Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

DIGITAL TWIN MODELING, CO-SIMULATION AND CYBER USE-CASE INCLUSION METHODOLOGY FOR IOT SYSTEMS

Saurabh Mittal

The MITRE Corporation
1051 Channingway Drive
Fairborn, OH 45324, USA

Andreas Tolk
Andrew Pyles

The MITRE Corporation
903 Enterprise Parkway, Suite 200
Hampton, VA 23666, USA

Nicolas Van Balen

The MITRE Corporation
1 Hospital Drive
Charlottesville, VA 22908, USA

Kevin Bergollo

The MITRE Corporation
202 Burlington Road
Bedford, MA 01730, USA

ABSTRACT

Cyber Physical Systems (CPS) and Internet of Things (IoT) communities are often asked to test devices regarding their effects on underlying infrastructure. Usually, only one or two devices are given to the testers, but hundreds or thousands are needed to really test IoT effects. This proposition makes IoT Test & Evaluation (T&E) cost and management prohibitive. One possible approach is to develop a digital twin of the IoT device and employ many replicas of the twin in a simulation environment comprised of various simulators that mimic the IoT device's operational environment. Cyber attack experimentation is a critical aspect of IoT T&E and without such a virtual T&E environment, it is almost impossible to study large scale effects. This paper will present a digital twin engineering methodology as applicable to IoT device T&E and cyber experimentation.

1 INTRODUCTION

Cyber Physical Systems (CPS) and Internet of Things (IoT) communities are often asked to test devices regarding their effects on underlying infrastructure. Usually, only one or two devices are given to the testers, but hundreds or thousands are needed to really test IoT effects. IoT effects are defined as the effects resulting from the operation of a collection of IoT devices on the critical infrastructure. For example, take the Power Grid (Soltan et al. 2018), and the impact of smart thermostats acting in synchronization to draw power from the Power Grid during the peak hours. Another example would be a smart washing machine using the Water infrastructure. A collection of smart washing machines may load the water infrastructure if they act synchronously. While smart devices are pushing the society to smart houses, they have limited access and control to mitigate the synchronous activity happening at the neighborhood or geographical block levels, when a synchronized execution is resulting from a cyber attack involving more than one domain (e.g. water and electricity domains together). Currently there exists no legal or commercial entities that are tasked to perform such cross domain analyses that have societal implications.

This proposition makes IoT Test & Evaluation (T&E) cost and management prohibitive. One possible approach is to develop a digital twin of the IoT device and employ many replicas of the twin in a simulation environment comprised of various simulators that mimic the IoT device's operational environment. A digital twin is a digital replica of a living or non-living physical entity that co-evolves with the entity.

978-1-7281-3283-9/19/$31.00 ©2019 IEEE

Many academic and research groups have studied the impact of such smart devices on a single infrastructure using modeling and simulation (M&S). The process usually starts with developing a model of the smart device at an abstract level and interfacing the model with a critical infrastructure simulator to study the impact of such a device, in collective, in a simulated environment. While the methodology has become well established by (Mittal et al. 2015) and (Pratt et al. 2017), the process of developing the model, or a digital twin, is not straightforward. Due to the nature of the modeling activity, the abstraction introduces gaps between the real and modeled activities. To close the gap, hardware-in-the-loop (HWIL) is used where the model is replaced by the actual hardware device, which facilitates model calibration and validation. Once validated, the model then is scaled up in the simulated environment.

This paper reports a similar methodology – IoT Digital Twin Engineering– and advances the state of the art by incorporating the modeling of a cyber attack on a collection of digital twins for smart devices, e.g. smart thermostat, and examines the effect of such an attack in a simulated environment. Cyber attack experimentation is a critical aspect of IoT T&E and without such a virtual T&E environment, it is almost impossible to study large scale effects (Mittal and Zeigler 2017).

The research presented in this paper grew out of the discussions at earlier expert panels and tracks conducted in support of better understanding the usefulness of simulation methods to cope with challenges of CPS. The first selected event of interest was conducted at the Spring Simulation Multi-Conference 2018. Our research group provided the results of a literature survey on the use of hybrid simulation support for CPS (Tolk et al. 2018), followed by a panel discussion to provide insights into current research conducted in this domain (Tolk et al. 2018). The need for a common formalism to represent simulation methods as a computational capability for CPS was a recurring discussion point in this panel. Although Discrete Event Systems (DEVS) formalism was not the only solution envisioned, the use for co-simulation based on this formalism was featured by several panelists. Co-simulation is the co-existence of multiple simulators to interoperate in a common scenario. Additional requirements have been identified within (Shao and Kibira 2018), addressing among others the need to allow the evaluation of CPS solution in the context of the portfolio or enterprise in which it is used. Finally, the recent edition of Zeigler's seminary text (Zeigler et al. 2018) focuses explicitly on the support of iterative system computational foundations, including the design of digital twins and the application of hybrid approaches for the development of simulation systems.

However, another main driver for the research is coming from the modeling efforts within the systems engineering discipline: the Digital Engineering transformation (Zimmerman et al. 2017). It is not only the defense community being interested in these efforts, but NASA is addressing similar issues (Puchek et al. 2017). The definition used in the defense community is: *"Digital Engineering is an integrated digital approach that uses authoritative sources of systems' data and models as a continuum across disciplines to support lifecycle activities from concept through disposal."* As such, digital engineering is more an evolutionary than a revolutionary approach.

The use of M&S within acquisition projects has a decades long history, but was often conducted as a parallel activity. Nonetheless, the 1995 US Department of Defense M&S Master Plan identified the support technology assessment, system upgrade, prototype and full-scale development, and force structuring as facets of one of the two pillars of the M&S vision (USDoD 1995). The introduction of Simulation Based Acquisition by the Director for Test Systems Engineering and Evaluation (DTSE&E) during the same period promoted the use of M&S throughout the phases of system design, test and evaluation, and modification and upgrade (Davis 2000). However, the success was not as expected, as the technologies were not aligned on multiple conceptual levels.

With the introduction of Model-based Systems Engineering (MBSE), the systems engineering discipline started a consolidation process of the artifacts used to describe the various facets of systems over the various phases of its life cycle (Wymore 1993). This process took its time, but overall resulted in the convergence of methods and concepts, which also provided a foundation for M&S methods to be integrated. In particular, the new challenges of complexity and emergence did lead to the recognition of the necessity for increasing use of M&S methods to discover, understand, and manage the new phenomena (Sheard et al. 2015).

Our research contributes to this vision: to utilize systems engineering artifacts and enrich them to make them executable digital twins that can represent the systems in a valid representation of their operational environment, allowing for the dynamic evaluation of system behavior in the context of its portfolio. In our proof of feasibility, we furthermore wanted to be able to ensure the representational equivalence of a real system and its digital representation, so that we could conduct experiments on the small scale with the real device, such as a cellphone or a thermostat, and experiments on the big scale, where hundreds or thousands of such systems are needed to evaluate the effect on the grid within a smart city, could be conducted with the digital twins. The following sections will describe these concrete examples and present some tentative results.

The paper is organized as follows. Section 2 elaborates the elements of Digital Twin Engineering and describes the evolutionary process. Section 3 describes the use case in detail. Section 4 describes the technical process and implementation architecture for the M&S framework. Section 5 builds on the use case and modifies it to reflect a cyber attack. It also presents preliminary results and a comparative evaluation between the two use cases. Section 6 concludes the paper.

2 DIGITAL TWIN M&S ENGINEERING

There are three elements to any M&S-oriented digital twin development:

The Modeling aspect: This aspect digitizes the physical concept of the system for which a digital representation is desired. This relates to the well researched topic of MBSE with a new name digital twin. Many times the development and exploration stops at this stage and does not graduate to the next aspect (below). The outcome of this aspect are static digital twin artifacts that help build consensus between different stakeholders, AND if not taken to the next step, become stale and serve as paper-weights and shelfware. The primary reason is the lack of foresight and ignorance of the two aspects below, coupled with the technological hurdle of taking a model to a simulation environment. One has to select an appropriate modeling paradigm and tool that lends itself to simulation.

The Simulation aspect: This aspects takes the digital twin, a model (for simplicity), and implements the dynamic behavior of the model in a computational environment to show the models interaction with other entities (or models) and the environment it resides in. The computational environment may have different flavors such as Live, Virtual, and Constructive (LVC). Live environment deals with interfacing models with real systems and real humans. Virtual environments deal with interfacing models with simulated and emulated systems involving real humans. And lastly, Constructive environments interface models with simulated systems and simulated humans (or agents). The simulation aspect takes the digital twin to a mature level of understanding of its operation within a given operational context and the associated dynamics. This is the validation step. More scenarios and operational contexts (e.g. cyber) can be constructed at this stage that capitalize on the models structure, behavior, and its situatedness within the modeled environment. The digital twin development can stop here and at this stage the validated digital twin may belong to a repository with reuse opportunities as an asset/component, to be reused in future contexts and scenarios.

The Experimentation aspect: This aspect builds on the simulation aspect and actually, is the true motivation behind the whole concept of digital twin. Experimentation involves taking the model, the simulation, and performing parametrization with the digital twin. This aspect helps evolve the model and creates experimental frames that can themselves be part of a repository for large scale experimentation, test, and evaluation in varied contexts.

Together, these three aspects realize the vision of Digital Twin Engineering. These aspects were elaborated in the context of M&S-based complex systems engineering (Zeigler et al. 2018).

Coming to the IoT context, this research hits on these three points for an IoT device. Fundamentally, the problem with test, evaluation, experimentation, verification, and validation of IoT devices is the lack of availability of the contextual virtual environment (as in LVC above). With reference to the IoT context:

- *The Modeling aspect*: This employs a modeling paradigm such as discrete event, continuous time, and agent-based or a hybrid approach to model the IoT device structure and behavior in an operational context. The resulting hybrid model is not a one-off solution. Using a formal modeling paradigm such as DEVS facilitates automated inclusion in the next step.
- *The Simulation aspect*: This entails taking the hybrid model and standing up a simulation environment to execute the model's dynamic behavior. The hybrid simulation environment is called Co-Simulation environment, and as the name suggests, it is the co-existence of simulators and emulators that need to be brought together for simulation of a hybrid model. At this point, the model is in a position to go through a validation exercise and once validated, is reusable.
- *The Experimentation aspect*: This capitalizes on the simulation aspect and performs large scale experimentation with validated models in different operational contexts.

2.1 Digital Twin Evolutionary Process

A digital twin is an evolving entity as it is exposed to different scenarios over time and gets refined as a result. Figure 1 shows the evolutionary cycle for a typical Digital System (IoT System as an example). The process has two distinct sections: End-user and M&S Services, separated by a dotted line. The *End-User* section refers to the inputs that an end-user provides, with regards to operational scenario, parametrics, and operational constraints. The *M&S Services* section refers to the technical M&S infrastructure that provides the Digital Twin capability for modeling, simulation, and experimentation activities.

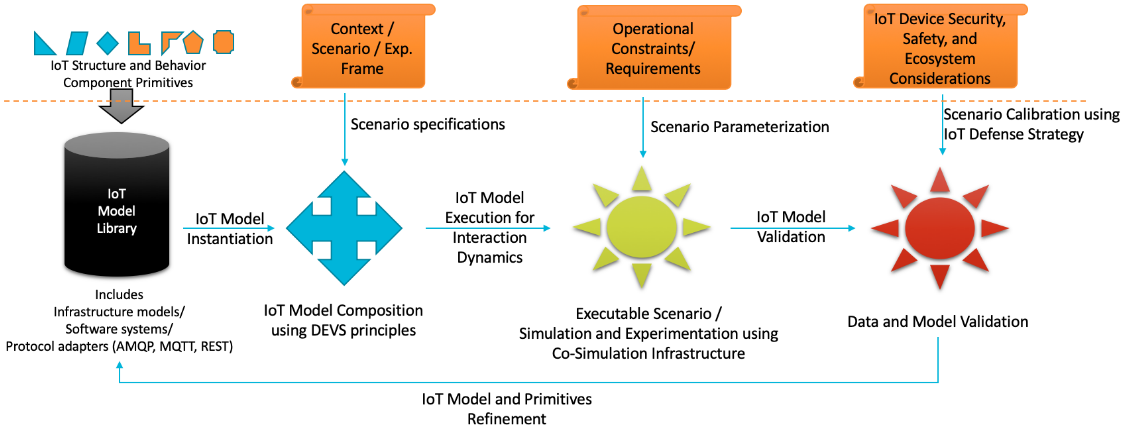

IoT Structure and Behavior
Component Primitives
Context / Scenario / Exp. Frame
Operational Constraints/ Requirements
IoT Device Security, Safety, and Ecosystem Considerations
Scenario specifications
Scenario Parameterization
Scenario Calibration using IoT Defense Strategy
IoT Model Library
IoT Model Instantiation
IoT Model Execution for Interaction Dynamics
IoT Model Validation
Includes Infrastructure models/ Software systems/ Protocol adapters (AMQP, MQTT, REST)
IoT Model Composition using DEVS principles
Executable Scenario / Simulation and Experimentation using Co-Simulation Infrastructure
Data and Model Validation
IoT Model and Primitives Refinement

Figure 1: Digital twin evolutionary process.

The process begins with the existence of a Model Library (IoT Model Libary in our case) that holds various IoT-device digital twins in a format that lends them to a composable scenario. The IoT device digital twin follows a configuration management process that allows the usage of various versions of an IoT-device digital twin with other relevant dependencies. The IoT Model Library also consists of virtual infrastructure models, software dependencies, and protocol adapters. The IoT system is a multi-protocol system, and is described in the next section. These components are then composed in an IoT scenario using formalisms such as DEVS, that allow explicit structure and behavior composability. This step develops the notion of scenario specification and provides details of the composed IoT digital twin (in entirety). The next step formulates the notion experiments that the scenario is evaluated on. It defines the parameters that the scenarios are experimented upon and the outcomes each experiment should end up with. Employing DEVS formalism allows the inclusion of simulation as the enabling technology for performing experimentation. If more than one simulator is needed, the accompanying co-simulation infrastructure provides seamless simulation capability. Once the scenario is executable and experiments available, more experiments with non-functional requirements are added in the next step. The last step brings in various other considerations such as IoT Device security, safety and IoT ecosystem considerations that may impact the overall IoT system behavior. These considerations, which are rooted in IoT operational environment, provide relevant data that is then used to calibrate the IoT-device digital twin. This calibration results in creating more versions of the digital twin, which are then stored back in the IoT Model Library.

3 USE CASE DESCRIPTION

To illustrate the methodology, we considered a smart thermostat, such as a NEST thermostat (www.nest.com). A smart thermostat is a device that allows superior access and control of a thermostat by being more embedded in the local environment. The environment may consist of input sources, not just the house temperature, but also occupancy, occupant mobility, time of the day, outside weather, remote operation, and market conditions. While input sources related to occupants are internal to the house, other sources are external and universally applicable. An Internet-enabled thermostat allows remote modification of these parameters using a mobile app, installed on a mobile phone. With remote operation comes the cyber implications as the smart thermostat is now amenable to hacking.

The use case is depicted in Figure 2. It shows the entire IoT System (as considered in this article), comprised of 3 smart homes (containing smart thermostat, smart hubs, mobile apps, and user), interacting with the electricity Power Grid. The IoT-enabled smart thermostat also communicates with the manufacturer (NEST) network and a user-accessible mobile app. Through our use-case, we would like to study the following two things:

1. Information and resource flow interaction dynamics between various system entities, and
2. Effect of a cyber attack through the IoT-device Application Programming Interface (API) when the API is compromised by a hacker using a software bot or the mobile app that utilizes the API. The hacker/bot may execute an attack where many smart devices are synchronized in operation.

Figure 3 shows the system entity structure (SES) for the IoT system in a very minimalistic way. The SES notation is a very succinct way of depicting architectures (Mittal and Martin 2013), including IoT systems (Mittal et al. 2019).

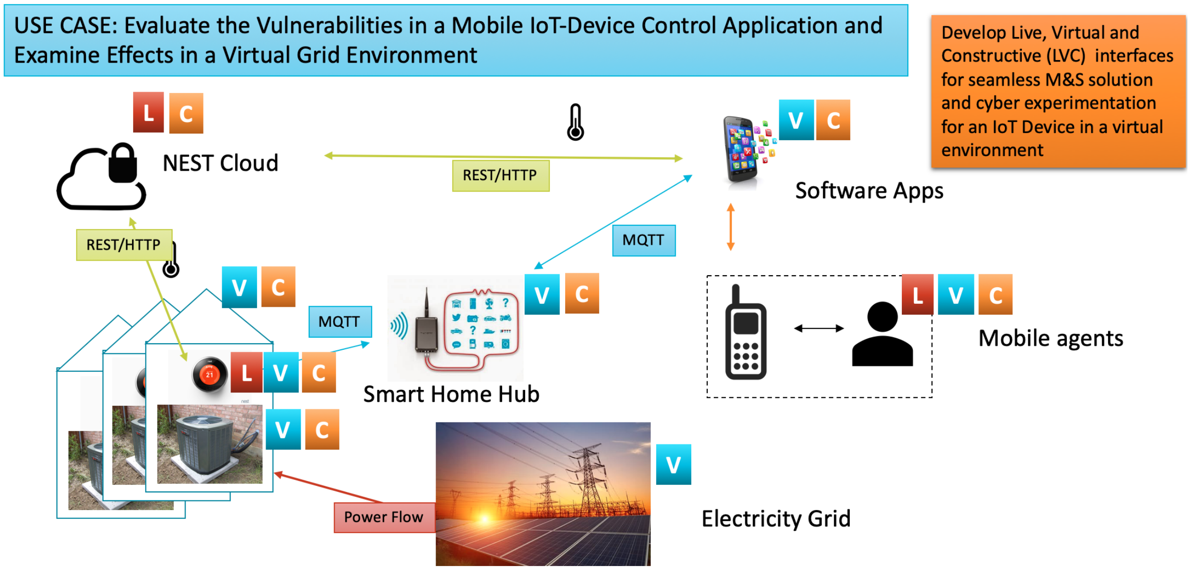

USE CASE: Evaluate the Vulnerabilities in a Mobile IoT-Device Control Application and Examine Effects in a Virtual Grid Environment
Develop Live, Virtual and Constructive (LVC) interfaces for seamless M&S solution and cyber experimentation for an IoT Device in a virtual environment
L
C
NEST Cloud
REST/HTTP
V
C
Software Apps
REST/HTTP
MQTT
V
C
MQTT
V
C
Smart Home Hub
L
V
C
Mobile agents
L
V
C
V
C
Power Flow
V
Electricity Grid

Figure 2: IoT use case and live, virtual and constructive categorization.

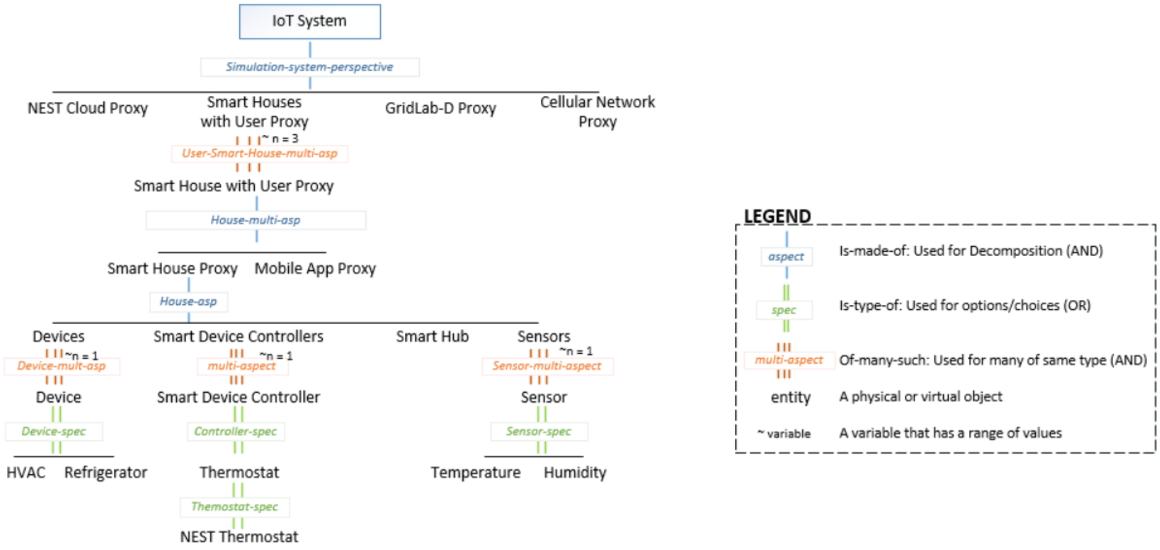


Figure 3: System Entity Structure (SES) for IoT system.

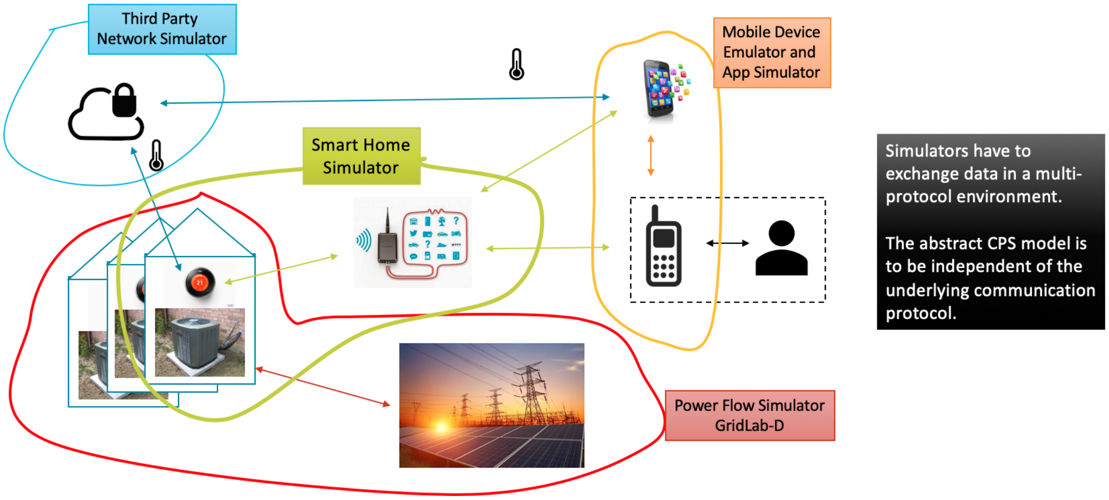
4 TECHNICAL PROCESS AND IMPLEMENTATION ARCHITECTURE

Figure 2 also depicts the IoT System architecture and shows constituent system components labeled as Live (L), Virtual (V), and Constructive (C). A constructive system is an abstract representation, a model, of the real system. A virtual system is a simulated or emulated version of the real system; and a live, as the name suggests, is the real system itself. To perform M&S-based IoT systems engineering, the IoT device's digital twin begins its evolutionary path as an isolated abstract constructive model, followed by its replacement with a higher fidelity digital twin, which many times is an emulator provided by the device manufacturer itself, followed by the actual hardware device. In this progressive path, the digital twin continues to interact with other system components within the architecture that may be in different L, V, C configurations themselves. The M&S methodology conforming with DEVS formalism allows such model continuity (Hu 2005). An IoT system is a multi-protocol system i.e. there does not exist a single communication protocol connecting all the constituent systems. Primarily, we are considering MQTT and REST/HTTP protocols. This requires an appropriate middleware be engineered in the developed M&S test-bed. The whole objective of the test-bed is to provide a virtual environment for the IoT digital twin's design, experimentation, and validation. Since incorporating live elements is both cost-prohibitive and technically challenging, much of the experimentation is carried in Virtual or Constructive mode, with only validation of the digital twin occurring in the Live mode. Once the digital twin is validated, then experimentation at scale can further take place purely in a virtual environment.

Figure 4 shows the architecture from a simulation system perspective. To create an M&S workbench (i.e. editor and execution test-bed), the IoT System architecture was decomposed of various simulation systems as there does not exist a single system that can serve as the test-bed. The simulation systems need to cooperate for creating a virtual, constructive environment for both the modeling and simulation activities. The co-simulation architecture consists of four simulation systems: (1) Smart Grid simulation such as GridLab-D, (2) Smart Home simulation incorporating the IoT-device digital twin and smart hub, (3) Mobile Device app emulator and (4), network communication simulator. The co-simulation architecture also contains a multi-protocol middleware that facilitates message exchange between the constituent simulation systems.

4.1 Co-Simulation Framework, Analytics, and Complex Event Processing

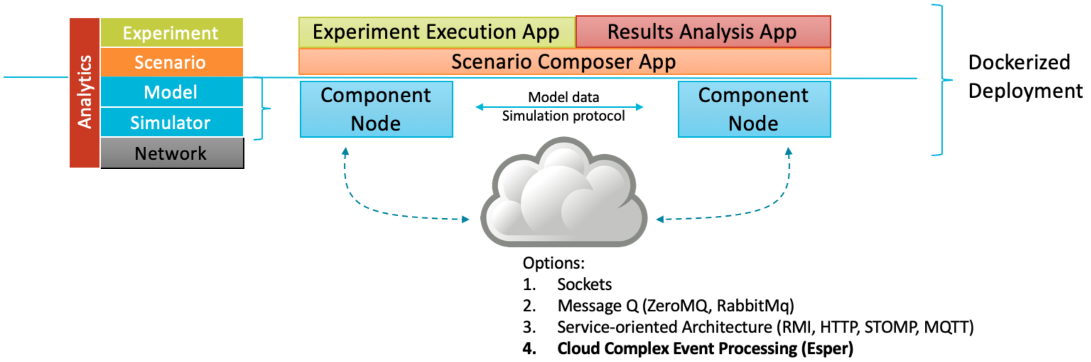
Figure 5 shows the co-simulation architecture. It also depicts the communication middleware that may be implemented using four methods: (1) network sockets, (2) messaging queue, such as RabbitMQ or ZeroMQ, (3) Service-oriented architecture (using protocols such as Remote Method Invocation [RMI], HTTP, STOMP, and MQTT), and (4) Event cloud providing complex event processing (CEP) capabilities. We selected the last option of event cloud for superior runtime pattern discovery capabilities provided by the CEP engine. As we are using DEVS formalism as the core kernel for the co-simulation architecture and the included middleware, various plugins (as DEVS wrappers and proxies) were developed that allowed simulation systems to interact across different media.



Simulators have to exchange data in a multi-protocol environment.

The abstract CPS model is to be independent of the underlying communication protocol.

Figure 4: Need for co-Simulation environment.



Options:

1. Sockets
2. Message Q (ZeroMQ, RabbitMq)
3. Service-oriented Architecture (RMI, HTTP, STOMP, MQTT)
4. **Cloud Complex Event Processing (Esper)**

Figure 5: Co-simulation architecture and complex event processing options.

4.2 IoT Device Live Integration

The NEST thermostat digital twin was validated using a hardware in the loop (HWIL) approach with a NEST thermostat and a Java agent (to serve as an interface between the digital twin and the hardware device). More elaborate description will be reported in an extended article. Below is a brief summary.

Hardware Circuit: A *NEST T4000ES Learning Thermostat E* was used in a hardware circuit that allows for monitoring of the thermostat output. We needed to isolate the NEST thermostat in our test-bed without actually connecting it to the HVAC system. Power was supplied from a 24VAC adapter and two LED lights, one red and one blue, were connected to the thermostat's leads for the heating and cooling systems respectively. When either system sent power, the corresponding LED lit up signalling the system was active in a given mode. The thermostat was connected to wireless network and communicating with the NEST Cloud as is the case of normal operation. This was a minimalist configuration that allowed us to communicate with the NEST device through the established procedure that required NEST Cloud be in the loop. For a more detailed integration configuration refer (Pratt et al. 2017).

Java Agent: A Java agent or a software bot was created using a custom library to interface with both the digital twin and the NEST Cloud. Although any communication protocol would be possible, for this particular case, the agent communicated with the digital twin using MQTT protocol over RabbitMQ as a message broker. When an action was requested over the MQTT interface, the agent used an authentication code, PIN, and a device ID obtained from the NEST developer's API to send the appropriate command to the NEST Cloud to task the device. Currently, these commands include getting the list of devices associated with the account, and the thermostat commands, which include getting and setting high and low target temperature setpoints, and the operating mode (heat or cool) for the HVAC.

5 FORMULATING CYBER USE CASE AND EXPERIMENTATION

5.1 Cyber Considerations

The overall IoT system architecture consists of one or more smart IoT devices installed in a residential neighborhood environment in multiple smart homes. We focused specifically on the Nest T4000ES Learning Thermostat E, an IoT-enabled thermostat, which can be remotely accessed and can control the home HVAC unit. If the communication to the thermostat is intercepted, a remote attacker could send arbitrary commands to the HVAC unit at will.

The Cyber use-case technical approach consists of two steps. First, externally observe and detect network traffic and protocols used. Second, observe normal usage patterns and determine which patterns are of interest. To externally observe and detect network traffic, the first step is to power on the device and externally observe network device behavior. This includes monitoring network traffic and ideally observing and decoding protocols. In order to observe normal usage patterns and determine patterns of interest, we investigated the NEST Cloud API, and found the necessary sequence of API calls to: (1). Set MAX temperature, (2). Set MIN temperature, (3). Determine location (zipcode), and (4). Change HVAC mode.

After these were implemented, we observed the Java Client/bot could set these values for the NEST device. Further, we observed that a single account using the API could set the MIN temperature, thus powering the HVAC unit ON without raising suspicions. A more detailed NEST device integration is described in our prior work at NREL (Pratt et al. 2017). The focus of this research was to ensure that the digital twin was a valid digital twin and can externally interact with other systems displaying correct behavior.

5.2 Cyber Use Case

After manual analysis of a single NEST residential device, we made the following observations:

- The Java bot could arbitrarily power ON/OFF a single HVAC unit via the NEST Cloud.
- The HVAC unit powering ON produced a localized, minuscule power spike.
- Java bot required unique account credentials.

While a single HVAC unit produced an insignificant power spike, the cyber use case we envision is a localized coordinated attack on the Power Grid. We made the following assumptions:

- A given zip code has a large population of NEST users.
- Most users use the NEST mobile application.

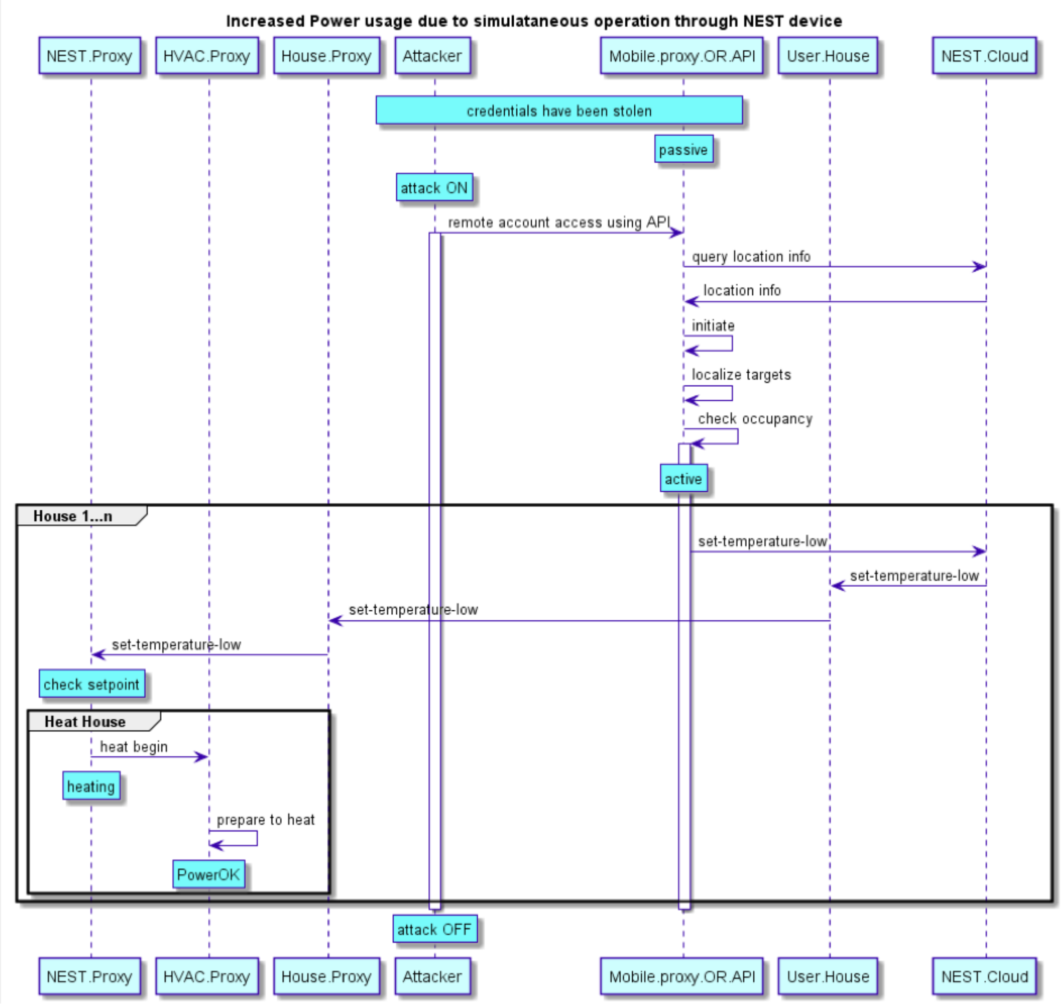


Figure 6: Cyber attack use case.

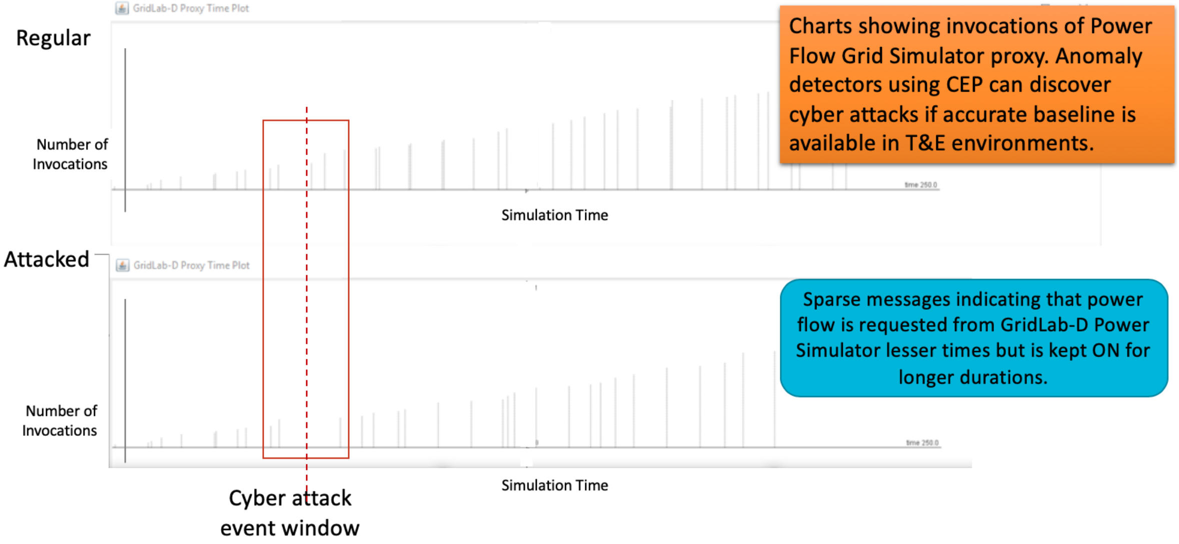
- Attacker has access to a list of mobile phones of users within a specific area code. This might be obtained from Home Owner Association (HOA) directories, neighborhood swimming pools, etc.
- Attacker has access to an exploit of a mobile phone and uses the exploit to steal NEST credentials from a mobile phone, for example, see (Google. 2019).
- Attacker can quickly deploy Java Agent bot instances associated with each stolen credential.

With the stolen credentials in hand, the attacker can either (a) leverage the NEST API and develop a custom application or (b) use the official NEST Android App. If the latter, the attacker can use an Android emulator (as shown in Figure 6, labeled as mobile.proxy). This has the added benefit of using the preferred method and may set off fewer flags. Both methods have the potential to scale quickly. Following the use case, we made the following sequence diagram: Figure 6. It shows how the attack can be executed by the existing IoT system components by augmenting existing functionalities. The block area is standard operation of any NEST device for enabling heating mode in HVAC. However, once the credentials were compromised at NEST Cloud, the Mobile.proxy or Java agent bot can use the NEST Cloud to exploit the normal operation of NEST devices in a coordinated attack.

5.3 Preliminary Results

The 4-node 3-house IoT system structure shown in Figure 3 is modeled using DEVS formalism as implemented in xDEVS (www.github.com/jlrisco/xdevs). The results of such a co-simulation were reported earlier in (Ruth et al. 2015) and (Mittal et al. 2015), and were reproduced to establish a baseline before cyber experimentation was attempted.

When the cyber attack was enabled by the Mobile.proxy component on the NEST Cloud proxy component on the simulated timeline, we saw the IoT System behavior adapting to the attack. Figure 7 shows the comparison between the executions. It shows: (1) the instant the cyber attack event was injected (dotted line in the red box), the GridLab-D proxy continued to draw the power and did not shut off any of the HVACs, and (2) the frequency of invocation of GridLab-D proxy is reduced with the HVACs switched on for longer durations. More elaborate results will be reported in our extended article.



Charts showing invocations of Power Flow Grid Simulator proxy. Anomaly detectors using CEP can discover cyber attacks if accurate baseline is available in T&E environments.

Sparse messages indicating that power flow is requested from GridLab-D Power Simulator lesser times but is kept ON for longer durations.

Figure 7: Comparison between normal execution and attacked execution.

6 DISCUSSION AND CONCLUSIONS

Digital Twins are a necessity in today's digital world. Digital twin engineering comprises of three elements: modeling, simulation, and experimentation, each requiring specific undertaking. Without all three aspects included in any digital twin endeavor, it will fall short of expectations and will lead to failure.

IoT Device experimentation needs a digital twin and IoT experimentation needs a virtual environment. Virtual environment consists of simulators and emulators. Simulators are specialized virtual environments that implement domain knowledge, e.g. weather, power flow, transportation, vehicle, etc. Co-simulation platform is imperative to such experimentation. Effects of any system's actions cannot be studied until system component interactions are studied in a virtual environment. IoT digital engineering requires multi-domain simulator integration. Cyber experimentation can build on such a platform and inform more robust studies.

DEVS formalism provides a proven tested methodology to incorporate MBSE best practices coupled with M&S infrastructure. Abstract representations of IoT devices can be used to develop models in a progressive manner: Constructive to Virtual (Simulator/Emulator) to Live (Hardware) systems. Validated constructive models can be scaled up in virtual and constructive environments to do large scale studies and impact on the modeled environment.

We illustrated a sample use case employing digital twin of a NEST IoT-enabled smart thermostat and the IoT system it is part of. We described the use case, the technical architecture, and the modeling of a cyber attack by augmenting the use case. We demonstrated the progressive development of a digital twin by first isolating it at all three levels: constructive, virtual, and live, and then employing the virtual and constructive to perform cyber experimentation. We also reported how HWIL can be used to validate the digital twin's behavior.

IoT experimentation at scale and its impact on critical infrastructure can be studied effectively with this methodology. The Internet of Things, where things can be devices, drones, or any online system for that matter, requires a formal methodology to do large scale experimentation and analysis.

Digital Twins and the associated simulation and experimentation environment, saves money, resources, and mitigates risk before system designs are fielded. A forward leaning Digital Twin experimentation environment lends itself to inclusion of new models/systems before they are integrated in the real-world.

DISCLAIMER

Approved for Public Release; Distribution Unlimited. Case Number 19-1646. © 2019 The MITRE Corporation. All Rights Reserved.

REFERENCES

Davis, W. J. 2000. "Simulation-based Acquisition: An Impetus for Change". In *Proceedings of the 2000 Winter Simulation Conference*, edited by J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, 1061–1067. Orlando, FL: Institute of Electrical and Electronics Engineers, Inc.

Google. 2019. "Android Security Bulletin - May 2019". https://source.android.com/security/bulletin/2019-05-01, accessed 13th May 2019.

Hu, X. 2005. *Model-Continuity-based Life Cycle Systems Methodology for Real-time Systems*. Ph.D. thesis, Department of Electrical and Computer Engineering, University of Arizona, Tucson, Arizona.

Mittal, S., S. A. Cane, C. Schmidt, R. B. Harris, and J. Tufarolo. 2019. "Taming Complexity and Risk in Internet of Things (IoT) Ecosystem using System Entity Structure (SES) Modeling". In *Complexity Challenges in Cyber Physical Systems: A Modeling and Simulation Approach*, edited by S. Mittal and A. Tolk. Hoboken, NJ USA: John Wiley & Sons.

Mittal, S., and J. Martin. 2013. *Netcentric System of Systems Engineering with DEVS Unified Process*. Boca Raton, FL USA: CRC Press.

Mittal, S., M. Ruth, A. Pratt, M. Lunacek, D. Krishnamurthy, and W. Jones. 2015. "A System-of-Systems Approach for Integrated Energy Systems Modeling and Simulation". In *Proceedings of Summer Simulation Multi-conference*, edited by D. Cetinkaya, , J. L. R. Martin, I. C. Moon, E. Syriani, F. De Rango, and S. Mittal, 1–10. Society of Modeling and Simulation International.

Mittal, S., and B. P. Zeigler. 2017. "Theory and Practice of M&S in Cyber Environments". In *The Profession of Modeling and Simulation: Discipline, Ethics, Education, Vocation, Societies, and Economics*, edited by A. Tolk and T. Oren, 223–264. Hoboken, NJ: John Wiley & Sons.

Pratt, A., M. Ruth, D. Krishnamurthy, B. Sparn, M. Lunacek, W. Jones, S. Mittal, H. Wu, and J. Marks. 2017. "Hardware-in-the-Loop Simulation of a Distribution System with Air Conditioners under Model Predictive Control". In *IEEE Power and Energy Society General Meeting*. Institute of Electrical and Electronics Engineers, Inc.

Puchek, B., M. Bisconti, P. Zimmerman, J. Guerrero, P. Kobryn, G. Kukkala, C. Baldwin, J. Hale, and M. Mulpuri. 2017. "Digital Model-Based Engineering: Expectations, Prerequisites, and Challenges of Infusion". Technical report, Model-based Systems Engineering (MBSE) Infusion Task Team, Interagency Working Group on Engineering Complex Systems (IAWG), Washington, D.C.

Ruth, M., A. Pratt, M. Lunacek, S. Mittal, H. Wu, and W. Jones. 2015. "Effects of Home Energy Management Systems on Distribution Utilities and Feeders under various market structures". In *23rd International Conference on Electricity Distribution*. Lyon, France: Association des Ingnieurs de Montefiore, Belgium.

Shao, G., and D. Kibira. 2018. "Digital Manufacturing: Requirements and Challenges for Implementing Digital Surrogates". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 1226–1237. Gothenburg, Sweden: Institute of Electrical and Electronics Engineers, Inc.

Sheard, S., S. Cook, E. Honour, D. Hybertson, J. Krupa, J. McEver, D. McKinney, P. Ondrus, A. Ryan, R. Scheurer, J. Singer, J. Sparber, and B. White. 2015. "A Complexity Primer for Systems Engineers". *INCOSE Complex Systems Working Group*

White Paper. https://www.incose.org/docs/default-source/ProductsPublications/a-complexity-primer-for-systems-engineers.pdf, last accessed: 15 May 2019.

Soltan, S., P. Mittal, and H. V. Poor. 2018. "BlackIoT: IoT Botnet of High Wattage Devices Can Disrupt the Power Grid". In *27th USENIX Security Symposium*. Baltimore, MD USA: USENIX.

Tolk, A., F. Barros, A. D'Ambrogio, A. Rajhans, P. J. Mosterman, S. S. Shetty, M. K. Traoré, H. Vangheluwe, and L. Yilmaz. 2018. "Hybrid simulation for cyber physical systems: a panel on where are we going regarding complexity, intelligence, and adaptability of CPS using simulation". In *Proceedings of the Symposium on M&S of Complexity in Intelligent, Adaptive and Autonomous Systems, Spring Simulation Conference*, edited by S. Mittal and J. Martin. San Diego, CA: Society for Modeling and Simulation International.

Tolk, A., E. H. Page, and S. Mittal. 2018. "Hybrid Simulation for Cyber Physical Systems: state of the art and a literature review". In *Proceedings of the Annual Simulation Symposium*, edited by E. Frydenlund, S. Jafer, and H. Kavak, 10:1–10:12. San Diego, CA: Society for Modeling and Simulation International.

USDoD 1995. "Modeling and Simulation (M&S) Master Plan". Technical Report DoD 5000.59-P, Office of the Under Secretary of Defense for Acquisition and Technology, Washington, DC. https://apps.dtic.mil/docs/citations/ADA301787, last accessed 15 May 2019.

Wymore, A. W. 1993. *Model-based Systems Engineering*. Boca Raton, FL USA: CRC Press.

Zeigler, B. P., S. Mittal, and M. Traore. 2018. "MBSE with/out Simulation: The State of the Art and Way Forward". *Systems* 6. https://doi.org/10.3390/systems6040040, last accessed, May 15, 2019.

Zeigler, B. P., A. Muzy, and E. Kofman. 2018. *Theory of Modeling and Simulation: Discrete Event & Iterative System Computational Foundations*. San Diego, CA USA: Academic Press.

Zimmerman, P., T. Gilbert, and F. Salvatore. 2017. "Digital engineering transformation across the Department of Defense". *The Journal of Defense Modeling and Simulation*. https://doi.org/10.1177/1548512917747050, last accessed 15 May 2019.

AUTHOR BIOGRAPHIES

SAURABH MITTAL is Chief Scientist for Simulation, Experimentation, and Gaming Department at The MITRE Corporation, USA. He holds Ph.D. and M.S. in Electrical and Computer Engineering from the University of Arizona, Tucson. His e-mail address is smittal@mitre.org and ORCID is 0000-0003-1430-5799.

ANDREAS TOLK is Technology Integrator for the Modeling, Simulation, Experimentation, and Analytics Division of The MITRE Corporation and Adjunct Professor at Old Dominion University. He holds a PhD and M.Sc. in Computer Science from the University of the Federal Armed Forces in Munich, Germany. He is a senior member of ACM and IEEE and a Fellow of SCS. His e-mail address is atolk@mitre.org and ORC ID is 0000-0002-4201-8757.

ANDREW PYLES is Lead Cybersecurity Engineer at The MITRE Corporation. He received both Ph.D. and M.S. in Computer Science from College of William and Mary. He has two patents related to energy efficiency on mobile devices. His e-mail address is apyles@mitre.org.

NICOLAS VAN BALEN is Mobile Security Software Engineer at The MITRE Corporation. His interests include mobile security, IoT networking, and mobile Android devices. He holds a Ph.D. and M.S. in Computer Science from the College of William and Mary. His e-mail address is nvanbalen@mitre.org.

KEVIN BERGOLLO is Sr. Software Engineer, UX, Visualization, & Decision Support Department at The MITRE Corporation, USA. He holds a Bachelor of Computer Science from the University of Puerto Rico, Mayaguez. His e-mail address is kevinbl@mitre.org.