Proceedings of the 12th Colloquium for Information Systems Security Education
University of Texas, Dallas
Dallas, TX June 2 - 4, 2008

Teaching of Security in Cyber-Physical Systems

Ravi Akella, Bruce McMillin, Travis Service
Department of Computer Science
Intelligent Systems Center
Missouri University of Science and Technology
{rcaq5c, ff, tcsvw5}@mst.edu

Abstract – This paper describes the results of applying formal security models to Cyber-Physical systems work in a classroom setting. The structure of the course required that each student select an infrastructure that had significant cyber and physical components. During the course, when they learned a model, they applied it to their infrastructure. Formal models included the HRU, Take-Grant, Bell-LaPadula, Biba, Non-interference, Non-inference, and Non-deducibility. The approach is described, results of the models, and student feedback are reported.

Index terms – Security Education, Cyber-Physical systems, Security Models, Semantic domain knowledge.

I. INTRODUCTION

Controlling access to sensitive information is crucial to the security of any organization. Information security can be decomposed into three basic categories:

- Confidentiality, the prevention of unauthorized access to information,
- Integrity, the prevention of unauthorized or improper modifications to existing data,
- Availability, the assurance that information is accessible by authorized entities.

These categories are not mutually exclusive as a loss in confidentiality can often times lead to a loss in integrity and/or availability. Many different security models have been proposed to help address the concerns of confidentiality, integrity and availability.

Information security is often applied to purely information (Cyber) systems. Cyber-Physical Systems (CPS) [1], by contrast, are the integration of computational capabilities with real-world physical processes. When we use computers (mostly embedded) and networks to control such physical systems, transformation of the computational data to the physical terms and vice-versa plays a vital role. The interaction between the cyber and physical worlds uncovers complex interactions that go beyond those found in the purely computer world.

Supported in part by NSF MRI award CNS-0420869, NSF CSR award CCF-0614633, and the UMR Intelligent Systems Center.

The security requirements of a CPS depend on the cyber information flow, physically observable behavior, and the interactions between [2, 3]. Take, for example, the confidentiality of process control in a Steel foundry. By simple observation by an External Observer (through a window of the plant, perhaps) the movement of hot steel, heating schedules, and cooling schedules can be observed to deduce the (presumably confidential) process control. Of interest is how to formally capture this interplay. Formal security models provide a basis for information systems, but how do they capture cyber-physical interactions?

This paper describes a class project undertaken in CS 483, 'Advanced Computer Security' held at the Missouri S&T. As a part of the course different CPS applications are chosen and their security is investigated in light of existing formal security models. The access control methods of HRU, Take-Grant, Bell-LaPadula and information flow models of Non-Interference, Non-Inference and Non-Deducibility are considered [4, 5]. In doing so, the students explore the commonalities and differences in existing security models as applied to different scenarios of Cyber-Physical Systems.

This is an advanced graduate level class with pre-requisites of Networking and CNSS 4011 material. The material had both a research component and an educational component. Thus, the class had two objectives.

1) The research component was to treat a variety of infrastructures that have significant cyber and physical components and explore, primarily, rights leakage and information flow leakage and identify commonalities, differences, and challenges in applying each model to a variety of infrastructures

2) The educational component was to teach formal security models by giving each student a unique experience beyond abstract models and purely cyber models.

We used Matt Bishop's book [4] (chapters 1-8) to guide the course using problems from the text to explain the appropriate security models. Each of the seven models

was applied during the semester, roughly spaced every two weeks. Students received feedback on their models until their analysis applied each model correctly. The resulting assignments were each compiled into a term paper. For the HRU and T-G models, we used Mike Collin's Java applets [6] to help debug the application of the models and express them clearly.

The remainder of this paper is organized as follows. Section II surveys the infrastructure models used. Section III surveys the security models used and Section IV shows the results of application of the models to each infrastructure. Section V describes the results of the educational experience. The infrastructure descriptions, security model descriptions, and results of application of models all follow the students' lines of reasoning from their class projects.

II. Infrastructure Survey

The following infrastructures were presented to the students as brief topic descriptions. The students then researched existing infrastructures to develop a rich enough system to analyze. Interaction with the instructor aided in this process so that the system would have security issues of interest.

A. Transportation Networks

An example transportation network, the natural gas transport system is modeled as a network of pipes. On a subset of these pipes are Remote Terminal Units (RTUs), used to monitor and affect changes to the state of the gas within the pipe on which they sit (such as pressure). A number of operators sit in front of terminals called Human Machine Interfaces (HMIs). Each HMI is connected to some subset of the RTUs out in the field and are used as an interface between the human operators and the RTUs. In this way a given operator has some level of control, and monitoring ability, over some subset of the pipes in the transport network.

A commonality was found to exist between this infrastructure and some other infrastructures we considered including:

- *Telesurgery* in which a remote connection has to be made in the public domain over which private patient information and surgical/medical data can be securely transferred,
- *Oil Drilling Rig* in which data received from an oil well to a company headquarters may be confidential but the actions of the drilling company may subtly divulge its intentions simply by exploring the region using a drill and
- *Automated Air Traffic Control System* in which there are several security issues, including confidentiality (of the information about passengers and fight control), integrity (of information which is susceptible to modifications at different airports on the route) and availability (authorized ground control station that should have access to the airplane information).

B. Chinese wall Security in an Organization

The Chinese wall model is a model of security policy that refers equally to confidentiality and integrity. This commercial security policy was proposed in [7]. In terms of information flow, we can classify it as a Multi lateral Security policy. It is generally employed in environments like commercial or financial services firms such as investment banks who typically consult for different clients. In many cases, the clients share common interests as they may be market competitors. The Chinese wall policy attempts to uphold the confidentiality of information provided by the clients to the consulting firms by preventing such conflicts of interest which might be called as "Chinese wall". The data is organized as *Objects, Company dataset* (contains objects related to a single company) and *Conflict of interest class* (contains datasets of companies in competition). We considered a model problem to analyze the Chinese wall policy with *Conflict of interest classes* [COIs]: Banks, Oil Companies and *Company Data Sets* [CDs]: BOA, CITI, SHELL, ARCO and *Subjects*: Auditor, Analyst1, Analyst2, External Observer.

The problem is to assign to each Analyst with one or more clients such that confidentiality of each client's data is retained. In other words, he cannot be allowed to handle a client where he has some insider knowledge of its competitor.

C. Automated Vehicle Traffic Control

In platooning for Automated Vehicle Traffic Control (AVTC) systems [8], a platoon is a caravan of vehicles, where each vehicle's actions are largely controlled by the vehicle directly in front of it. Platoons are helpful with relieving road congestion as cars are able to travel in tight packs minimizing the delay between stopping and going.

To demonstrate the security issues in AVTC system, we define the security policy as an integrity policy: *no two cars may be in the same position at the same time*. The AVTC system has temporal and spatial aspects that we need to consider. We will view the time as a set of discrete time points t_1, t_2, …. and space as a set of discrete points in two dimensions. Now we can define the security mechanism formally as: A car can enter a position (x_i, y_j) at time t_k if position (x_i, y_j) is unoccupied by any other car at time t_{k-1}. With this security mechanism in place, the

security policy can be intentionally violated by a malicious driver if he has the information about the immediate future platoon behavior. Such information is passed around via secret communication between cars in a platoon.

D. Engine control system

Engine control systems are not typically thought of in the matter of security, but there are complex security issues that are inherent with the architecture of its design. Figure 1 shows a brief overview of the components of an engine control system.

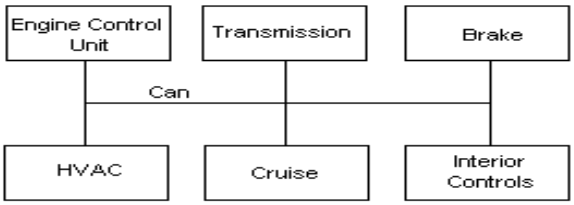


Figure 1. Design of the Engine system.

The engine control unit is responsible for a bulk of the work load. Since it cannot manage all facets of control of the entire engine system it must rely on the other components to pick up parts that it is unable to manage. The most important of these systems are the braking, transmission, and gas flow. Since there is some synchrony that is required to achieve optimal engine response and power output a fast but reliable communication bus is used. The most common bus that is used is the CAN bus. Since all of the components have access to this there are inherent security flaws due to component interactions that cannot be avoided. To show these flaws, the engine control system is modeled with respect to the security architecture of the system.

E. Inter-organization information flow

The infrastructure considered was the supply chain management structure of a car manufacturing company. The car manufacturing company has several entities within it and the corresponding entities outside the organization. The entities within the organization are data manager, parts manager, production manager, data center, parts data, production data, order, loading dock and finances. The entities outside the organization are the engine supplier, metal supplier, glass supplier, and an electrical supplier and the databases related to each of these entities. Each of these companies also has access to the loading dock of the car manufacturing company. An External Observer can observe all the operations taking place in this infrastructure. In such an environment, there can be some serious right leakages which might result in the External Observer gaining some unwanted rights over confidential information of the organization. The main goals of security are Prevention of attacks and leakages, Detection of attacks and leakages, and Recovery of damaged or lost information [9]. This applies strongly to this infrastructure.

We also analyzed the process in a *Steel foundry* which has a similar infrastructure. The similarity makes us group these two infrastructures together, under inter-organization information flow.

F. Disaster Responders

In a disaster response situation, there are many wounded people in the affected area and there is an individual medical record for each of those patients. The HIPAA (Health Insurance Portability and Accountability Act) regulation states that the privacy of the patients should be maintained and therefore only doctor and the patient should have the necessary access to that record. HIPAA even describes more specifically on the rights that the patient and the doctor should have over the medical record [10]. The patient will have just the read access to that record, however they may decide whether that record be made public or not. The doctor however has write access to the record so that those records may be updated from time to time. In the absence of the patient, a family member of that patient can grant the access of the patient's record to others. The scenario described is one of the information flow problem because here the information about a patient is passed around the network.

We also examined similar information leaks with respect to a *Smart house, Farm intelligence system and Bear tracking system*. Though they had a different infrastructure, similar problems were posed with their setup and the way information flows through the system.

III. Security Models

Security models are concerned with access control to restrict what operations can be performed on objects or information flow that restrict what subjects can infer about objects by observing the system behavior. In this section, we shall survey some of several such security models [6]. We give the informal definitions, here, for brevity, but the students worked with formal definitions.

A. Harrison-Ruzzo- Ullman (HRU) Model

The Harrison-Ruzzo-Ullman (HRU) model [11] represents a state of the security system as a set of subjects, objects, and a set of rights subjects have over objects, defined as an Access Control Matrix (ACM). The state of the system changes when commands are executed. A system is safe with respect or a right **r** if *no*

subject that does not have right **r** initially gets right **r** after a sequence of commands. It is undecidable to test whether a given system is safe with respect to a generic right **r**.

B. Take-Grant

In the Take-Grant protection model there are only two possible ways of passing rights between the entities (where entities are subjects and objects) of the system: with *take* or *grant* commands [4]. The rights that entities have over each other are represented as a digraph, where each vertex is an entity, and a directed edge **(a, b)** with a set of labels **Z** represents that **a** has rights in **Z** over **b**. Here, if **Z** contains **t**, which stands for *take*, then **a** can acquire any right that b has over any other entity if **a** is a subject. If **Z** contains **g**, which stands for *grant* then **a** can grant **b** with any right **a** has over any other entity, if **a** is a subject. *Theft* in the T-G model is a specific case where two subjects can share a right, but the subject originally holding the right does not grant it, it is taken. *Conspiracy* in T-G model builds on the idea of right sharing, creating a new graph showing the paths along which rights can be transmitted in the system. With the take-grant model it is always possible to determine if the system is safe with respect to any generic right **r** .

C. Bell-LaPadula

The Bell-LaPadula (BLP) model is a multi level security model that corresponds to a military style classification for preserving the data confidentiality. In BLP security levels are defined such that subjects with lower security levels can not read from objects of higher security levels (``no read up" rule) and subjects of higher security levels can not write to objects with lower security levels (``no write down" rule). Having such rules prevents the leakage of ``secure" information to lower levels.

D. Biba

The Biba model is not concerned with the confidentiality of information, as the BLP model, but rather with the integrity of information. Just like the BLP model, the Biba model defines levels, only here these are *integrity* levels, as opposed to security levels of BLP. The rules of the Biba model are the opposite of the rules of the BLP model: no subject can read from an object of a lower integrity level (``no read down" rule) and no subject can write to an object of a higher integrity level (``no write up" rule).

E. Non-Interference

Non-interference [12] is an information flow security model. A system is non-interference secure if the low level projection of removing all high level events from a command sequence of finite length does not change the projection of the original low-level output. If the output gets changed, then the high-level events interfere with the low level outputs and the system is not non-interference secure. In this way the high level events interfere with the low level observation.

F. Non-Inference

A system is considered non-inference secure if and only if for any legal trace of system events, the trace results from the legal trace purged of all high-level events is still a legal trace of the system. A system that is non-inference secure leaves a low level observer in doubt about high level events.

G. Non-Deducibility

A system is considered non-deducible secure if it is impossible for a low-level user, through observing visible events, to deduce anything about the sequence of inputs made by a high-level user. In other words, system is non-deducible secure if the low-level observation is compatible with any of the high-level inputs. Thus, while a low level observer knows that high level events are occurring, they cannot tell which events they are.

IV. Analysis and results

The results of applying each security model to every infrastructure have been tabulated as shown in Table 1.

HRU & T-G
We were able to model every infrastructure in terms of the HRU and the T-G models. With a diversified type of entities in each infrastructure and the execution of carefully chosen commands, both the models provided a robust access control framework; but then, almost every infrastructure suffered an unintended leakage of rights leading to a compromised system.

One challenge in modeling Cyber-Physical systems' confidentiality was the notion of how an attacker could discover knowledge about the system if they knew something about the physics of the system, or, in general about its semantics. Previous research work in [2] showed that electric power system knowledge could violate non-deducibility for a power flow controller. An abstraction developed during the class is the object "SDK" or Semantic Domain Knowledge. In the T-G model, this is represented by an Object that a Subject takes rights from.

Consider the case of our Gas Transport system. Our subjects consist of all the system operators, HMIs, RTUs and pipes. Our ACM is initialized in the following manner. An operator O who is in control of some HMI *H*

is initially given read, write and own rights over that *H* and *H* is given read and write rights over O. All RTUs connected to the HMI *H* are given read and write rights over H and H is given read, write and own rights over all such RTUs. Finally, all pipes on which one such RTU sits are given read and write rights over the RTU placed on them and that RTU is given read, write and own rights over that pipe. In our model pipes consist of the values, compressors and various other devices which control the state of the gas within them, and as such need to be written to, to control these devices.

The topology of the system is encoded into the ACM through the use of a special connected right, c. If pipe *i* and pipe *j* are connected at some junction point then pipe *i* has c rights over pipe *j* and vice versa. In this manner the individual operators have indirect control over their pipes. Communication goes through a HMI-RTU path before reaching the individual system pipes. An operator can obtain access to the pipes other than the ones he was intended to monitor/control by knowing the state of pipes in the system. The operator does this through a basic understanding of the concept of conservation of flow/the physics describing gas pressure. The following sequences of commands illustrate the same idea.

```
AddOpReadRTU(Op,HMI,RTU)
  If r in a(Op,HMI)
  and r in (HMI,RTU)
    Enter r in a(Op,RTU)
End

AddOpReadPipe(Op,RTU,Pipe1)
  If r in a(Op,RTU)
  and r in a(RTU,Pipe1)
    Enter r in a(Op,Pipe1)
End

InferStateOfPipe(Op,RTU,Pipe1,Pipe2)
  If r in a(Op,Pipe1)
  and r in a(Op,RTU)
  and r in a(RTU,Pipe1)
  and c in a(Pipe1,Pipe2)
    Enter r in a(Op,Pipe2)
End
```

The encoding of the natural gas transport system in to the Take Grant model follows directly from the HRU encoding. Consider an operator in the system. This operator has read, write and take rights over the HMI which he is operating. Each HMI has read, write and take rights over all RTUs connected to it. Finally each RTU has read and write rights over the pipe on which it is placed. To encoding the ability of the operator to infer information about the state of pipes in the system not directly in his control, the object SDK is used. It is in this object that the notion of flow conservation, which was applied in the HRU model commands to allow operator inference, is placed.

Figure 2 shows the T-G graph with the additional SDK object. Through the SDK object the operator has the ability to take the take right over all pipes in his possession and then to take read rights over their neighbors. Thus by using his domain knowledge, the operator can infer the state of neighboring pipes.

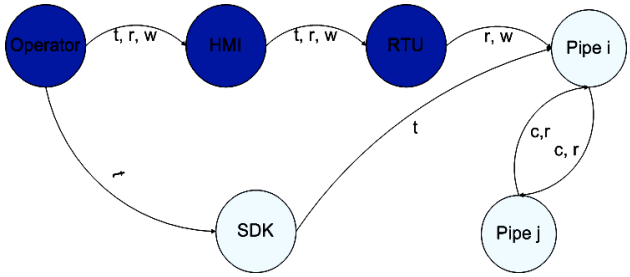


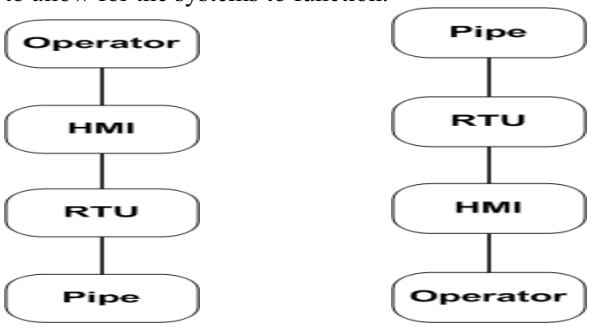
Figure 2. T-G graph for Gas transport system.

Similar use of SDK in every infrastructure caused the unintentional leakage of rights; but modeling these infrastructures without the notion of SDK was practically impossible to enforce security. Also, we observed that it is very difficult to model real systems with the take-grant model because all possible passages of rights between subjects must be known in advance. These factors made it difficult for us to build secure infrastructure with respect to these models.

A. Bell-LaPadula

The Bell-LaPadula model was difficult to adapt to a Cyber-Physical System. In the gas transport system, for example, to allow two way communication between operators and HMIs as well as between HMIs and RTUs, the encoding of the natural gas transport system contains only one security level. The security categories and discretionary read and write access are used to encode the fact that operators can only interact with the HMI in their control, HMIs can only interact with the RTUs connected to them, and that RTUs can only read and write to the pipe on which they sit. Figure 3**a** depicts these security levels. Formally every entity in the system has security level Medium. Operators all have security category {HMI, RTU, PIPE}. All HMIs have security category {RTU, PIPE}. All RTUs have security category {PIPE} and all pipes have security category {}. We encapsulate the notions of operators using specific HMIs, HMIs connected to certain RTUs and RTUs sitting on specific pipes by the discretionary read and write access provided in the BLP model. Using the BLP model it is difficult to encode the notion of an operator inferring the state of pipes which are neighboring his own.

Similar results were obtained with the other infrastructures we considered. BLP was too restrictive and to allow for the systems to function.



a. Security levels for BLP **b**. Integrity levels for Biba

Figure 3. Classification for Gas transport system.

B. Biba

The gas transport system is encoded into the Biba model as follows. The lower level objects, such as pipes and RTUs, have a higher integrity level than the higher level objects, such as operators and HMIs. The integrity level classification is as shown in Figure 3**b**. More trust is assigned to lower level objects because it is less likely that they contain false information. For example the system pipes simply follow the physics of the system; that is they do not lie. The RTUs get their information directly from the pipe on which they are sitting. It is less likely that a RTU will contain false information than it is that an HMI or operator will, whose information is being disseminated from multiple different devices spread out over a large distance.

Using Biba's Strict Integrity Policy, operators are able to read from any object of integrity level no less than their own. However with his current integrity level the operator is not able to write to any objects in the system. Because of this fact alone the Biba model is not appropriate for this infrastructure.

Similar results were obtained with the rest of our infrastructures. However, using the variants of the Biba model called the low-watermark policy and the ring policy which provide a temporary dynamism to the (subject) integrity levels, we were able to convincingly emulate this model in case of most of the infrastructures. We can do so at a compromise due to untrusted subjects, which is many times intolerable.

C. Non-Interference

The concept of non-interference [12], universally, could not be shown for the Cyber-Physical infrastructures considered. Almost all the infrastructures we considered resulted in non-secure implementations with respect to non-interference. For example, looking back in to our Gas transport system, the actions of one operator can interfere with the observations of another operator. The condition of the gas in all pipes in the system is considered to be the state of the system. The outputs of actions taken by an operator are the changes those actions induce in the system pipes. An operator can only observe changes to the state of the gas in pipes under his control. However due to the physics of the system changes in system pipes are not confined to those pipes as the effect propagates outwards, in this manner the actions of one operator can interfere with the observations of another.

Formally we let our set of commands contain the two commands: raise and lower pressure, which respectively, raise and lower the pressure of the gas within the pipes they are executed on. Each operator has the ability to observe the state of the gas only within the subset of the system pipes under his/her control. Consider the topology presented in Figure 4. Any of the two commands that operator A executes on his pipe will necessarily have an affect on its neighboring pipes and as such will constitute an observable event by both operators B and C. An invariant on this relationship due to the physics of the system is that pressure a = pressure b + pressure c where a, b and c represent the pressure of the gas in the pipes controlled by operators A, B and C respectively.

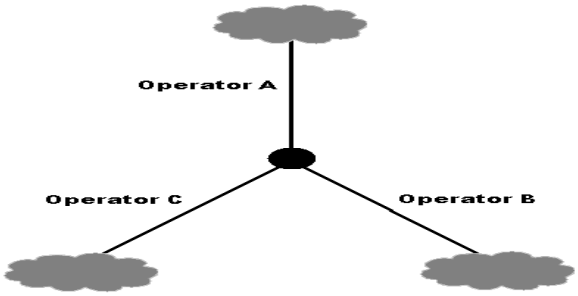


Figure 4. Topology of the Gas transport system.

Consider the system trace Γ where operator A lowers pipe a's pressure to half that of its current value. Operator B will observe a change in pipe b's pressure as dictated by the physics of the system. Thus, non-interference does not hold as

[Projection (Γ) = b's pressure drops] ≠ [Proj (Γ removing A's action) = b's pressure does not change]

A similar situation with other infrastructures occurred in which any change in the system made by a high level subject interfered with the other subjects (which happen to be at low-level) in the system. For this reason non-interference does hold for the considered Cyber-Physical infrastructures.

D. Non-Inference

Many infrastructures satisfied the criteria to be non-inference secure. For example, the Chinese wall policy can be modeled using the non-inference model. Let us consider the information flow across a company dataset (CD) as shown in Figure 5.

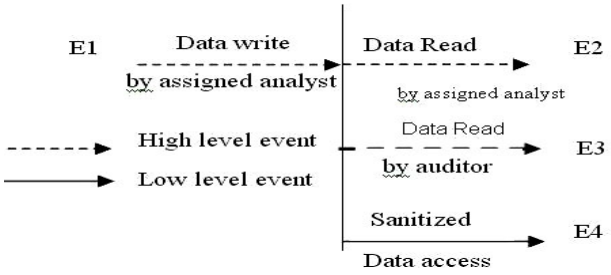


Figure 5. Data flow across a CD in CW-policy.

The CD is formed from the traces of the form {{}, E1, E2, E3, E4, E1E3, E1E2, E2E3, E1E4, E1E3E4, ...}. The traces could be obtained from a closure of such sequence of events. Purging all high level events, we would be left with E4 or {} E4 is a legal trace since it is a low level event. So low level events cannot infer any information from high level users meaning that our system is non-inference secure. Other infrastructures which are non-inference secure are the Automated traffic control system (with additional traffic lights), Telesurgery, Disaster responders, Steel foundry and the Inter-organization information flow.

The Non-inference model did not work for our Gas transport System. We can see this by considering the ability of an operator to infer commands executed by another operator by observing the state of the gas in certain system pipes. We use the same system topology of the Gas Transport system as in Figure 4, as well as the same set of commands. Again consider the system trace Γ where operator A lowers pipe a's pressure to half that of its current value. As stated above, operator B will observe a change in pipe b's pressure as dictated by the physics of the system. This could be due to increased consumer demand or by A's actions (both of which are high-level events). That is the gas's pressure in a pipe will not change unless acted upon. Thus, we have Γ restricted to low level events that consist only of output events. This is not a valid trace (pressure cannot change without any inputs) from which it follows that the system is not non-inference secure. Other infrastructures where non-inference did not work for similar reasons are the Air traffic control system, Oil rig, Smart house, Engine control system, Bear tracking and the Farm intelligence system.

E. Non-Deducibility

We now show that our Gas Transport system is non-deducibility secure. Informally this is easy to see. We can never know for sure if the change in the state of the gas in our pipe is a result of a command executed by another operator or caused by a change in consumer demand or by some combination of both (all high level inputs). As a result we can never rule out any particular sequence of high level inputs, no matter what the observed low level effects are.

Formally let us assume that we observe a change in flow of Δp in pipe b. Let Γ_{HI} be any valid high level input trace of our infrastructure that is a sequence of commands executed by the other operators in the system. To show that there is a Γ Є Tr that is consistent with both the high level input trace Γ_{HI} as well as the observed change of Δp in pipe b we must show that there is some value of the consumer demand which given the high level inputs in Γ_{HI} will produce a observed change of Δp in pipe b. Clearly the same reasoning used in the non-inference models satisfies this requirement. Therefore the natural gas infrastructure is non-deducibility secure. Other infrastructures which are non-deducibility secure are Telesurgery, Air traffic control system, Oil rig, Disaster responders, Bear tracking and the Farm intelligence system.

On the other hand, we had models that failed to be non-deducibility secure. One of them is the Chinese wall policy. Consider the information flow across a company data as shown in Figure 5. We could have the traces of the form of the form {{}, E1, E2, E3, E4, E1E3, E1E2, E2E3, E1E4, E1E3E4, ...}. However, the system does not satisfy non-deducibility requirements, because there is only one high level input E1. If a particular low level subject is not allowed access to a CD, it means that some other subject has already written to that CD before. So basing on the low level activity, we can find out from the system trace that a high-level user has written to the CD which makes the system non-secure with respect to non-deducibility. Other infrastructures where non-deducibility did not work are the Smart house, Steel foundry and the Inter-organization information flow.

The applicability of each model to each infrastructure is summarized in Table 1.

V. RESULTS

The course many times went into "discovery" mode, exploring new ways of modeling. As such, not every problem had a nice, clear, solution. The research component uncovered a number of issues where each model was successful and/or unsuccessful as well as

	HRU	TG	BLP	Biba	NI	NF	ND
Transportation Networks	A¶	A¶	NAΨ	NA§	NA	NA	A
Chinese wall Security	A¶	A¶	A	NA§	NA	A	NA
Automated Vehicle Traffic Control	A¶	A¶	NAΨ	NA§	NA	A	A
Engine control system	A¶	A¶	NAΨ	NA§	NA	NA	A
Inter-organization information flow	A¶	A¶	NAΨ	NA§	NA	A	NA
Disaster Responders	A¶	A¶	NAΨ	NA§	NA	A	A

NI:Non-Interference **NF**:Non-Inference **ND**: Non-Deducibility
A: Applicable **NA**: Not Applicable
¶ Applicable but Compromised by Domain Knowledge
Ψ Too restrictive to implement
§ Applicable with dynamic subject integrity levels

Table 1. Results of Applying Security Models to the Infrastructures

reflected by the groupings indicated in Section IV. From a research perspective, the course was a success as it has spawned several M.S. theses in Computer Science exploring the area of CPS security.

Feedback was measured by anonymous student surveys collected near the end of the course. Empirically, on the plus side, some students liked the real-world infrastructures, the iterative nature of the project, and the discovery process in applying the theory to the infrastructures. On the negative side, some students wanted more textbook-type examples to understand the material instead of exploring the Cyber-Physical issues and felt that too many models were covered. There were students who didn't really understand the project. There was negative feedback on the extensive amount of classroom participation and discussion required. From these responses, the results were mixed. Future study on the next offering could try to match student responses with learning styles or course expectations.

We gratefully acknowledge the participants of CS 483, Nilav Adhikari, Adam Nichols, Ravi Akella, Xuan Gong, Chris Harris, Ekaterina Holdener, Charles Huber, Jason Madden, Vaibhav Reddy, Adam Schneider, Travis Service, Yi Wang, and Josh Wilkerson for their contributions to this paper.

VI. References

[1] Lee, E., "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, Austin, TX.

[2] Tang, H., McMillin, B., "Security of Information Flow in the Electric Power Grid," *Critical Infrastructure Protection*, Springer Boston, pp. 43-56, 2007.

[3] Tang, H., McMillin, B., "Security Property Violation in CPS through Timing," In *Proceedings of the 1st Workshop on Cyber-Physical Systems*, Beijing, June 2008 (to appear).

[4] Bishop, M. *Computer Security: Art and Science*, Addison-Wesley, Boston, 2003.

[5] McLean, J. "Security models," in *Encyclopedia of Software Engineering*, J. Marciniak (Ed.), John Wiley, New York, pp. 1136-1144, 1994.

[6] Schweitzer, D., Collins, M., Baird, L., "A Visual Approach to Teaching Formal Access Models in Security," In *Proceedings of the 11th Colloquium for Information Systems Security Education*, Boston University, Boston, MA, pp. 69-75, June 2007.

[7] Brewer, D.F.C., Nash, M.J., "The Chinese Wall security policy," In *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 206-214, May 1989.

[8] Shladover, et. al., "Automated vehicle control developments in the PATH program," In *IEEE Transactions on Vehicular Technology*, 40, 1, pp. 114-130, Feb.1991.

[9] Chen,Y., Chu,W., "Database Security Protection via Inference Detection," *Intelligence and Security Informatics*, Springer Berlin / Heidelberg, pp. 452-458, 2006.

[10] Health Insurance Portability and Accountability Act (HIPAA) of 1996 found at http://aspe.hhs.gov/admnsimp/pl104191.htm visited in Dec. 2007.

[11] Harrison, M., Ruzzo, W., Ullman, J., "Protection in operating systems," *Communications of the ACM*, 19, 8, pp. 461-471, August 1976.

[12] Gougen, J.A., Meseguer, J., "Security Policies and Security Models," In *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 11-20, April 1982.