An Experiment Platform for Security Education in Advanced Manufacturing Systems: Infrastructure and Materials

Weichao Wang
weichaowang@uncc.edu

Wesley Williams
wbwillia@uncc.edu

Aidan Browne
aidanbrowne@uncc.edu

University of North Carolina at Charlotte
9201 Univ. City Blvd, Charlotte, NC 28223

Abstract - With the fast development of Cyber-Physical Systems (CPS), security in these special application environments starts to attract more and more efforts. In this project, we form a team of researchers in information security, manufacturing, mechanical engineering, and education to jointly design a remotely accessible experiment platform for security education in advanced manufacturing systems (AMS). Students can reserve hardware equipment remotely and control it through a dynamically created virtual machine. Multiple types of hardware equipment, including PLC/MyRIO controlled elevator simulator, PLC/MyRIO controlled golf ball gripper, and a robotic arm, have been connected to the platform. Students can remotely operate on these machines and see their movements through high definition cameras. Course modules and experiments for security education in AMS have been designed upon the platform. These materials have been adopted by one undergraduate level mechanical engineering course. More than 130 hours of hands-on exercises have been conducted by students on this platform.

Categories and Subject Descriptors

K.3.1 [Computers and Education]: *Computer Uses in Education – Computer-assisted instruction (CAI)*

General Terms

Security, Experimentation

Keywords

Security Education, Advanced Manufacturing, Infrastructure, Education Materials

1. INTRODUCTION

With the proliferation of Cyber-Physical Systems (CPS) and Mission Critical Operations (MCO), both higher education institutions and industrial corporations start to emphasize the connection between computer networks and real shop-floor environments. The ever-increasing penetration of Internet of Things (IoT) introduces new security threats to these application scenarios. Equipping the engineering students with basic knowledge and skills in information security and privacy becomes an essential task for the institutions.

As an important direction of support for multiple funding agencies such as National Science Foundation (NSF), advanced manufacturing has attracted a large amount of research efforts from both the mechanical engineering and computer science professionals. Before the integration of these two fields, the control of manufacturing equipment is often conducted by Industrial Control Systems (ICS). Such systems often run in an isolated environment, which makes cyber-attacks from outside sources difficult to conduct. When the ICS is connected to the Internet, malicious parties finally get a chance to launch cyber-attacks remotely upon manufacturing sites. The increased threats to advanced manufacturing environments have been backed up by multiple recent incidents. For example, hackers attacked Lubrizol, an Ohio-based chemicals company, through Industrial Control Systems (ICS) with techniques similar to Stuxnet, the worm that paralyzed the Iranian uranium refinement capabilities [1]. According to a report released by Verizon, over 70% of the attacks examined in 2012 on manufacturing systems were of low or moderate sophistication, which could have been prevented more effectively if the workforce were properly trained in information and network security [2].

What makes the situation even more challenging is the fact that corresponding education programs fall behind in many aspects. For example, to the best of our knowledge, there is no integrated experiment environment of advanced manufacturing systems dedicated to security educational programs that covers all essential components including real time sensing, control networks, and communication networks. UC Irvine's Desktop Integrated Manufacturing Platforms (DIMPs) [3] focus on 3D printing functionality. Since most Computer Integrated Manufacturing (CIM) systems are closed source platforms, they are not designed for in-and-out-of-class exercises since their software infrastructures and programming interfaces cannot be easily grasped by students or instructors. The Open-source Computer Aided Manufacturing (OpenCAM) project [4] can convert vectorial drawing files into machine code for computer numerical control (CNC) machines. However, the project has not been updated since 2006, which makes it out-of-date for modern AMS. The lack of such platforms puts a serious challenge for the training of qualified workforce to fill tens of thousands of positions in the traditional yet fast evolving manufacturing industry [5], which becomes an urgent demand for the recovery of economy [6].

To bridge this gap, we plan to design and implement an experiment platform that includes both the manufacturing equipment and control/communication networks. Since students may need to remotely reserve and operate on the equipment, we propose to design the overall architecture as a cloud. We will then design multiple course modules and hands-on exercises that focus on the security education of the control, communication, and sensing modules of the platform. This integrated environment will provide transformative learning experiences to students in both computer science and mechanical engineering majors since the tight conjoining of, and coordination between cyber and physical systems are embedded throughout the proposed materials. The architecture of our project is illustrated in Figure 1.

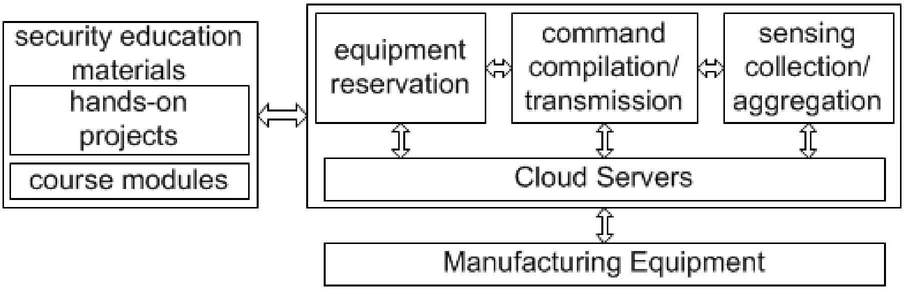

security education materials
hands-on projects
course modules
equipment reservation
command compilation/ transmission
sensing collection/ aggregation
Cloud Servers
Manufacturing Equipment

Figure 1. The logic architecture of the project.

While the architecture of the project is clear, we need to invest efforts in multiple aspects to turn it into a real system. For example, we need to carefully choose the cloud platform so that students can remotely connect to the manufacturing equipment. At the same time, video streams from high definition cameras installed around the machines need to be fed back to end users. As another example, one machine could be controlled through two different methods (e.g. PLC (Programmable Logic Controller) and MyRIO). Mutual exclusion between the two schemes must be enforced. Last but not least, since the platform will be used for security education in advanced manufacturing, we need to provide a channel to students through which they can observe or conduct malicious attacks and test their mitigation mechanisms.

To solve these problems, we have designed and built a prototype system. We use only open source software to build our system in order to reduce future adoption difficulty by other institutions. We have connected multiple types of hardware devices including PLC, Cartesian, and a robotic arm to the platform. Students can remotely reserve and operate on any of these devices through a dynamically created virtual machine. We have designed and implemented multiple course modules and hands-on exercises upon the platform and introduced them to students in the undergraduate course “ETME 4163: Instrumentation and Control”.

The contributions of the paper can be summarized as follows. First, we design and implement a platform for instruction and hands-on exercises for security education in advanced manufacturing systems. Since the platform is built upon open source software and it supports remote access, the difficulty level for adoption is low. At the same time, it is relatively easy for other instructors to design new modules for the system. Second, we have designed multiple course modules and hands-on experiments upon the platform. The course modules cover the basics of infrastructure and data security in advanced manufacturing systems. The modules and experiments have been adopted by one undergraduate mechanical engineering course. Last but not least, more than 130 hours of user exercise data on our platform has been collected for future analysis.

The remainder of the paper is organized as follows. In Section 2, we introduce the architecture of our experiment platform. We will also introduce a few hardware machines that we have built and connected to the platform. In Section 3, we will describe the design of the course modules and experiments using the proposed approach. Detailed description of the materials will be provided. In Section 4 we will describe the planned evaluation efforts. Finally, Section 5 concludes the paper.

2. THE EXPERIMENTAL PLATFORM

2.1 The Platform Architecture

When we choose components for the proposed platform, we have several criteria that we need to consider. First and most importantly, we want the architecture to be built upon open source software packages. The major reasons that we want this property are two folds. First, open source software is usually free or can be acquired with a very low cost. In this way, it is cost-effective for future deployment of our approaches. Second, open source software will allow us to make changes to the packages so that security education materials can be embedded into the software. The second criterion that we consider is the width of adoption of the software packages. If there are many current users of the software packages, future dissemination and adoption of our platform will experience low difficulties. Third, since many students may need to access the platform remotely, we want to build a cloud-based approach. Finally, yet importantly, since the experiments contain both security attacks and defense mechanisms, we want to achieve isolation among the physical equipment when different students are operating them.

We have investigated and compared several possible choices for the platform. For example, the software NetLab+ [7] has been widely used by universities and community colleges to teach computer networks [8, 9] and intro-level engineering courses [10, 11], and build virtual environments [12]. The software package can support tens or even hundreds of user virtual machines, through which end users can control different hardware devices. However, NetLab+ has two potential problems. First, NetLab+ requires institutions to pay a relatively expensive annual fee for the usage. Second, the software packages of NetLab+ are not open to all end users. Therefore, it will be very difficult for us to extend the packages with new components.

Based on the criteria described above, we finally choose the platform architecture as shown in Figure 2. Here we illustrate a setup that contains three servers. In server one, we use the Virtual Computing Lab (VCL) to manage interfaces among different components. VCL is a cloud-style management system of computation resources and is widely used by many institutions [13]. End users can connect to the VCL Scheduling Application (the web VCL portal) and request access to a desired application environment that usually consists of an operating system and a suite of applications. The Web Service component provides tools to request, release, reserve, manage, and govern all VCL resources. All reservation information, access control policies, and machine and environment inventory records are managed by the MySQL database. We have chosen the VMware vSphere as the software package to manage all the virtual machines on the VM servers.

In our platform, we have two VM servers to host user virtual machines. These VMs are dynamically created based on user requests and reservations of the hardware equipment. Both servers use VMware ESXi to manage the virtual machines. More servers that host user virtual machines can be added into this architecture later, if needed.

In our prototype system, any manufacturing hardware that can be assigned with an IP address can be connected to the user virtual machines. At this moment, we have three types of devices that have been tested with user virtual machines: a PLC/MyRIO controlled elevator system, a PLC/MyRIO controlled golf ball gripper, and a robotic arm. We are currently testing several other types of devices (e.g. laser cutter, 3D printer) by connecting them to a computer. After successful testing, we will connect them into our system.

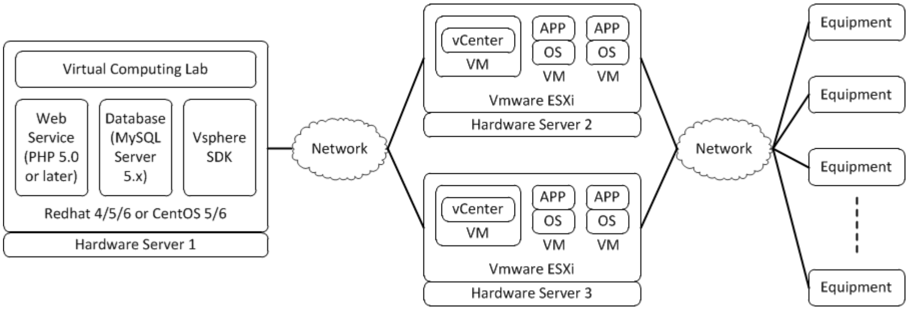
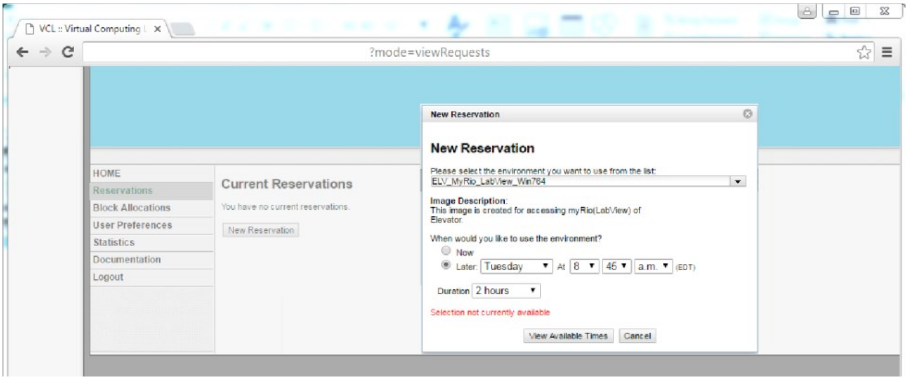


Figure 2. The architecture of our experiment platform

2.2 User Interfaces

In the following section, we will introduce the design of user interfaces. When a user wants to reserve one or several hardware devices for an experiment, she/he can login to the system and check the available slots of the devices. In Figure 3, a student tries to reserve the Elevator for 2 hours starting from the next Tuesday 8:45 am. Our system reminds him that the requested slot is not available. He can click on the "View Available Times" button to see the slots that he can use.



Current Reservations

You have no current reservations.

New Reservation

Please select the environment you want to use from the list:
ELV_MyRio_LabView_Win764

Image Description:
This image is created for accessing myRio(LabView) of Elevator.

When would you like to use the environment?
○ Now
● Later: Tuesday At 8 45 a.m. (EDT)

Duration 2 hours

Selection not currently available

Figure 3. User reservation interface.

When the requested time arrives, a virtual machine will be created for the user. She/He can use her/his credential to login to the VM. Our extended component of VCL will guarantee that the reserved hardware equipment is bound to this VM during this period and will accept commands from this VM only. At the same time, this VM will get the control of the high definition cameras around the equipment and receive the video streams. Figure 4 shows the two video streams from the front and back side cameras of the elevator

During the reserved time slot, three conditions may lead the user to logout from the virtual machine. In the first condition, the user has not finished the project but she/he has to leave for a while. She/He can logout of the account. The virtual machine and the reserved hardware equipment will be kept for the user. She/He can come back and login again later. In the second condition, the user has successfully finished her/his project and no longer needs the VM or the equipment. Under this condition, she/he can go back to the VM reservation interface and release the resources. The physical equipment and the remaining time slot will then become available to other users. In the third condition, when the reserved time slot expires, the student has not yet finished her/his project. About 10 minutes before the ending time, the system will remind the user to store the current progress of the project to remote user spaces. If the time slot immediately after is still available, the user can renew her/his reservation. On the contrary, if some other user has already reserved the next slot, the current user will be forced out once the time expires. Her/His VM will be destroyed. The bundle between the current VM and the hardware equipment will also be removed so that the next user can start to operate on the machine.

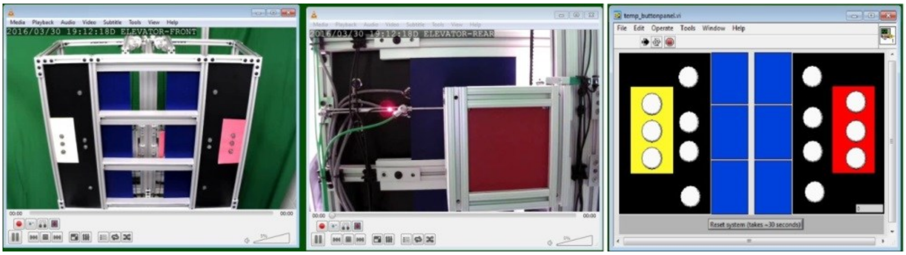

Media Playback Audio Video Subtitle Tools View Help
2016/03/30 19:12:18D ELEVATOR-FRONT
2016/03/30 19:12:18D ELEVATOR-REAR
temp_buttonpanel.vi
File Edit Operate Tools Window Help
Reset system (takes ~30 seconds)

Figure 4. From left to right: front and back side camera view of the elevator, and the virtual control panel of the elevator.

Other than the reservation procedure described above, our system also provides several functionalities to support the special needs of users and instructors. First, some project may need the user to reserve multiple devices as a group. Our system allows the user to identify all equipment that she/he needs for the current reservation. If some device is not available for the specified time slot, our system will issue a warning. The user has to change her/his reservation or the project cannot proceed.

In the second condition, multiple users may need to conduct a group-based project. Under this case, one user will be assigned as the leading user and only her/his VM can control the equipment. At the same time, other group members will also get their VMs created. They will receive the camera streams and see the movements of the machines. However, they cannot control the hardware equipment.

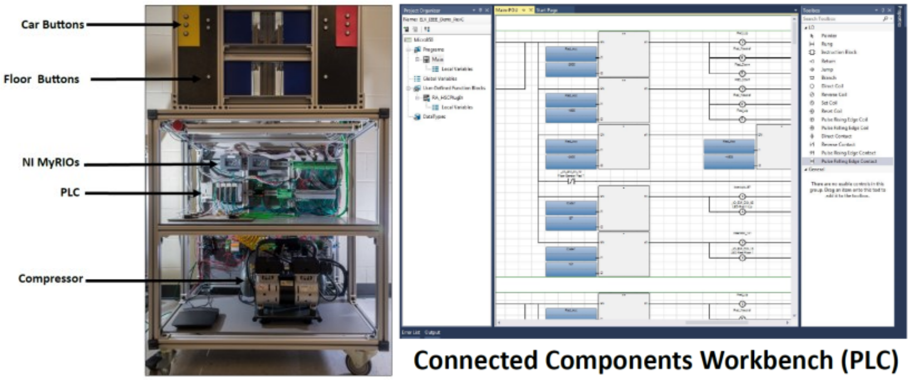
In the last condition, sometimes a student may need help from the instructor. Since the student could be controlling the hardware remotely, it is not always practical for the instructor to go to the student's VM and help her/him. To solve this problem, we always keep an "instructor's VM" active in our system. This VM can take over the control of the hardware equipment from a student's VM. After adjustment or debug, the instructor can give the control back to the student.

2.3 Hardware Equipment

Since in our experiment environment a VM controls hardware equipment through networks, in theory any device with an IP address can be connected to our platform. To support security education in the undergraduate level "Instrumentation and Controls" course, we design and build two remotely controllable devices. Below we describe their designs.

The first machine we build is a small-scale elevator. The elevator can be controlled by either PLC or Reconfigurable I/O (MyRIO) through National Instruments' LabVIEW software. The major components of the prototype include two National Instruments' MyRIO microcontrollers, one Allen-Bradley Micro 850 PLC, and two Pan Tilt Web-cameras. The design is shown in Figure 5.

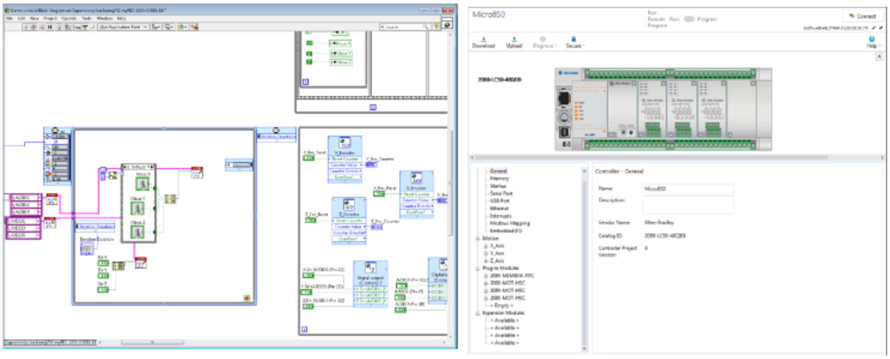
The second device we build is a three-axis Cartesian gantry gripper. Here the gripper can move golf balls in three-dimensional spaces. By including a tilt function on the table, we can remotely reset the setup so that all golf balls will align along one edge of the table. In this way, the device will become ready for the next reservation without instructor involvement. The design is shown in Figure 6.



Elevator Cart (Front View)

Connected Components Workbench (PLC) Programming of the Elevator

Figure 5: Design of the elevator.



MyRIO Programming of the Cartesian Setup

PLC Programming of the Cartesian

Figure 6: Design of the golf ball gripper.

3. EDUCATIONAL MATERIALS AND EXPERIMENTS

3.1 Course Modules

We have designed several course modules on security of advanced manufacturing systems. Since they emphasize different aspects of this topic, they can be adopted by different courses. Below we introduce several modules in detail.

3.1.1 Overview of Advanced Manufacturing System and Its Cyber Security

The goal of this module is to provide an introduction to the overall architecture of the advanced manufacturing systems including both the cyber and physical sub-systems. For the physical sub-system, we will introduce the complete lifecycle of manufacturing and the functionality of each component including design and specification, production method selection and scheduling, process control, measuring and monitoring, and quality control. For the cyber sub-system, we will cover various communication networks in advanced manufacturing systems, industrial control systems, sensing and monitoring, and adaption to system dynamics. Advantages and disadvantages of two types of control mechanisms, programmable logic controller (PLC) and embedded controller, will be discussed. Mutual impacts between the cyber and physical sub-systems will be emphasized in the materials.

In the cyber security part, we will first answer the question of what makes security enforcement in AMS different from that in a computer system. We will provide high-level discussion of the problems such as the attack surface in both sub-systems and threats from both inside and outside attackers. Since many traditional security measures such as authentication, authorization, and accountability demonstrate unique properties in advanced manufacturing systems, we will reintroduce these concepts based on the new application environments. As a special emphasis, we will demonstrate how cyber-attacks can lead to delay in product release, ruined equipment, leakage of intellectual property, and even risk to human safety. State-of-the-art security measures in AMS and their impacts on system performance will also be introduced.

3.1.2 Infrastructure Security and Reliability in Advanced Manufacturing System

This module provides an in-depth coverage of the control flow and information network infrastructure in AMS and its security. We will first describe the roles and functionality of the components in the infrastructure including computers, controllers, manufacturing equipment, and sensors/actuators. We will then introduce different types of computer networks and industrial control systems that connect the hardware together. Many network attacks in Internet have their companions in advanced manufacturing systems. Using a PLC system as an example, we plan to introduce multiple representative attacks. The examples will include: (1) attacks on infrastructure and equipment availability through control command compilation, resource reservation, and task scheduling; (2) Denial of Service attacks towards computer networks or industrial control protocols; and (3) malware for industrial sensors and actuators. Since flexible manufacturing [14, 15] has become a widely-adopted technique in AMS, we will demonstrate how cyber-attacks on the reservation and task scheduling algorithms can impact the infrastructure reliability of AMS.

3.1.3 Data Security in Advanced Manufacturing Systems

This module tries to provide an in-depth coverage of the threats to integrity and confidentiality of data in AMS. We will first describe the closed-loop information flows including command compilation and transmission, sensor data collection and aggregation, and feedback and adaptation. Since different network protocols may be used at different stages of data processing, we will discuss the vulnerabilities during data format transformation procedures. We will introduce several concrete examples of command and data manipulation attacks [16, 17, 18] such as command injection and replay attacks as well as the impacts of contaminated sensing data on the control procedures. Since data transmitted in AMS without sanitization may lead to disclosure of properties of product under construction and sensitive information of end users, we will also introduce the countermeasures to preserve data privacy.

3.2 Hands-on Experiments

We have designed a group of projects upon the proposed experiment platform. These experiments focus on the mutual impacts between the cyber and physical systems. Below we describe a few examples in detail.

3.2.1 Experiment 1: DoS attacks on equipment availability through resource reservation

Although many flexible manufacturing approaches [19] have implemented tool and machine sharing, the granularity and flexibility of sharing in AMS cannot be compared to those in computer systems. For example, in computer systems you can interrupt and recover a process almost instantly through context switch without much performance penalty. In manufacturing systems, however, it is usually very costly to interrupt a job. This special property enables new types of DoS attacks upon equipment reservation and task scheduling. For example, a manufacturing task needs to use a specific type of machining equipment for at least two hours. If there is no control on equipment reservation, an attacker can schedule many jobs with an interval of 1.5 hours. Therefore, although there are many time slots in which the equipment is unoccupied, we cannot fit the task in.

To defend against such attacks, students need to design their own priority based reservation management algorithm and implement it in the experiment platform. Since this is an open-ended question, there does not exist a single correct answer. After the algorithm is implemented, a sequence of manufacturing tasks will be submitted to the system. The overall duration to finish these jobs and the average waiting time of each job will be used to evaluate the algorithms. Through this project, we expect the students to understand infrastructure level DoS attacks on AMS and the decisions that they have to make for tradeoff between system security and performance.

3.2.2 Experiment 2: Attacks on command integrity in PLC and their defense

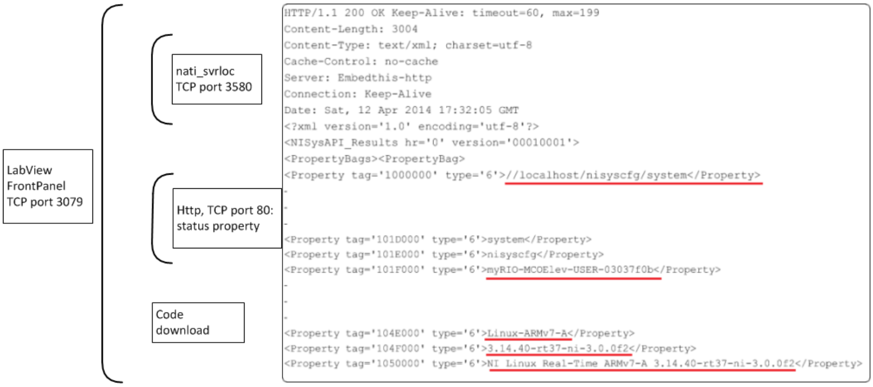
In this project, students are first required to program the Cartesian gripper to pick up golf balls in different colors and arrange them into a specific pattern. Since the commands do not have any confidentiality or integrity protection, attackers can remove or insert a command to change the patterns.

The defense mechanisms will be implemented in two phases. In phase one, students need to implement a simple encryption algorithm to protect the confidentiality of the commands. Here we assume that the attacker does not know the secret key. Therefore, the attacker can no longer insert a random command into the system to affect the final color patterns.

To demonstrate that protection to information integrity is way beyond encryption, in the first half of phase two of the project, we require students to use replay attacks to impact patterns of the golf balls. Here an attacker does not need to compromise the encryption key since she/he can reorder/resend the commands to affect the final result. In the second half of phase two, students are required to implement two more features into their defense mechanisms: command freshness and order. Both sequence number based and challenge-response based mechanisms will be required so that students can properly select solutions in their future jobs based on application scenarios.

3.2.3 Experiment 3: Manipulation of Device Status Properties

In this experiment, students will first set up network eavesdropping software such as WireShark on the virtual machine. The communication history between LabVIEW at the VM and the MyRIO device will be captured and analyzed. Finally, the students will use an IP packet manipulation tool to generate fake packets that report false values of device status properties to the LabVIEW software.


LabView FrontPanel TCP port 3079
nati_svrloc TCP port 3580
Http, TCP port 80: status property
Code download

```
HTTP/1.1 200 OK Keep-Alive: timeout=60, max=199
Content-Length: 3004
Content-Type: text/xml; charset=utf-8
Cache-Control: no-cache
Server: Embedthis-http
Connection: Keep-Alive
Date: Sat, 12 Apr 2014 17:32:05 GMT
<?xml version='1.0' encoding='utf-8'?>
<NISysAPI_Results hr='0' version='00010001'>
<PropertyBags><PropertyBag>
<Property tag='1000000' type='6'>//localhost/nisyscfg/system</Property>
-
-
-
<Property tag='101D000' type='6'>system</Property>
<Property tag='101E000' type='6'>nisyscfg</Property>
<Property tag='101F000' type='6'>myRIO-MCOElev-USER-03037f0b</Property>
-
-
-
<Property tag='104E000' type='6'>Linux-ARMv7-A</Property>
<Property tag='104F000' type='6'>3.14.40-rt37-ni-3.0.0f2</Property>
<Property tag='1050000' type='6'>NI Linux Real-Time ARMv7-A 3.14.40-rt37-ni-3.0.0f2</Property>
```

Figure 7. Communication history between LabVIEW and MyRIO. The underlined contents contain sensitive information of the equipment.

Figure 7 shows some of the experiment results. On the left side, we illustrate the communication history between LabVIEW and the MyRIO controller. We can see that LabVIEW uses HTTP to get the status properties of the device. On the right side we show a part of the transmitted data. All status properties are transmitted in plaintext. We can see that some information is related to system security, such as the position of the configuration file and the version of the OS kernel. Its disclosure could lead to subsequent attacks.

Students can change the configuration at the MyRIO side and acquire the status property file again. Comparing the results of the two files, they can learn to decode the meanings of some property tags. As another exercise, the students are asked to generate a fake HTTP packet in which the MAC address of the MyRIO device in the property file is changed to the broadcast address 0xffff:ffff:ffff. They will then observe the changes at the LabVIEW side.

4. EFFORTS FOR ADOPTION AND EVALUATION

4.1 Material Adoption

The proposed materials are currently being adopted in the undergraduate level course "ETME 4163: Instrumentation and Controls" at UNC Charlotte. The course module on security of AMS is delivered to the students. This effort serves as a pilot study of the project on Security Education in Advanced Manufacturing Systems. All students in the class choose to participate in this study after the debriefing of the objectives of the project. The study was approved by the Institutional Review Board at our university.

4.2 Evaluation: Survey Instruments

The students who participate in this study will be asked to take two surveys. First, the "Self-Efficacy to Use AMS and Its Security Knowledge" survey is developed by the project director and the program evaluator to measure how well students feel they can reach the course objectives. Theories and practices in self-efficacy were adopted while designing the items [20, 21]. The survey consists of 16 items on a 5-point Likert scale ranging from 1 (cannot do at all) to 5 (certainly can do). This survey will be administered to the students before and after the instruction on the topics.

Self-Regulation for AMS Security is also developed by the project director and the program evaluator to measure student self-regulated learning behaviors while learning the topics in the course. It consists of 13 items on a 5-point Likert scale ranging from 1 (not at all) to 5 (all the time). This survey will be administered to the students before the instruction. It will not be used as a repeated measure because we do not expect these behaviors to change in a short period of time. It will be used as a control variable in data analyses.

5. CONCLUSION

As an important component in Cyber-Physical Systems, advanced manufacturing starts to attract more and more attention from both researchers and educators. The lack of security education materials for advanced manufacturing systems (AMS) restricts the training of qualified workforce. In this project, we design multiple course modules to introduce infrastructure and data security in this field. We have also designed an experiment platform upon which students can conduct hands-on exercises. We describe several experiments of cyber-attacks on

AMS and their impacts on physical systems. We implement the materials in one undergraduate level course.

Immediate extensions to our project can be conducted in the following ways. First, we will conduct formal evaluation of the educational materials and their effectiveness in learning. The evaluation results will provide guidelines for future improvement. We will also implement these materials in more courses so that more students can benefit from the project achievements. Second, we will refine the interface design of our experiment platform and make it public. In this way, other educators can start to use it and design new materials upon it. Last but not least, we will present our materials to our industrial partners and collect feedback from them. In this way, we can train qualified workforce for them.

ACKNOWLEDGEMENT

This research is supported in part by NSF SaTC Award #1523096. The product reflects the opinions of the authors but not NSF.

REFERENCES

[1] Jennifer Bisceglie and Michael McGrath, "Cybersecurity for advanced manufacturing," technical report, White Paper prepared by National Defense Industrial Association Manufacturing Division and Cyber Division, 2014.

[2] Verizon, "Data breach investigations report," http://www.verizonenterprise.com/DBIR/, 2014.

[3] Gobind S. Bisht, Giulia Canton, Alireza Mirsepassi, Lawrence Kulinsky, Seajin Oh, Derek Dunn-Rankin, and Marc J. Madou, "Controlled continuous patterning of polymeric nanofibers on three-dimensional substrates using low-voltage near-field electrospinning," Nano Letters, 11(4):1831–1837, 2011.

[4] Celso Junior, "Opensource computer aided manufacturing," http://opencam.sourceforge.net/, 2006.

[5] Joe Biden, "Support for community colleges and apprenticeship programs," presentation at the American Association of Community Colleges 94th Annual Convention, 2014.

[6] Bruce Katz and Jennifer Bradley, "How can networks help modernize a manufacturing economy?" The Metropolitan Revolution book. Melcher Media, 2013.

[7] A. Maiti, "Netlab: An online laboratory management system," in IEEE Education Engineering (EDUCON), pages 1351–1358, April 2010.

[8] R. I. Dinita, G. Wilson, A. Winckles, M. Cirstea, and A. Jones, "A cloud-based virtual computing laboratory for teaching computer networks," in International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), pages 1314–1318, 2012.

[9] S. Sendra, E. Granell, I. Climent, and J. Lloret, "A remote control system to configure network devices," in International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, pages 169–174, 2012.

[10] Jan Machotka, Zorica Nedic, and Ozdemir Gol, "Collaborative learning in the remote laboratory netlab," International Journal of Systemics, Cybernetics, and Informatics, 6(3):22–27, 2008.

[11] Zorica Nedic and Jan F. Machotka, "Remote laboratory netlab for effective teaching of 1st year engineering students," International Journal of Online Engineering (iJOE), 3(3), 2007.

[12] Andrew Smith and Nicky Moss, "Cisco networking: using skype and netlab+ for distance practical learning," in IADIS e-Learning conference, 2008.

[13] S. B. Melhem, T. Daradkeh, A. Agarwal and N. Goel, "Virtual computing lab (VCL) open cloud deployment," in International Conference on Computing, Communication & Automation (ICCCA), 2015, pp. 600-605.

[14] Carsten Eckel and J. Peter Neary, "Multi-product firms and flexible manufacturing in the global economy," Review of Economic Studies, 77(1):188–217, 2010.

[15] Hoda A. ElMaraghy, "Flexible and reconfigurable manufacturing systems paradigms," International Journal of Flexible Manufacturing Systems, 17(4):261–276, 2005.

[16] Saurabh Amin, Alvaro A. Cardenas, and S. Shankar Sastry, "Safe and secure networked control systems under denial-of-service attacks," in Rupak Majumdar and Paulo Tabuada, editors, Hybrid Systems: Computation and Control, volume 5469 of Lecture Notes in Computer Science, pages 31–45. Springer Berlin Heidelberg, 2009.

[17] Alvaro A. Cardenas, Saurabh Amin, Zong-Syun Lin, Yu-Lun Huang, Chi-Yen Huang, and Shankar Sastry, "Attacks against process control systems: Risk assessment, detection, and response," in Proceedings of the ACM Symposium on Information, Computer and Communications Security, pages 355–366, 2011.

[18] Alvaro A. Cardenas, Saurabh Amin, and Shankar Sastry, "Research challenges for the security of control systems," in Proceedings of the Conference on Hot Topics in Security, pages 1–6, 2008.

[19] James R. Moyne and D. M. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data," in Proceedings of the IEEE, 95(1):29–47, Jan 2007.

[20] F. Pajares, "Motivational role of self-efficacy beliefs in self-regulated learning," in D.H. Schunk & B. J. Zimmerman (Eds.), Motivation and self-regulated learning: Theory, research, and applications. New York, NY: Routledge, 2009.

[21] C. Wang, D-H. Kim, R. Bai, and J. Hu, "Psychometric properties of a self-efficacy scale for English language learners in China," System, 44, 24-33, 2014.