Developing Simulation Capabilities for Supply Chain Cybersecurity of the Electricity Grid

Joseph M. Keller[1], Shuva Paul[2], Kevin Hutto[1], Santiago Grijalva[1], Vincent J. Mooney III[1]
[1]School of Electrical and Computer Engineering, [2]Energy Systems and Resiliency
[1]*Georgia Institute of Technology*, [2]*National Renewable Energy Laboratory*
[1]Atlanta, Georgia 30345; [2]Golden, Colorado 80401, USA
Email: {jkeller40, khutto30, sgrijalva6, mooney}@gatech.edu[1], shuva.paul@nrel.gov[2]

***Abstract*—The electricity grid has evolved into a cyber-physical system that integrates electric systems with advanced information and operation technology. With the frequency of cyber-incidents increasing in recent years, the need for secure and robust cybersecurity solutions for supply chain networks for critical infrastructures, such as the electricity grid, has intensified. This paper proposes a testbed that involves hardware-in-the-loop simulation of power system operations and real-time data acquisition and management using the *OSIsoft* PI System and a common set of power control devices. A discussion of the development of this simulator's capabilities for supply-chain cybersecurity is included. The simulator is used to emulate the operation of a remote terminal unit, an *SEL* 751 relay, a vendor, a utility, and an attacker to execute the power system adversarial operation in a power grid environment. The testbed can provide researchers with a streamlined method to assess the impact of adding security measures related to the supply chain to the electricity power grid.**

***Index Terms*—Supply chain cybersecurity, real-time data acquisition and management, intelligent electronic devices**

I. Introduction

The U.S. electricity grid is constantly expanding and integrating new resources. With each new resource, a variety of different digital management and control software is required. This opens new supply chain cyber attack vectors. Supply chain cyberattacks are an emerging attack strategy that utilizes a less secure third party to gain access to a target. Some examples of supply chain attacks include SolarWinds [1], the Colonial Pipeline [2], Big Hack [3], and NotPetya [4]. Effective cybersecurity must be integrated into newly installed devices, systems, and infrastructure with a core component of "security by design" [5] to prevent such attacks from occurring.

Georgia Institute of Technology has developed a supply chain cybersecurity framework based on the concept of hardware-oriented cybersecurity to prevent such attacks [6], [7].[1] The proposed cybersecurity system uses a physical unclonable function (PUF) [8] incorporated within a multi-party cryptographic authentication scheme to provide secure communication and data transmission between supply chain entities. To properly evaluate the effectiveness of this security framework, multiple different electricity grid supply chain entities (such as the Vendor, Utility, Third-Party Analysts, etc.) must be simulated. This paper describes the development of simulation capabilities by modeling the power grid, vendor, utility, and operation of substations and control centers. The contributions of this paper are the following:

1) Modeling the simulator functionalities mimicking real-world power system operation.
2) Modeling the functionalities of vendors that request and release updates for different Intelligent Electronic Devices (IEDs) in a substation environment.
3) Development of communication capabilities between different devices to exchange information.

II. Functionality Development

In this section, we discuss the functionality development for the analysis of the *GridTrust* system to enhance the supply chain cybersecurity of the electricity grid. The electricity grid power simulation framework includes several grid devices: a computer that functions as an RTU, a *SEL* 751 relay, a temperature sensor, *OSIsoft* computers, and computers representing the vendor and the utility. Figure 1 displays the architecture of the connections between the components in the simulation. The security requirements for this framework involve secure communication between multiple entities. The architecture can be divided into three layers: a power system layer, a cybersecurity layer, and a control layer. The *power system layer* includes a power system simulator (PSS) developed in Python, the PI System from *OSIsoft*, an interface computer, and a signal generator used to simulate real power values and their real-time data acquisition from a power substation. The PSS reads a model of the power system before calculating the power flows on the lines. The interface computer then receives these values and sends them to other devices. One such device is the signal generator, which will produce an electrical current proportional to the power flow calculated by the PSS. Specifically, the signal generator has a ratio of $1000 : 1$ for the current sent to the relay. The power system layer includes the test devices, an *SEL 751* relay, a temperature sensor, and a computer representing an RTU. The relay receives the current produced by the signal generator, mimicking real-world power operations, before sending its trip response back to the interface computer. The temperature

[1]This work was supported in part by The US Department of Energy Office of Cyber-Security, Energy Security and Emergency Response (CESER), Cybersecurity for Energy Delivery Systems (CEDS) Award to the Georgia Institute of Technology, # DE-CR0000004.

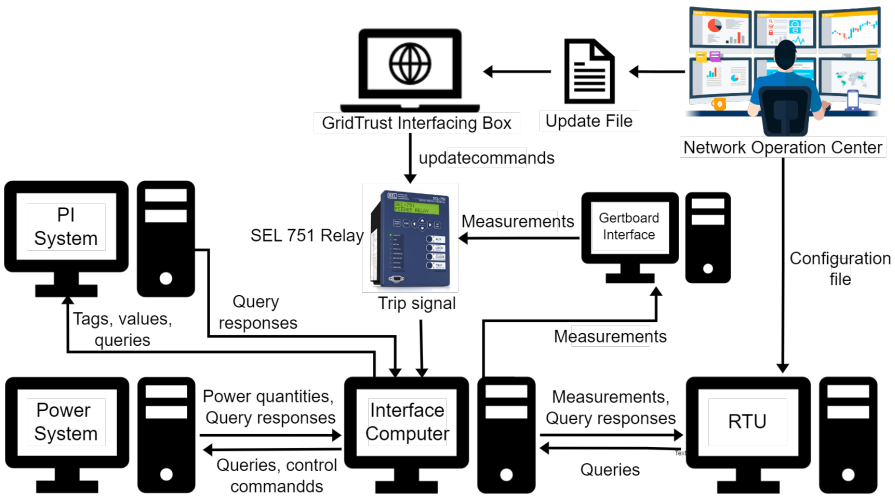

Figure 1: Overall connection and information flow diagram between *OSIsoft* computers, RTU, Relay, and Network Operation Center (NOC).

sensor measures the ambient temperature and reports it to the interface computer. Lastly, the computer that functions as an RTU records and presents specific values created from the PSS. The *cybersecurity* layer involves the components required to handle security in the overall architecture. This includes a centralized network operations server (representative of a utility) that performs key management and acts as a repository for device update and configuration files. The cybersecurity layer also includes the *GridTrust* Interfacing Box connected to various devices. The *control layer* typically deals with the control operations associated with devices and PSS.

A. OSIsoft *Computers and Connections*

The PI System is an *OSIsoft* [9] real-time industry-level data acquisition and management platform. The role of the PI System is to serve as the control system for the power system simulator.

Figure 2: *OSIsoft* Computers in Van Leer 244, ACES Laboratory at Georgia Institute of Technology.

There are three computers dedicated to providing the control system and power system functionalities as shown in Figure 2. The physical power system is simulated in 'VL244-11' and connected to the interface computer/device named 'VL244-12'. The interface computer is responsible for pushing the data/measurements received from the power system to the PI system/control system named 'VL244-10'.

The PSS will generate values representative of realistic power system quantities and send them to the interface computer. It can also query for specific control commands from the PI System via the interface computer. The interface computer sends the queries, values, and associated tags to the PI System. The PI System then stores the tags and their associated values from different assets in the PI server. The PI System also

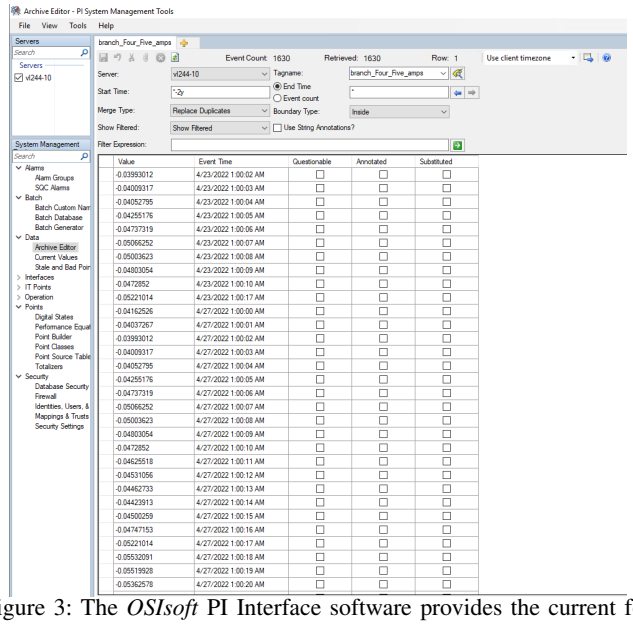

Figure 3: The *OSIsoft* PI Interface software provides the current for branches four to five in the example power system.

sends responses based on the queries made by the interface computer. Figure 3 displays the measured current recorded for an example line in the power system simulator.

The *OSIsoft* interface computer plays a vital role in the management of connections between the PI server, PSS, *SEL* relay, and the RTU. These connections typically send and receive measurements and queries. The connections between the interface computer and other devices are as follows:

- Power System and *OSIsoft* Interface: The power system at VL244-11 generates power system quantities (such as real power flow, currents, voltages, etc.). The simulator at VL244-11 generates an output file containing the timestamps, tag names, and tag values which the *OSIsoft* interface computer at VL244-12 sends to the PI server. The power system also sends queries for specific

commands (such as trip commands) from the PI System via the interface computer.

- *OSIsoft* Interface and PI Server: The output file containing the power system quantities generated and sent by the PSS is pushed to the PI System via a PowerShell script. The interface computer also sends queries from the PI System to the PI Server and receives the responses.
- *OSIsoft* Interface and Relay: The interface computer also connects to the *SEL* relay via an Ethernet to USB cable and through a signal generator. A Python program uses the line current values received from the PSS to scale a current output to a signal generator connected to the relay input. Concurrently, another Python program receives the trip signal from the relay through an Ethernet to USB converter. This trip signal is then sent to the PI System (control center at VL244-10).
- *OSIsoft* Interface and RTU: The *OSIsoft* interface computer connects to the RTU to exchange data (such as measurements, queries, etc.). The RTU queries for the measurement values at different IEDs (from the PSS via the interface computer), performs necessary updates periodically (based on the configuration) from the NOC, and connects back to the interface computer.

B. OSIsoft *Protocols*

The PI System is a real-time data management software system manufactured by *OSIsoft*. It is an integrated portfolio software that collects, stores, views, analyzes, and shares operational data with end users within and outside the network. The data are typically sent from the control devices (such as relays) to the PI System via interfacing devices.

```
Query List Sent to the RTU
[NEW CONNECTION] ('192.168.20.5', 60791) connected.
[RECV] Receiving the filename.
[RECV] Receiving the file data.
[DISCONNECTED] ('192.168.20.5', 60791) disconnected.
branch_two_three_amps_instant.csv
[NEW CONNECTION] ('192.168.20.5', 60792) connected.
[RECV] Receiving the filename.
[RECV] Receiving the file data.
[DISCONNECTED] ('192.168.20.5', 60792) disconnected.
instant_output.csv
[+] Connecting to 192.168.20.8:5016
[+] Connected.
```

Figure 4: Sample operation of the *OSIsoft* interface computer sending and receiving data.

Figure 4 presents a snapshot of the sample operation of the *OSIsoft* PI System interacting with other devices.

The *OSIsoft* PI System supports a wide range of communication protocols for data collection and transfer. Some of the most commonly used protocols include OPC (OLE for Process Control), Modbus, DNP3, TCP/IP, and HTTP/HTTPS. The simulator uses OPC to communicate between the PI System and other elements in the system.

C. Control Layer Execution

Each IED has an internal logic that determines its responses to typical inputs (current, voltage, etc.). The IED responses can be altered through an update to internal logic. We use an update protocol implementing security features that aim to prevent unauthorized software updates [10]. The vendor computer is the original creator and provider of an update. After making an update, the vendor computer sends an update to the utility computer, signed with the vendor's RSA private key. The utility computer then signs the update with the utility's RSA private key. With the update signed by both the vendor and utility, a *GridTrust* device receives an update when required as determined by the utility computer. The *GridTrust* devices have the final authority to decide whether to accept an update or not. An update is only applied to the IED connected to the *GridTrust* device when both signatures are verified by the *GridTrust* software. Through the *GridTrust* verification protocols, *GridTrust* determines if the update is valid or not. If it is not, the process stops there. If the update is valid, the update is transmitted to the associated IED for installation.

D. Modeling Security Layer Functionalities

1) Remote Terminal Unit: The remote terminal unit is simulated on a Linux (Fedora 36 operating system) machine running as a *systemd* service. *Systemd* is an *init* system and service manager for Linux operating systems. The *systemd* service enables the initialization of the RTU functionalities at the boot time of the Linux kernel. By booting the service with *systemd* as manager, the interruption of the RTU functionalities is minimized, improving the quality of service.

```
[rtu@rtu tmp]$ systemctl status sim
● sim.service - "RTU Simulator Code."
     Loaded: loaded (/usr/lib/systemd/system/sim.service; enabled; vendor preset: disabled)
     Active: active (running) since Mon 2022-05-02 10:28:07 EDT; 1h 6min ago
   Main PID: 7502 (python)
      Tasks: 9 (limit: 76783)
     Memory: 34.6M
        CPU: 1.180s
     CGroup: /system.slice/sim.service
             └─ 7502 /usr/bin/python /sim/rtu_error_debug_001.py

May 02 11:32:16 rtu python[7502]: The query list is:  {'IED008': [True, True, True, True], '
May 02 11:32:16 rtu python[7502]: [+] Connecting to 192.168.5.11:5095
May 02 11:32:16 rtu python[7502]: [+] Connected.
May 02 11:32:16 rtu python[7502]: /tmp/rtu_query.json<SEPARATOR>108
May 02 11:32:16 rtu python[7502]: The file was sent to the interface.
May 02 11:32:16 rtu python[7502]: [NEW CONNECTION] ('192.168.5.11', 54153) connected.
May 02 11:32:16 rtu python[7502]: [RECV] Receiving the filename
May 02 11:32:16 rtu python[7502]: This is the filename:  instant_ied_output.csv
May 02 11:32:16 rtu python[7502]: [RECV] Receiving the file data.
May 02 11:32:16 rtu python[7502]: [DISCONNECTED] ('192.168.5.11', 54153) disconnected.
```

Figure 5: RTU operation status as *systemd* service.

Figure 5 shows the RTU operational status as a *systemd* service. The RTU requests specific IED measurements from the *GridTrust* Power System through the interface computer. First, the RTU reads a configuration file to obtain values that it will request from the PSS in the form of a JSON file query. Next, the interface computer forwards this file to the *GridTrust* Power System computer to receive the IED measurements associated with the RTU in response to the query.

2) Vendor: The vendors of the IEDs generate update files for the respective devices. In this work, the vendor is modeled in such a way that it generates the update file and its signature before sending them to the utility for further processing. The vendor has its signature separate from the utility which the IED uses to validate the update.

3) Utility: A utility in the electricity grid supply chain is responsible for producing electricity and delivering the generated electricity to the consumer level. Typically, utility companies own the IEDs and manage the distribution of electricity through a control center. The role of the utility in

this simulation is to receive and direct new updates for the various IEDs. Upon notification of an update, the utility will validate the update before attaching its signature. The utility will then send a command to begin updating the devices.

4) Attacker: This section discusses the capabilities and methods of the attacker.

a) Attacker's Capabilities: An attacker in the practical world is capable of executing a wide range of attacks depending on the skills possessed and the access to the system being targeted. In most cases, the attacker or the advanced persistent threat (APT) groups take months if not years to gain access to the target, stay hidden after compromising the target, and observe/steal valuable information from the target. For the electricity grid as a supply chain network, the attacker may aim to compromise the control center and manipulate/override the control commands to create generation/load loss, cause long-term equipment damage, change generator set-points, etc. The testing scenario involves the attacker being capable of intercepting communications between the Vendor and Utility, as well as between the Utility control center and IEDs under the Utility's purview. This interception gives the attacker the ability to manipulate the configuration and update the files that are transmitted between entities.

b) Attack Function: For the attacker to function, access is required to the cyber-physical system. It is assumed that the attacker has access to the system through the control system or the SCADA network through insider access as an entry-level employee of either the *Vendor* or the *Utility*. With insider access to the network, the attack may intercept and alter files by a man-in-the-middle (MiTM) attack, typically between the Utility control center and a target victim device. As an insider, the attacker has access to a valid update configuration file with authentication signatures for the file from both the Vendor and the Utility. The attacker may then attempt to alter the configuration file and signatures so that the update appears genuine on the victim device. Therefore, the attack attempt validates or invalidates the effectiveness of the security framework in place through the propagation of effects to the electricity grid simulator.

5) Intelligent Electronic Devices: A variety of IEDs are used in this project. An *SEL 751* relay is one of such devices. This relay is a feeder protection relay and its maximum phase overcurrent is the target of test updates. Another IED is a temperature sensor that reads the surrounding temperature in either Celsius or Fahrenheit. For the electricity grid, such devices are utilized to measure and protect devices such as transformers from overheating. Updates to this device include changing the temperature unit on the display. The last example of an IED used is an RTU modeled in a standalone computer. This device collects multiple values from the PSS and displays them. Updates for the RTU include changing the configuration file, which changes the query list sent to the PSS via the interface computer.

III. Developed Use Cases

The effectiveness of the simulator capabilities is demonstrated through three use cases. The use cases are modeled with the aim of performing software updates from the NOC. Representing real-world devices, the use cases include a temperature sensor, a relay, and an RTU.

A. Relay

The initial scenario for the relay involves reducing the maximum phase overcurrent of a line from $0.15\ A$ to $0.13\ A$. Note that these values are scaled down for simulation purposes. The actual currents applied to similar relays in field applications are typically greater than $1\ A$. If an attacker wants to damage the electricity grid by forcing the relay to trip unnecessarily, the attacker can lower the maximum phase overcurrent even further. In this use case, the attack value is chosen to be $0.05\ A$, since that is the lowest the *SEL* 751 relay is capable of setting the threshold to be `(model P/N: 751001A0X0X0XA11DF0)`. Relay update has two components, a *Text file* containing the serial commands to update the relay and a *Rust file* that reads the text file and sends the commands to the relay. Serial commands are chosen to update the computer as it takes less time to complete than an actual software update and is easier to edit. The attacker will attempt to replace the text file before or during transmission with the malicious alternative. Figure 6 summarizes these updates.

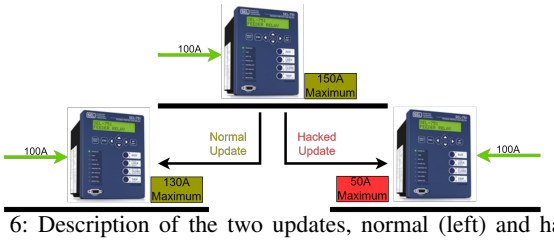

Figure 6: Description of the two updates, normal (left) and hacked (right).

In previous work [11], the authors summarize the experimentation scenarios for this use case. The results of this use case managed to successfully prevent expected trips by using the described method.

B. RTU Update

The remote terminal unit is modeled as a representative unit of a real-world RTU collecting data from multiple sensors and IEDs. The devices of which the RTU records the values are based on a configuration file read by the RTU. Furthermore, the example RTU is also responsible for sending queries for the IEDs to receive, process, and forward the measurement information. The RTU update process in this case study is the retrieval of a new device configuration file (*.config file*). The RTU retrieves a configuration file from the NOC containing the list of IEDs for the RTU to report on. If the configuration/update file is compromised, the requested measurements associated with the IEDs may not be provided or could be provided in an incorrect format hampering the electricity grid operation.

Table I: RTU configuration update with and without protection

TAGS in the Configuration	Pre-Attack Status	Post-Attack Status
IED0001	ON	ON
IED0002	ON	OFF
IED0003	ON	OFF

To emphasize the potential consequences of this use case, we present Table I that presents the statuses of the three specific IEDs, namely IED0001, IED0002, and IED0003. Before the attack, all three IEDs had their statuses set to "ON". However, under adversarial conditions, the configuration file exhibited an unauthorized modification, resulting in a change in the statuses of IED0002 and IED0003 to "OFF", as shown in Table I. This alternation essentially inhibited the RTU from querying the values of these IEDs.

C. Temperature Update

Temperature sensors are ubiquitous in the power system to detect the proper or improper operation of equipment such as generators and transformers. The *Temperature Sensor* is a device we custom-built in our laboratory using a commercial industrial temperature sensor, the LM95172 [12], that measures temperature in either Fahrenheit or Celsius. Updates for this device involve changing the unit that the sensor outputs (i.e., Celsius or Fahrenheit). The sensor initially measures in Celsius. If the sensor configuration requires a Celsius output the sensor reports the temperature as recorded. If the sensor is configured to display Fahrenheit, the sensor converts the temperature from Celsius to Fahrenheit before reporting. When an update occurs, the temperature sensor is reconfigured to convert or not convert the unit and append the correct unit symbol at the end of the report (C or F). A malicious update can involve appending the incorrect unit symbol to the report. This could cause a controller to incorrectly believe that a device is at, for example, $150°$C instead of the actual $150°$F which may cause an automatic trip off of vital equipment.

Table II: Temperature sensor update with and without protection

Protection Status/Attack Timing	Pre-Attack	Post-Attack
Without Protection	150°F	150°C
With Protection	150°F	65.5°C

Table II represents an update changing the temperature units given by the device from Fahrenheit to Celsius. An incorrect update could utilize the Fahrenheit output by appending the incorrect unit symbol.

IV. Discussion and Challenges

With the developed simulator and three physical devices, various tests were conducted that performed malicious software updates on devices in a simulated electricity grid. Through this, the impacts of the proposed security defenses on the electricity grid were quantified [10], [11].

However, some challenges remain in the PSS. In particular, the RTU was simulated using a computer instead of an actual RTU. A similar method to the relay integration may be required to include a physical version. In addition, other real control devices would need different integration and simulation requirements to integrate into the model. Another challenge is to extend the PSS modeling from DC to AC. While DC was useful for simplicity and power estimations, to properly emulate high-power systems, the Python code will need to be updated or replaced for AC. Furthermore, it will be beneficial to simulate more complex attacks within this environment. These attacks include but are not limited to, denial of service, false data injection, and ARP spoofing. These challenges should be addressed in future work.

V. Conclusion

In this paper, simulation techniques and models are proposed to represent cyberattacks on the software supply chain in the context of the electricity grid. The proposed model includes a generic transmission-level power grid operational simulation, models for individual devices involved in a typical electricity grid operation and intercommunication between the devices and a utility control center. With the combination of multiple layers simulated at various levels of abstraction (power system, cybersecurity, and control), the effects of various cyberattacks on critical devices in the electric grid and the transmission system can be simulated. The laboratory setup described in this paper showcased three devices representative of real-world power grid devices. Future work to further develop the simulation capabilities can provide researchers with tools that can accurately provide an estimation of the impacts of including various security features in an electric grid under an adversarial cyberattack.

References

[1] J. Huddleston, P. Ji, S. Bhunia, and J. Cogan, "How vmware exploits contributed to solarwinds supply-chain attack," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 760–765, 2021.

[2] J. W. Goodell and S. Corbet, "Commodity market exposure to energy-firm distress: Evidence from the colonial pipeline ransomware attack," *Finance Research Letters*, vol. 51, p. 103329, 2023.

[3] J. Robertson and M. Riley, "The big hack: How china used a tiny chip to infiltrate u.s. companies," October 4, 2018.

[4] A. Greenberg, "The untold story of notpetya, the most devastating cyberattack in history," *Wired, August*, vol. 22, 2018.

[5] S. Fluchs, R. Drath, and A. Fay, "A security decision base: How to prepare security by design decisions for industrial control systems," in *Proceedings of the 17. EKA (Entwurf Komplexer Automatisierungssysteme)*, 2022.

[6] S. Paul, Y.-C. Chen, S. Grijalva, and V. J. Mooney, "A cryptographic method for defense against mitm cyber attack in the electricity grid supply chain," in *2022 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, 2022.

[7] B. Newberg, S. Grijalva, and V. Mooney, "Open-source architecture for multi-party update verification for data acquisition devices," in *2022 IEEE Power and Energy Conference at Illinois (PECI)*, pp. 1–7, 2022.

[8] R. Maes, *Physically Unclonable Functions Constructions, Properties and Applications*. Springer, 1st ed. 2013.. ed., 2013.

[9] "OSIsoft PI system website," *https://www.osisoft.com/pi-system/*.

[10] K. Hutto, S. Paul, B. Newberg, V. Boyapati, Y. Vunnam, S. Grijalva, and V. Mooney, "Puf-based two-factor authentication protocol for securing the power grid against insider threat," in *2022 IEEE Kansas Power and Energy Conference (KPEC)*, pp. 1–6, 2022.

[11] J. Keller, S. Paul, S. Grijalva, and V. J. Mooney, "Experimental setup for grid control device software updates in supply chain cyber-security," in *2022 North American Power Symposium (NAPS)*, pp. 1–6, 2022.

[12] National Semiconductor, "13-bit to 16-bit 200°c digital temp sensor with 3-wire interface," December 16, 2009.