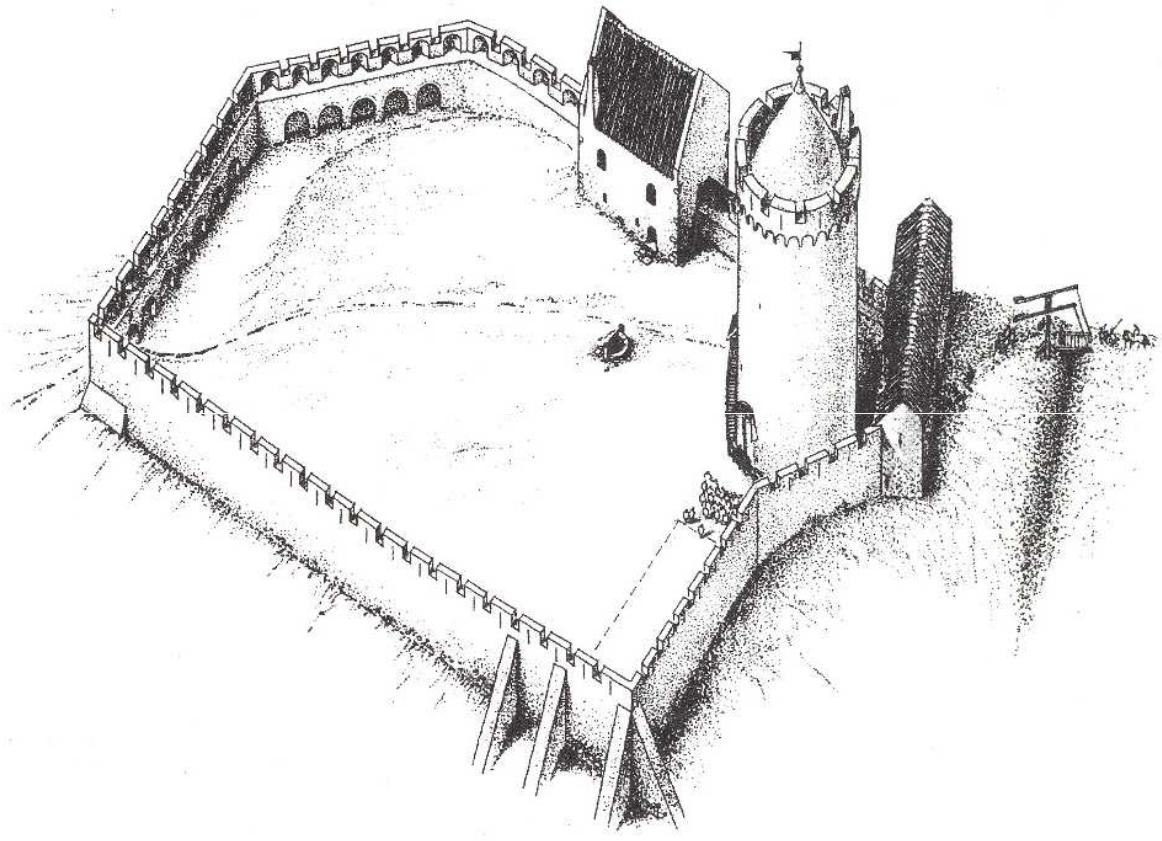
Missouri University of Science and Technology

Founded 1870 | Rolla, Missouri | www.mst.edu

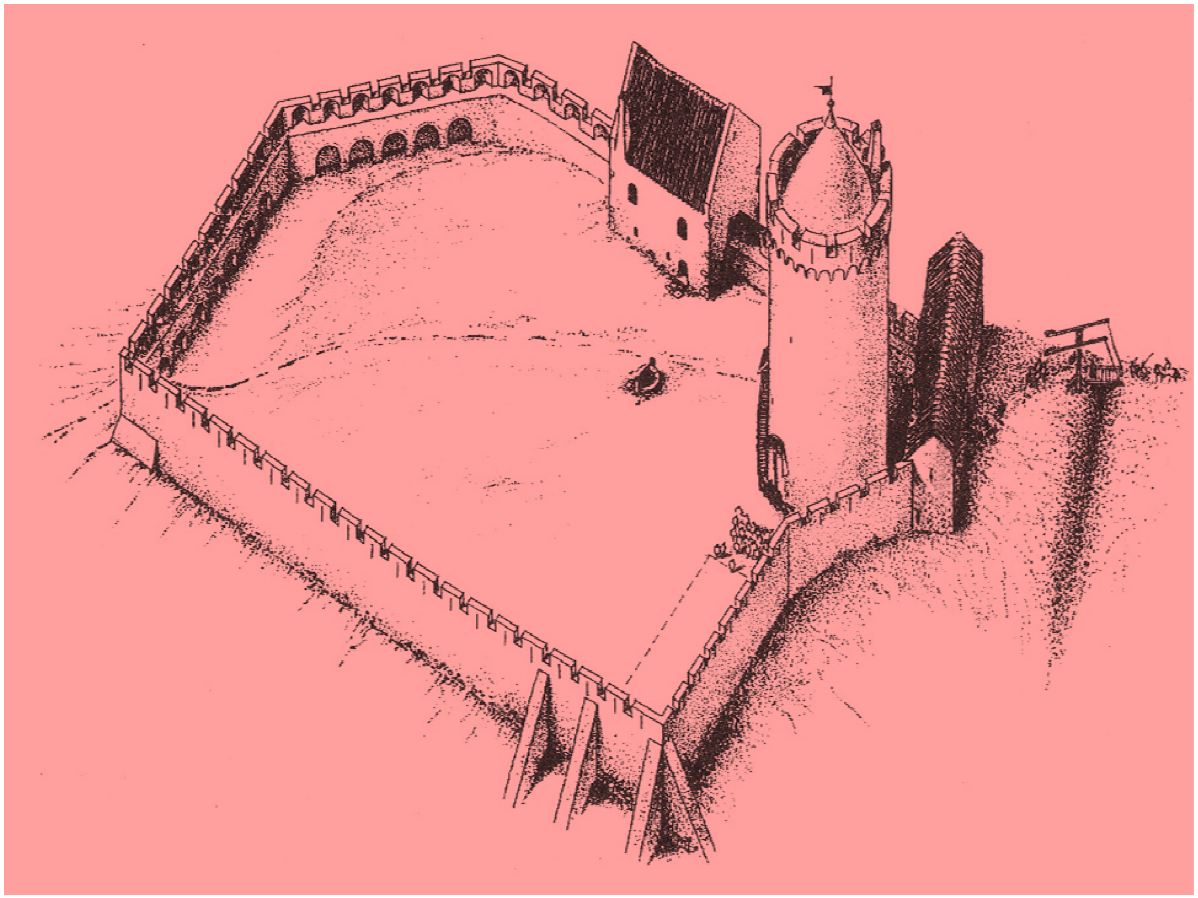
A Multiple Security Domain Model of a Drive-by-Wire System

Gerry Howser and Bruce McMillin

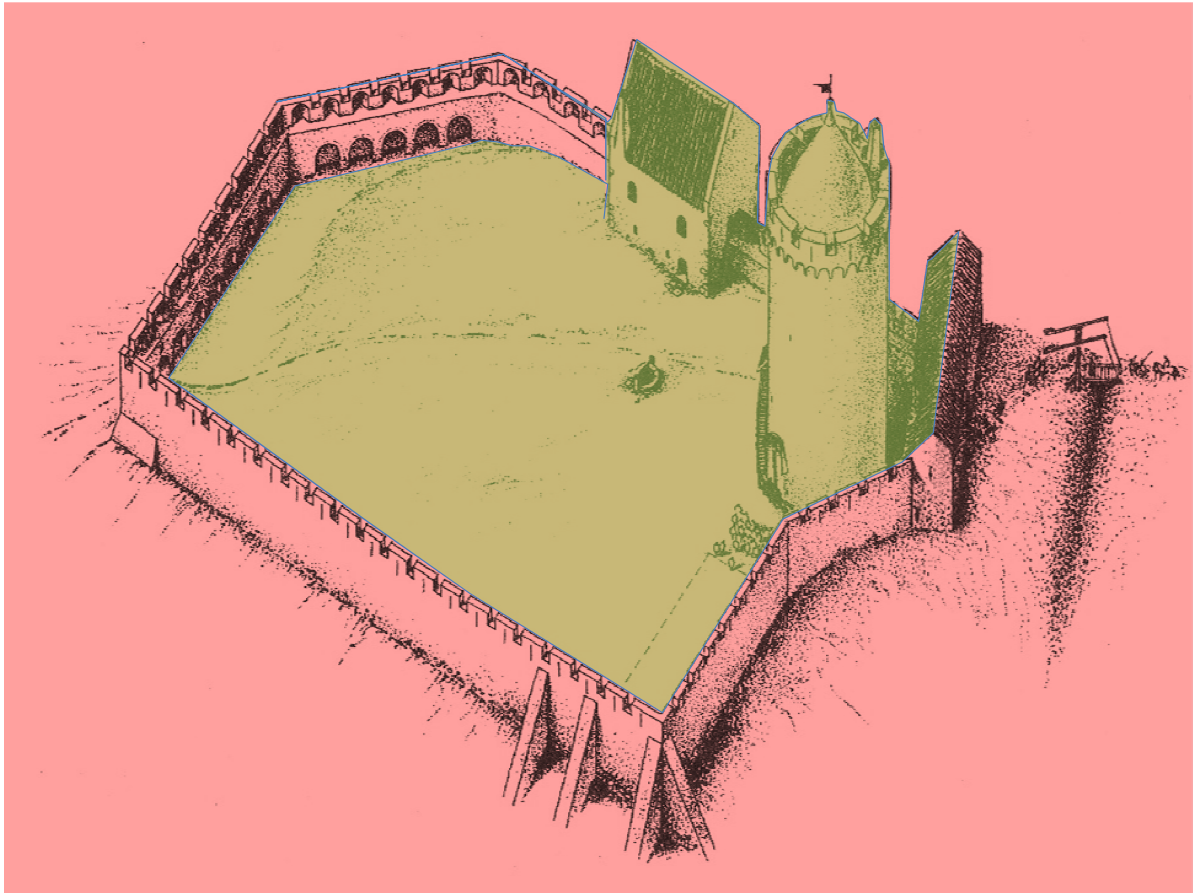
The 'Fortress Mentality'



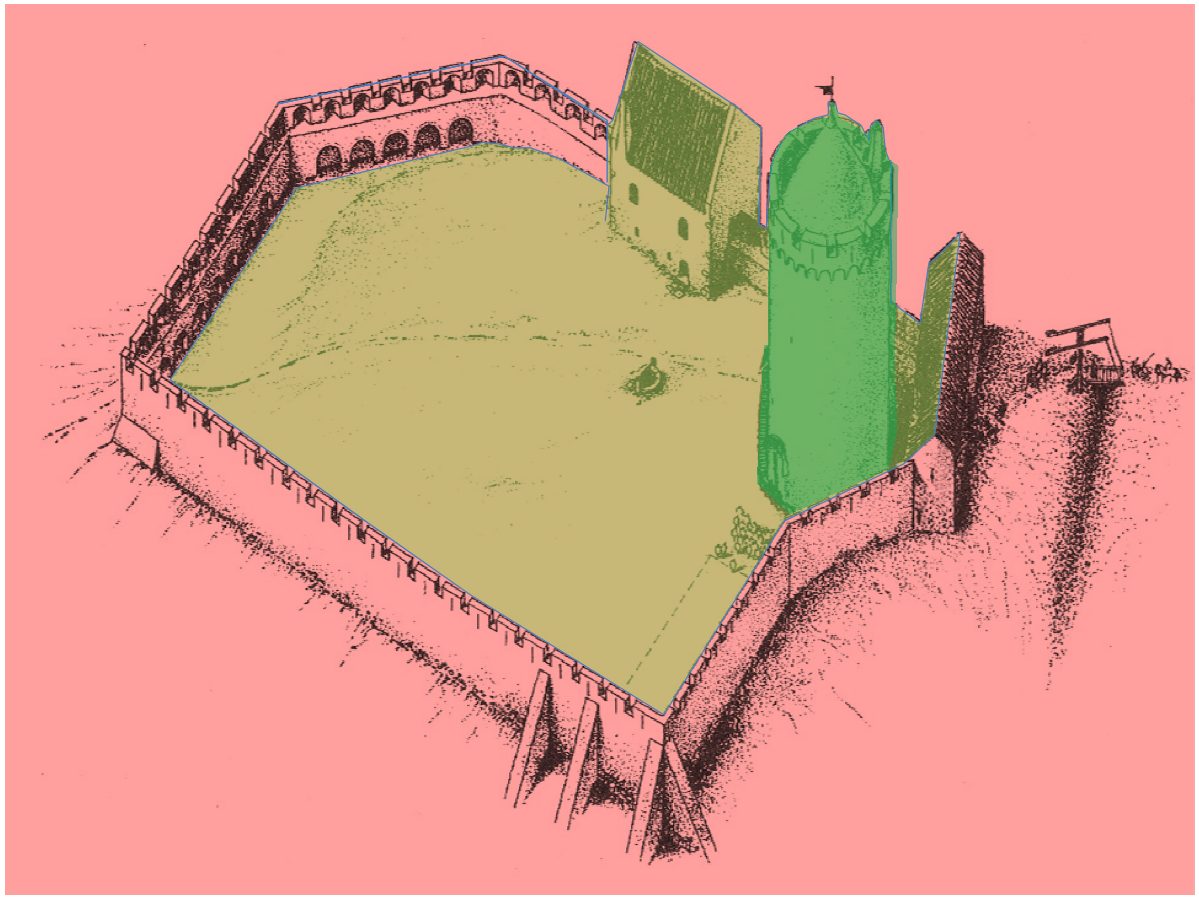
The 'Fortress Mentality'



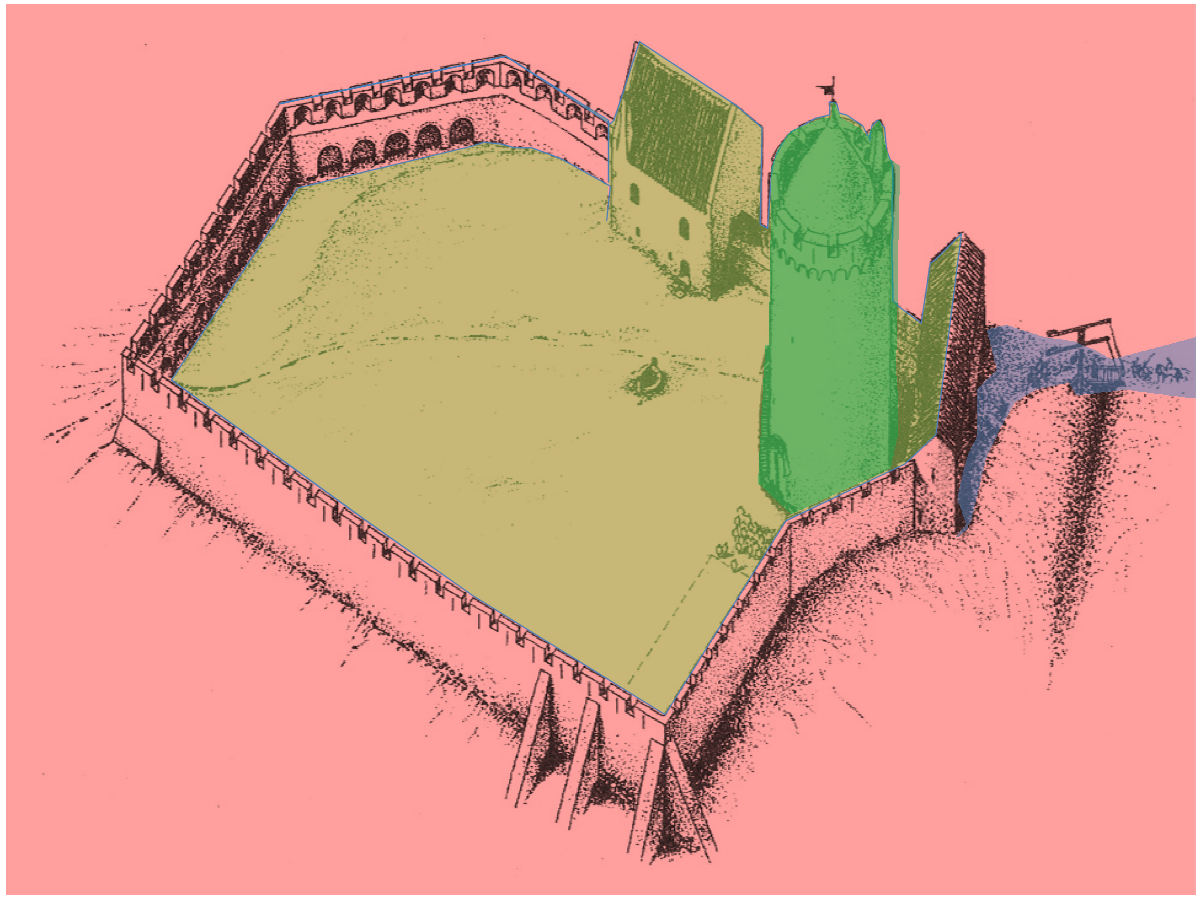
The 'Fortress Mentality'



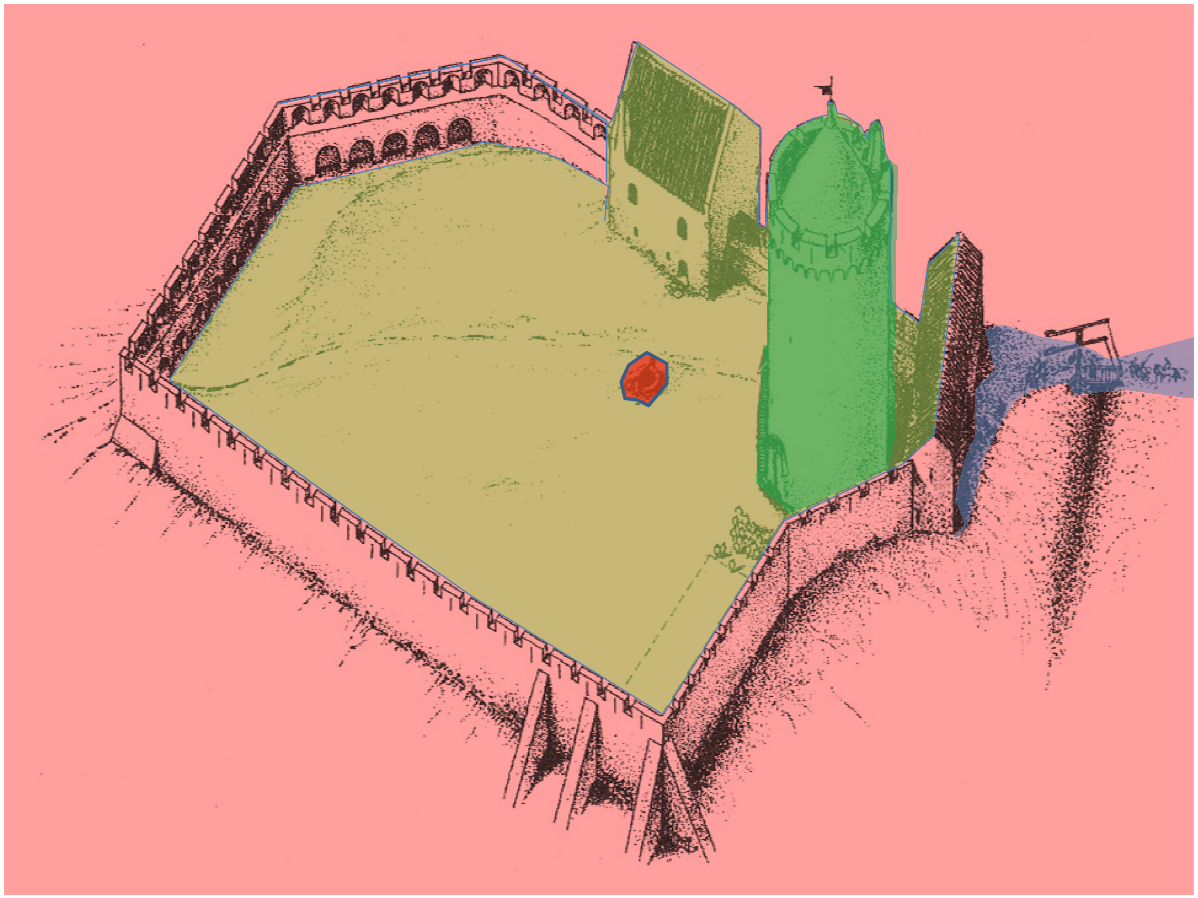
The 'Fortress Mentality'



The 'Fortress Mentality'



The 'Fortress Mentality'



Traditional "Onion" Security

- Security domains wall them out
- "HIGH" and "LOW"
- Domains are contained within other domains
- Objects and Subjects are placed within domains

Cyber-Physical Systems (CPS)

- Cyber System (traditional)
- Physical System (observable)
- Combination leads to complex systems
  - Characterized by Information Flows
  - Physical system leaks information about the Cyber system

Typical CPS Security Domains

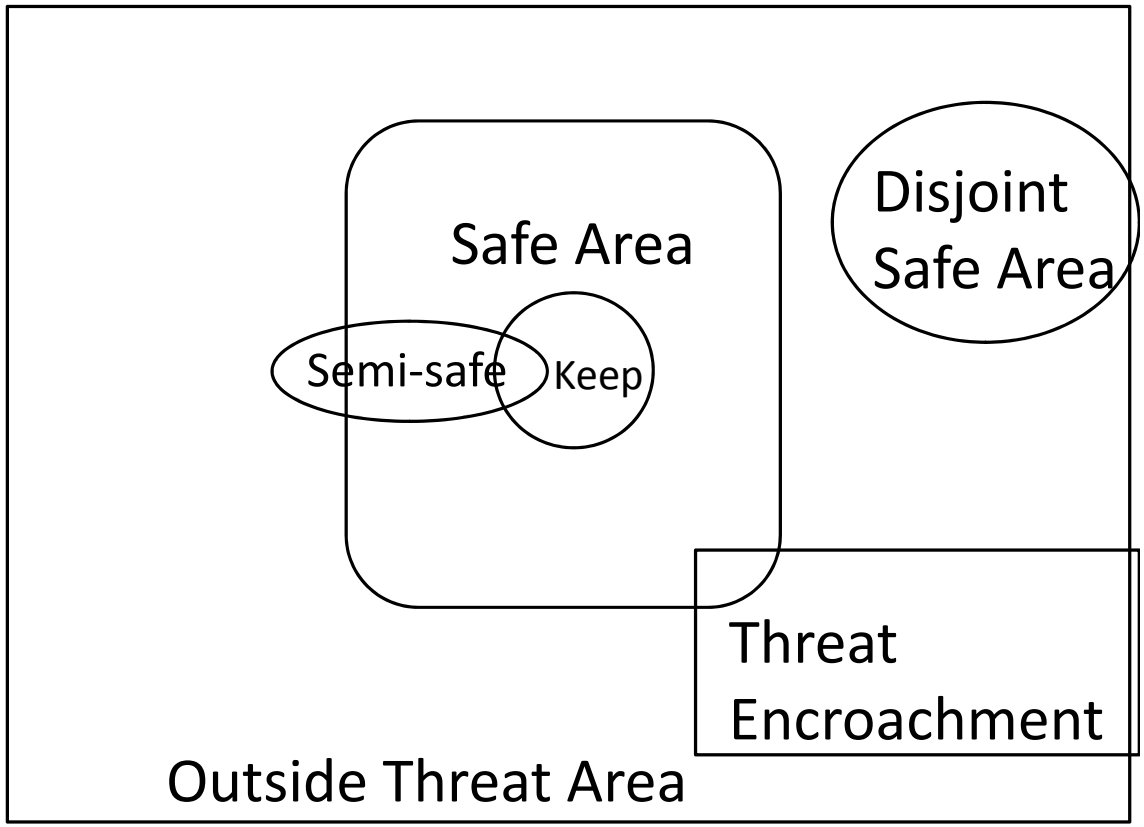


Figure : Problematic Overlapping Security Domains

The Problem

- Traditional security does not work with Cyber-Physical Systems
- CPS lead to security domains too complicated for traditional security
- Information is *always* leaked by observing the physical side
- Domain partitions are not ideal (more later)

Kripke Frames, Models, and Valuation Functions

- Each combination of event variables characterizes a node or world $w \in W$
- Any change in an event variable, or relationship R, moves to a different node, or world
- The nodes and the connecting events form a framework or Kripke Frame $(\mathfrak{F}(W, R))$
- Event variables can be queried via Valuation Functions
  - *Only those event variables with Valuation Functions can be queried*
  - Queries forming a set, Φ, will be denoted by φ and ψ
  - Valuation Functions for Object/Subject d will be denoted by V_i^d
  - A Model, M, can be formed over the frame as $M(\mathfrak{F}, V_i^d)$

Modal Queries

Suppose we have a set $\varphi, \psi \in \Phi$ 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 φ is a wff, so are $\neg\varphi$, $\Box\varphi$, and $\Diamond\varphi$
- if φ and ψ are wff, then so is $\varphi \vee \psi$
- if φ and ψ are wff, then so is $\varphi \wedge \psi$
- As usual, other classical logical operators $\wedge$ (and), $\Rightarrow$ (material implication), $\oplus$ (exclusive OR) and $\Leftrightarrow$ (if and only if), can be defined as abbreviations.
- We define the modal operator, $\Box\varphi$, as an abbreviation for $\neg\Diamond\neg\varphi$.

The Axiomatic System

1. Definition of logical and modal operators (abbreviations)
   - D1: $\varphi \wedge \psi \equiv \neg(\neg\varphi \vee \neg\psi)$ **AND**
   - D2: $\varphi \oplus \psi \equiv (\varphi \vee \psi) \wedge \neg(\varphi \wedge \psi)$ **(Exclusive OR)**
   - D3: $\varphi \Rightarrow \psi \equiv \neg\varphi \vee \psi$
   - D3: $\varphi \Leftrightarrow \psi \equiv (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$
   - D4: $\Diamond\varphi \equiv \exists w \in W : w \vdash \varphi$
   - D5: $\Box\varphi \equiv \neg \Diamond \neg\varphi$
2. Axioms
   - P: all the tautologies from the propositional calculus
   - K: $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$
   - M: $\Box\varphi \Rightarrow \varphi$
3. Rules of Inference
   - R1: from $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer ψ **(Modus Ponens)**
   - R2: $\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$ **(DeMorgan's)**
   - R3: from $\vdash \varphi$, infer $\vdash \Box\varphi$ **(Generalization)**
   - R4: from $\vdash \varphi \equiv \psi$, infer $\vdash \Box\varphi \equiv \Box\psi$

A Modal View of Sutherland Nondeducibility ND(ES)

- Relies on valuation functions
- Usually, the system is partitioned into two domains *high* and *low*
- From a modal viewpoint, this leads to at least two valuations, $V_1(w)$ and $V_2(w)$
- Two events are Nondeducibility secure if, and only if:

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

- If we can evaluate $V_1(w)$ and $V_2(w)$ for all $w \in W$, we can determine if Nondeducibility holds
- If we cannot evaluate $V_1(w)$ and $V_2(w)$ for all $w \in W$, we cannot determine if Nondeducibility holds

Multiple Security Domain Nondeducibility MSDND(ES)

- Relies on valuation functions
- Usually, the system is not partitioned, but it can be
- The Event Systems is composed of security domains $SD_i : \bigcup_{i \in I} SD^i = (ES)$
- From a modal viewpoint, this leads to at least two valuations, $V_1(w)$ and $V_2(w)$

Multiple Security Domain Nondeducibility MSDND(ES) continued

- **Definition of MSDND(ES):** 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.
- The modal formulation of this definition is:

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

- An equivalent formulation would be:

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

Reduction of Sutherland ND(ES) to MSDND(ES)

Theorem

Any arbitrary case where ND(ES) holds can be shown to be a special case of MSDND(ES)

Given: A system with two security domains, *left* and *right*, and two arbitrary 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 $sf \Rightarrow (w = w'')$.

Reduction of Sutherland ND(ES) to MSDND(ES) (continued)

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
$\nVdash V_{st}^{left}(w)$	ND(ES)
$\nVdash V_{sf}^{left}(w)$	ND(ES)
$w \vDash \left[\nVdash V_{st}^{left}(w) \wedge \nVdash V_{sf}^{left}(w)\right]$	ND(ES)

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

$$MSDND(ES) = \quad \exists w \in W : w^{''} \vdash \Box(st \oplus sf)$$
$$w \vDash (\nVdash V_{st}^{left}(w) \wedge \nVdash V_{sf}^{left}(w))$$

Therefore, if Nondeducibility holds, MSD Nondeducibility holds as well.

THE CYBER-PHYSICAL SYSTEM

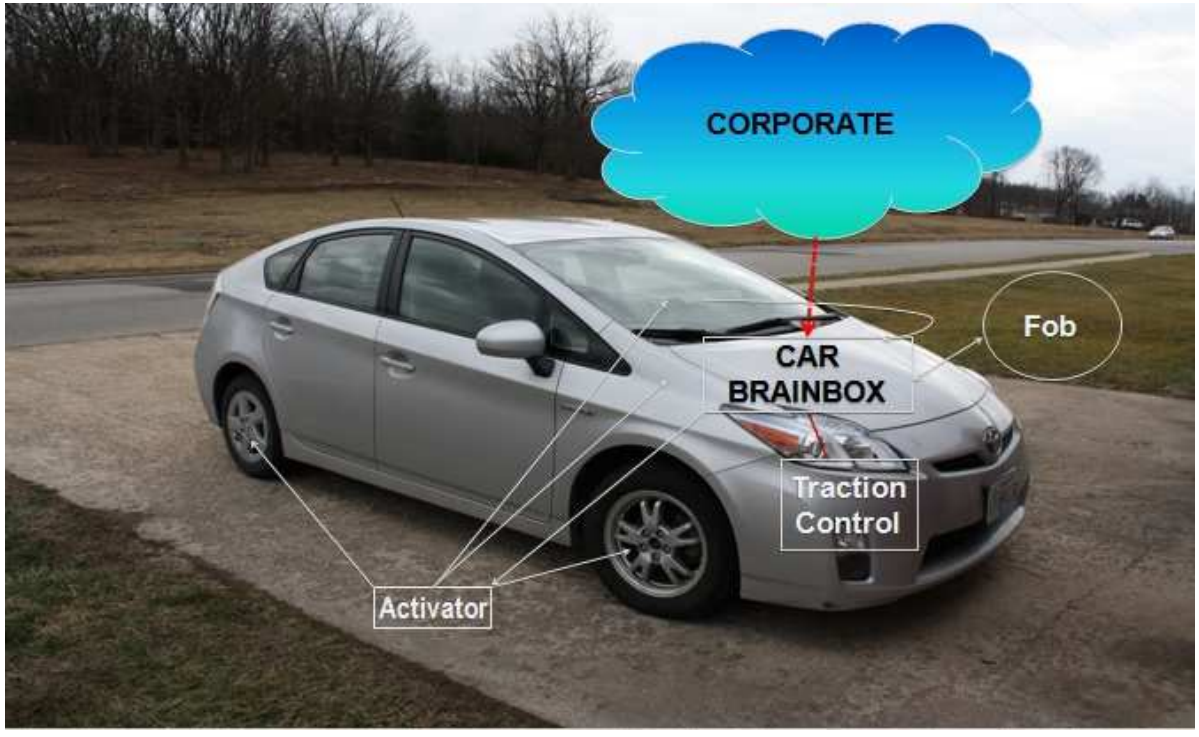


Figure : The Network Enabled Prius

Concern

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?
(Howser-McMillin 2010)

Modes of Operation for the System

Of interest are three different modes of operation

- Normal driving
- Hazardous road conditions when traction control takes over
- Remote control of the vehicle (hopefully by OnStar, ToyotaConnect or something similar)

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

Logical Statements of Interest

φ_i	state	
φ_0	s_0	The car is behaving normally
φ_1	s_1	*driver* is aware of car's behavior
φ_2	s_2	The *driver* is in command
φ_3	s_3	Traction Control is in command
φ_4	s_4	The *corp* is in command
φ_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	$f = \top$	$f = \neg\varphi_2 \wedge \neg\varphi_3 \wedge \neg\varphi_4 \wedge \varphi_5$

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

$$V_i^t(w) = s_i$$
$$V_i^c(w) = s_i$$

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

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:

$$
\begin{aligned}
w \vDash d \quad &\Leftrightarrow \quad w \vdash \Box\neg(t \vee c \vee f) \\
w \vDash t \quad &\Leftrightarrow \quad w \vdash \Box\neg(c \vee f \vee d) \\
w \vDash c \quad &\Leftrightarrow \quad w \vdash \Box\neg(f \vee d \vee t) \\
w \vDash f \quad &\Leftrightarrow \quad w \vdash \Box\neg(d \vee t \vee c).
\end{aligned}
$$

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

$$
\begin{aligned}
w \vDash d \quad &\Leftrightarrow \quad control_d = control_1 = 1 \\
w \vDash t \quad &\Leftrightarrow \quad control_t = control_2 = 1 \\
w \vDash c \quad &\Leftrightarrow \quad control_c = control_3 = 1 \\
w \vDash f \quad &\Leftrightarrow \quad control_f = control_4 = 1
\end{aligned}
$$

$$
w \vdash \Box\left(\sum_{i=1}^{4} control_i = 1\right).
$$

Hazardous Road Conditions

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.

Hazardous Conditions and the Sutherland Model

Using the worlds, states, and evaluation functions we have previously defined we see:

$$
\begin{array}{lllll}
V_2^d(w_3) & = & V_2^d(w_5) & = & (s_2 = \bot) \\
V_3^d(w_3) & = & V_3^d(w_5) & = & \top \\
V_5^d(w_3) & = & V_5^d(w_5) & = & \top \\
V_3^t(w_3) & \neq & V_3^t(w_5) & & \\
V_5^t(w_3) & \neq & V_5^t(w_5) & &
\end{array}
$$

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

$V_2^t(w_3) = V_2^t(w_5) \wedge (V_3^t(w_3) \neq V_3^t(w_5))$

From the viewpoint of the $driver(left)$:

$$
\begin{aligned}
& V_2^d(w_3) = V_2^d(w_4) = V_2^d(w_5) \\
& \wedge (\nexists\, V_3^d(w)) \wedge (\nexists\, V_4^d(w)) \\
& \wedge (\nexists\, V_5^d(w))
\end{aligned}
$$

Hazardous Conditions and the MSDND(ES) Model

Given: The *driver* knows something else is controlling the car and the constraint 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. $w \vDash\!\!\!\!/\ V_3^d(w)$ — *driver* can't tell it's *tc*
6. $w \vDash\!\!\!\!/\ V_4^d(w)$ — is it *corp.*?
7. $w \vDash\!\!\!\!/\ V_5^d(w)$ — is it broken?
8. Combining statements 3, 5, and 7 we obtain

$$MSDND(ES) = \exists w \in W : \left[w \vdash \Box\left(\sum_{i=2}^{4} control_i = 1\right)\right] \wedge \left[w \vDash (\not\vDash V_3^d(w) \wedge \not\vDash V_5^d(w))\right]$$

Remarks about ND(ES) and MSDND(ES) During Hazardous Conditions

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.

Remote Operations

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 we see:

Remote Operations and the Sutherland Model ND(ES)

$$
\begin{array}{lcll}
V_2^d(w_4) & = & V_2^d(w_5) & = (s_2 = \bot) \\
V_4^d(w_4) & = & V_4^d(w_5) & = \top \\
V_5^d(w_4) & = & V_5^d(w_5) & = \top \\
V_4^c(w_4) & \neq & V_4^c(w_5) & \\
V_5^c(w_4) & \neq & V_5^c(w_5) &
\end{array}
$$

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*):

$V_2^d(w_3) = V_2^d(w_4) = V_2^d(w_5)$
$\wedge(\nexists\, V_3^d(w)) \wedge (\nexists\, V_4^d(w))$
$\wedge(\nexists\, V_5^d(w))$

Remote Operations and the MSDND(ES) Model

Given: The *driver* knows something else is controlling the car and constraint from slide 21 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

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

Remarks about Remote Operations

From the physical actions of the car, it is correct to deduce that the driver is not in control. What is in control is both ND(ES)and 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.
From the viewpoint of the driver, it would be better if Nondeducibility did not hold for hazardous conditions and *Company* remote operations. The driver has no control in either case and cannot determine the source of the strange actions of the *Car*. The driver may not be able to turn off the car and coast to a safe stop. Some newer cars shift into park when turned off. Even worse, *driver* cannot evaluate the fault state, s_5, of the car at all! How does the driver determine if the car is under traction control, remote operations, or it has been hacked?

Conclusions

Multiple Security Domains NonDeducibility MSDND(ES):

- Produces comparable results when ND(ES) holds
- Produces results where ND(ES) does not hold
- Does not rely on idealized partitions
- Can describe systems with Onion partitions
- Does not rely on the structure of the partitions
- Holds where there are no domains

Acknowledgements

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.

Questions

Questions????