; accessed 3-October-2011].

[5] T. Roth and B. McMillin, "Breaking nondeducible attacks on the smart grid," in *Seventh CRITIS Conference on Critical Information Infrastructures Security*, Seventh CRITIS Conference on Critical Information Infrastructures Security. Springer, 2012, (to appear).