Security in the Industrial Internet of Things
The C-SEC Approach

Jose Romero-Mariona, Roger Hallman, Megan Kline, John San Miguel, Maxine Major and Lawrence Kerr

US Department of Defense, SPAWAR Systems Center Pacific, San Diego, California, U.S.A.

Keywords: Security as a Service, Machine-to-Machine Communications, SCADA, Industrial Internet of Things.

Abstract: A revolutionary development in machine-to-machine communications, the "Internet of Things" (IoT) is often characterized as an evolution of Supervisory Control and Data Acquisition (SCADA) networks. SCADA networks have been used for machine-to-machine communication and controlling automated processes since before the widespread use of the Internet. The adoption of open internet protocols within these systems has created unforeseen security vulnerabilities. In this paper we detail the Cyber-SCADA Evaluation Capability (C-SEC), a US Department of Defense research effort aimed at securing SCADA networks. We also demonstrate how the C-SEC framework could enhance the security posture of the emerging IoT paradigm.

1 INTRODUCTION

The "Internet of Things" (IoT) is revolutionizing machine-to-machine communications. Since the 1960s, machines have been communicating over diverse and proprietary protocols such as those employed in Industrial Control Systems (ICS) (Ecosteer, 2014). Supervisory Control And Data Acquisition (SCADA) networks are an important subset of ICS used to monitor and control critical infrastructure and automated processes, often over complex networks (Curtis and Wolfe, 2013). SCADA networks are implemented across industries as diverse as electrical, water, oil, and gas utilities. The US Department of Defense (DoD) depends on SCADA networks in unique operational environments where disruption of service cannot be tolerated.

The IoT concept is very broad, with differing visions for what it is, and may be seen as *things-oriented* or *internet-oriented*. A *things-oriented* view of IoT focuses on smart devices, near field communication and RFID, whereas an *internet-oriented* view of IoT is focused on IP for Smart Objects and "IP over everything" (Atzori et al., 2010). The *internet-oriented* view of IoT may then be seen as as a simplification of the current IP protocols that can be adapted to make any object addressable and reachable from anywhere in the world (Atzori et al., 2010). Each approach to security for the IoT presents its own set of concerns.

This paper addresses those concerns with the Cyber-SCADA Evaluation Capability (C-SEC), a novel approach to cyber-physical system security that:

- Provides a DoD-centric perspective of needs to be evaluated.
- Facilitates an independent evaluation, verification, and validation of various IA technologies.
- Defines a process that is standardized, flexible, and scalable.
- Prescribes security metrics that are granular and usable.
- Focuses on SCADA and ICS-specific test cases and evaluation metrics.
- Evaluates technologies of interest in a representative SCADA and ICS laboratory test environment.
- Provides an online repository to enable the comparison of C-SEC results and the effective reuse of previous C-SEC evaluations.

The rest of this paper is organized as follows. Section 2 gives a brief overview of SCADA and IoT security concerns, as well as an introduction to a US Navy research effort out of which the idea for C-SEC was conceived. Section 3 describes C-SEC in considerable detail, describing its evaluation process, metrics, framework, laboratory environment, and online colaborative environment. Section 4 surveys other ef-

Romero-Mariona, J., Hallman, R., Kline, M., Miguel, J., Major, M. and Kerr, L.
Security in the Industrial Internet of Things - The C-SEC Approach.
DOI: 10.5220/0005877904210428
In *Proceedings of the International Conference on Internet of Things and Big Data (IoTBD 2016)*, pages 421-428
ISBN: 978-989-758-183-0

forts in SCADA and IoT security, with a focus on security models and test beds. Finally, concluding remarks are offered.

2 SCADA AND IOT SECURITY

SCADA networks are an important subset of ICS, often used to monitor and control critical infrastructure over complex networks, that has been in use since the 1960s. There is currently an emphasis on Cyber Security for SCADA systems that suggests that many systems had never been connected, but this is mistaken (Ecosteer, 2014). SCADA networks are used for monitoring and controlling machinery over considerable distances on a continual basis (Russell, 2012). Because of their importance to critical infrastructure, system availability and security have been prized above all else (Simões et al., 2015). Therefore proprietary, rather than common protocols (e.g. ASCII) were used (Russell, 2012), though protocol standardization has been a significant effort of the IEEE (Kezunovic, 2002). SCADA networks are also notoriously fragile, with many legacy devices which are unable to support standard security technologies (e.g. anti-malware, network scanners, etc.) (Wilhoit, 2013).

Many of the security concerns for a *things-oriented* view of IoT are not dissimilar to recent security concerns for SCADA networks (Yu et al., 2015):

- Devices often lack timely software updates.
- Device lifespans may continue for many years after vendor support has ceased (e.g. patching).
- Vulnerable devices are deeply embedded within networks.
- Devices will communicate explicitly with one another, behaving in dynamic ways that will change with operating context.
- Common security solutions may require more memory and power than devices can support.

Due to the diversity of IoT devices and their resource limitations (Atzori et al., 2010), the above security concerns with the *things-oriented* approach are not easily managed. The security concerns for an *internet-oriented* view of IoT include well-known IP related threats, including man-in-the-middle attacks, DDoS, and reconnaissance. Therefore an *internet-oriented* security posture that takes device vulnerabilities into account is advised.

2.1 Office of Naval Research - ESTEP

Military installations resemble self-contained cities with internally supported critical infrastructures (e.g. electrical grids, water treatment facilities, etc). These installations rely on SCADA networks to monitor and control this infrastructure. As external connectivity of these networks increases, new security threats challenge SCADA network managers. Furthermore, the DoD is increasingly integrating smart technology into new and existing SCADA infrastructures.

Energy Systems Test and Evaluation Program (ESTEP) is a five-year effort, funded by the US Office of Naval Research (ONR), to test and evaluate energy technologies for use on Navy and Marine Corps installations. ESTEP focuses on energy technologies that reduce costs, improve energy security, and increase the reach of the US Armed Forces (ONR, 2012). In this paper, we highlight the Cyber-SCADA Evaluation Capability (C-SEC), a research effort under ESTEP to improve the cyber-security posture of energy systems, starting with decision-making support for securing critical infrastructures.

3 C-SEC

C-SEC supports Cyber Security and Information Assurance decision-making across new technologies by enabling a streamlined, flexible, and repeatable evaluation process against DoD-specific needs and requirements. Traditional security evaluation techniques are expensive, as they often require time and resources beyond what projects of smaller scales can afford. In addition, these evaluations tend to be non-repeatable and ultimately lack usability and applicability beyond just that one instance, thus jeopardizing their long-term return on investment (ROI) (Romero-Mariona, 2014).

C-SEC has three main components:

- a software evaluation tool,
- a laboratory environment,
- an online collaborative environment.

The software evaluation tool walks non-SCADA security experts through a quick, high-level evaluation process for determining the highlights of specific technologies of interest. The laboratory environment integrates the technology of interest into a prescribed configuration, which then provides a more detailed evaluation. Lastly, the online collaborative environment serves as a repository of past evaluations in order to facilitate reuse of results. The following sections describe each of these components in detail.

3.1 C-SEC Evaluation Tool

The C-SEC Evaluation Tool is composed of three main parts: a process, metrics, and a framework (Romero-Mariona, 2014). The approach provides not only the process necessary to determine if a certain technology meets DoD/Navy needs, but also provides the metrics to measure how well those needs are met, and a framework to enable the comparison of multiple technologies of interest. Figure 1 below shows all three of component parts.

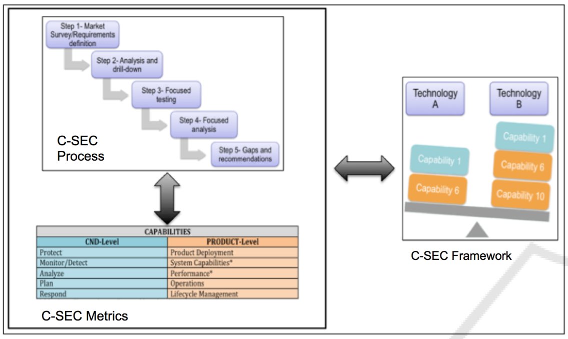


Figure 1: C-SEC Evaluation Tool Parts.

3.1.1 C-SEC Evaluation Process

The C-SEC Evaluation Process is the first major component of the C-SEC approach. It evaluates a specific cybersecurity technology to determine if it meets (or not) DoD/Navy needs. The following are the major steps of this aspect:

1. Market survey: High-level view of current offerings for the particular IA technology area of interest as well as a defined set of tests to determine the compatibility of those offerings with identified needs.

2. Analysis and drill-down: Results from the market survey are analyzed and the top percentage (varies based on each study, interest, and needs) of technologies will move down to the next step

3. Focused testing: This step takes the technologies identified during the analysis, and drill-down to test them more rigorously by means of test cases that go beyond the high-level used during step 1. Often times a simulated test environment is built to test the functionality of the test subjects. In addition, documentation reviews are also used to determine if needs are met.

4. Focused analysis: This step evaluates the results for each technology tested in Step 3 and apply metrics to determine how well each need was met.

5. Gaps and recommendations: The analyzed results are used to determine current technology gaps and suggest recommendations for future research.

3.1.2 C-SEC Metrics

Metrics, and specifically those which are related to non-functional aspects, are a tough problem. Traditional approaches to measuring are not well suited for aspects like security and usability (Romero-Mariona, 2014). As a result, researchers, industry practitioners, and the government, lack the necessary tools to baseline and track specific characteristics of today's technologies. C-SEC provides metrics support that is applicable for security and usability characteristics, as well as relevant to academia, industry, and government sectors.

C-SEC provides metrics support across three different areas:

1. Metrics Discovery and Application: Develops DoD-specific security metrics and applies them to C-SEC Process results.

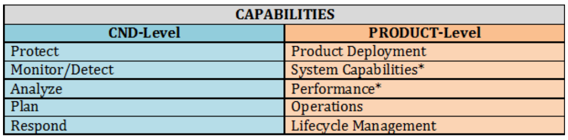
2. Metrics Manipulation: Enables manipulation and results integrity.

3. Metrics Visualization: Enables metrics traceability and decision making support.

These three areas make the C-SEC Process results usable. Figure 4 shows one of C-SEC's visualizations for metrics.

3.1.2.1 Metrics Discovery and Application

The first, and most basic, step is to develop metrics and determine the best way to apply them to the C-SEC Process results. In order to provide relevant metrics to a variety of IA technologies, we have selected ten different metrics areas, referred to as Capabilities. These Capabilities represent the highest level of granularity and cover aspects across two main areas, Computer Network Defense (CND) concepts as well as product-level. As shown in Figure 2 below, C-SEC prescribes five types of metrics (or Capabilities) under the CND area, and five types of metrics under the product-level category.



CAPABILITIES	
CND-Level	PRODUCT-Level
Protect	Product Deployment
Monitor/Detect	System Capabilities*
Analyze	Performance*
Plan	Operations
Respond	Lifecycle Management

Figure 2: C-SEC Metric Types.

The CND-level metrics refer to the basic aspects related to security, i.e. how well does a technology support the protection, monitoring and detection, analysis, planning, and response to threats and/or attacks. These types of metrics are more associated with aspects in which government programs are interested in.

The Product-level area metrics refer to aspects more commonly associated to “day-to-day” operations of a technology. Product-level metrics look at aspects that range from the cost and difficulty of deploying a specific technology, to the complexity of maintaining that technology once it is deployed. Each type of metric applied to the results obtained from the application of the C-SEC Process is assigned a numerical value that reflects how well the specific technology under evaluation meets (or not) the objectives defined for that metric.

3.1.2.2 Metrics Manipulation

Once the C-SEC Metrics have been established and applied, C-SEC supports the manipulation of these metrics in order to better understand the technology under various shades of light. C-SEC employs a granular approach to metrics manipulation; this enables flexibility as well as reusability of results. For example, suppose that Agency 1 just completed an evaluation of Technology X with an emphasis on the cost, but now Agency 2 also wants to evaluate the same Technology X but with a different emphasis on protection capabilities. Agency 2 could reuse the same C-SEC results that Agency 1 produced, and manipulate the C-SEC Metrics to put more weight into the protection aspects of the results (and less on the cost aspects) in order to obtain a different measurement of technology X’s ability to meet those needs.

C-SEC Metrics prescribe two new levels in addition to the Capability-level described in section 2.3.1, which further break down each Capability into Sub-Capabilities, and those into Sub-Capability Elements. As an example, Figure 3 below shows how a Capability, like Protection, is composed of two Sub-Capabilities: Vulnerability Protection and Listing (which refer to two possible ways to achieve protection). These are further broken into Sub-Capability Elements, such as Vulnerability Scanning and Vulnerability Reporting (which refer to two possible ways to achieve Vulnerability Protection).

This granular approach prescribes a few rules:

- Every Capability is composed of one or more Sub-Capabilities.
- Every Sub-Capability is composed of one or more Sub-Capability Elements.
- Sub-Capability Elements can be duplicated across other Sub-Capabilities.

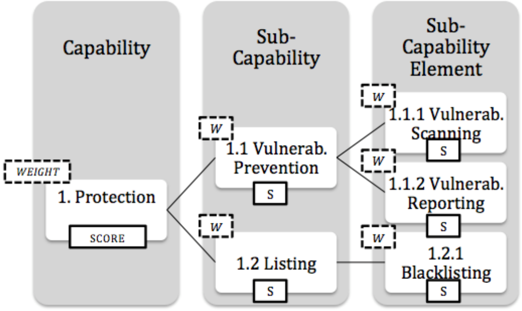
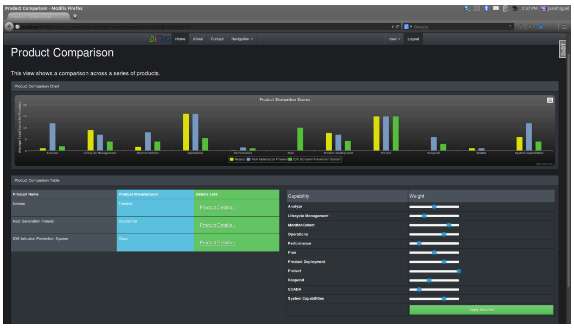


Figure 3: C-SEC Metrics Granularity.

C-SEC computes an aggregated score from various levels of granularity (Capability→Sub-Capability→Sub-Capability Element) as well as provides “weights” at each element to facilitate the flexibility and reuse of the C-SEC Metrics. This granular system is what would enable Agency 2, in our earlier example, to take the C-SEC Process results from Agency 1 and apply different weights to their scores in order to emphasize different aspects of interest.

3.1.2.3 Metrics Visualization

The last aspect supported by C-SEC Metrics is visualization. C-SEC Metrics provide a visualization for the manipulation of the various scores and weights applied to the C-SEC Process results, so that users can see in real-time the effect that changes have on the original results. The Metrics visualization component is mainly driven by C-SEC’s graphical user interface (GUI) and changes made to the original results are stored in a database. Finally, the visualization of C-SEC Metrics also supports decision-making by employing Bayesian-Network models in order to provide probabilities as well as (ROI) information.



Product Comparison

This view shows a comparison across a series of products.

Product Name	Product Manufacturer	Details Link
Nessus	Tenable	Product Details »
Next Generation Firewall	SourceFire	Product Details »
IOS Intrusion Prevention System	Cisco	Product Details »

Capability	Weight
Analyze	
Lifecycle Management	
Monitor/Detect	
Operations	
Performance	
Plan	
Product Deployment	
Protect	
Respond	
SCADA	
System Capabilities	

Figure 4: C-SEC Metric Types.

3.1.3 C-SEC Framework

The last piece of C-SEC is the Framework. This piece provides the format necessary to compare and contrast multiple technologies of a specific cybersecurity area. Furthermore, the framework also supports a repository of past and present evaluation results in order to facilitate reuse.

The C-SEC Framework serves as the key component of the online collaborative environment (to be discussed in Sec. 3.3), through which various users can share results and reuse information. While C-SEC applications are individually installed by users (clients), the Framework serves as the hub (server) that connects them together.

3.2 C-SEC Laboratory Environment

In order to further and in more detail evaluate the various security technologies of interest, we have established a SCADA laboratory environment. The C-SEC Laboratory Environment consists of several SCADA demonstration kits from various vendors, which are easily reconfigurable to simulate different environments. Using this setup, the Technology Under Evaluation (TUE) is integrated for a much more detailed evaluation beyond just the C-SEC software tool. Laboratory assets include:

- SCADA and ICS components including, but not limited to: programmable logic control units (PLCs) (Zhu and Sastry, 2010), networking equipment, valves, actuators, motors, and various other components, which create a realistic industrial environment. These components are representative of what is offered by the major SCADA and ICS suppliers.
- A DoD-mandated vulnerability scanner.
- Vulnerability visualization tool, Combinatorial Analysis Utilizing Logical Dependencies Residing on Networks (CAULDRON).
- A suite of internally developed, custom security test scripts to exercise the equipment beyond normal operation parameters.

CAULDRON, developed by George Mason University, is a network vulnerability visualization tool that takes vulnerability scan results in .xml format, parses them, and outputs a weighted network diagram (Jajodia et al., 2011). The nodes represent the IP addresses within the network and the edges show potential for information exchange. Each edge has an associated weight that represents the number of vulnerabilities between connected nodes and shows how they propagate throughout the network. This tool was designed to enable network modeling; a user could visualize their existing network, then model how a security product would change the state of the network based on placement. A set of scripts was developed to automate a number of well-known approaches to network penetration. These scripts are deployed on the SCADA network to test the effectiveness of security products.

The C-SEC process for testing the effectiveness of security technologies is as follows:

1. Perform a vulnerability scan to baseline SCADA equipment.
2. Install and configure security TUE.
3. Allow equipment to run for several weeks to generate data.
4. Re-scan SCADA equipment.
5. Visualize scan results on Cauldron.
6. Compare new and baseline results to determine effectiveness of security TUE.
7. Initiate scripted tests on network to validate scan results.

This process can be seen in Figure 5.

3.3 C-SEC Online Collaborative Environment

The C-SEC Online Collaborative environment consists of a web application providing users with an interface to create, search, and reuse standardized evaluations of security technologies that are specifically marketed for SCADA networks. This online collaborative environment was created to standardize what is currently a disparate process for evaluating security technologies and apply a set of weights that is representative of user needs. Users have the option to perform new evaluations or choose previously completed evaluations. Allowing users to choose existing evaluations enables them to make informed decisions about which technology (or suite of technologies) to implement on their networks.

The online environment is designed for easy deployment to both enterprise and tactical networks. To achieve this, C-SEC deploys virtual machines (VM) which are lightweight and have the ability to be hosted on almost any network.

The C-SEC web server sits on top of a 64-bit Linux-based platform (`http://linuxmint.com/`). Using a lightweight Linux distribution enables C-SEC to have a smaller footprint. NGiNX (`https://www.nginx.com/`) is the preferred web server to host C-SEC as it is a lightweight solution and memory-efficient, which is key when being hosted on a network with limited resources (e.g. tactical networks). NGiNX is able to handle modern concurrency issues on websites with numerous connections.

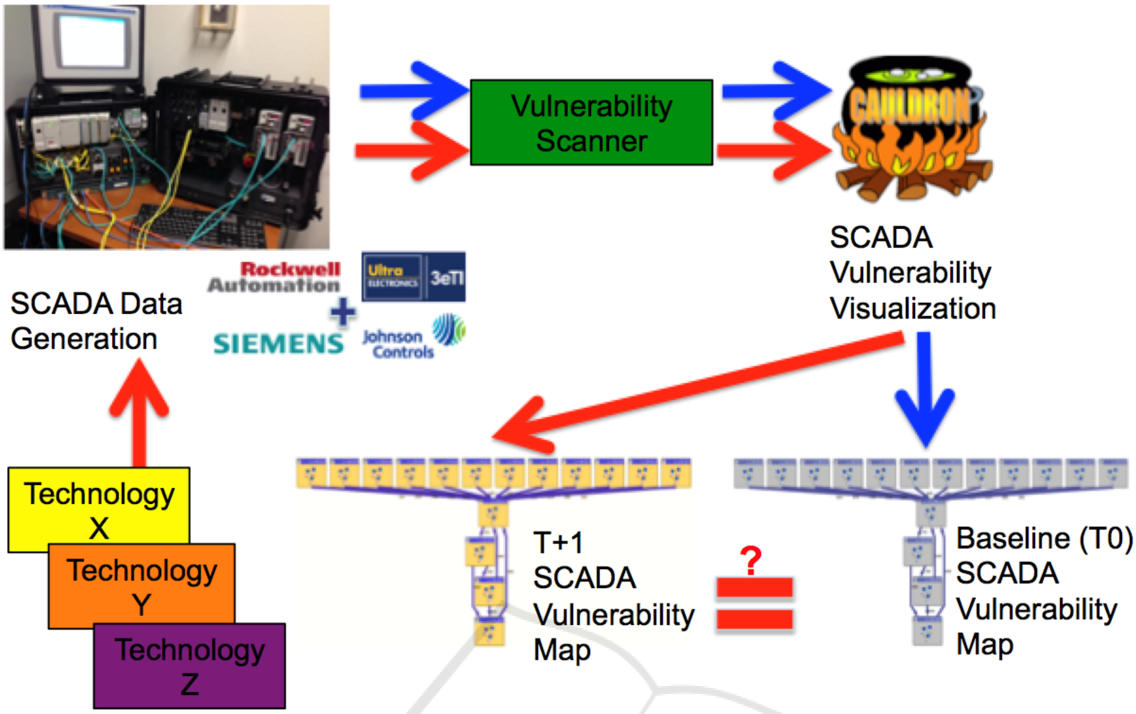


Figure 5: C-SEC Laboratory Environment.

uWSGI complements NGiNX by handling dynamic content.

The C-SEC website uses the Django framework because it provides required functionalities of a web framework, such as ease of use, scalability, and speed. Websites developed on Django use Python as the primary programming language, as well as HTML (`https://www.djangoproject.com/`). Django also allows for the incorporation of additional APIs, such as Highcharts (`http://www.highcharts.com/`).

Separating the database server from the webserver is an operational decision because running both services on a single machine is resource intensive. Turnkey Linux is a community that takes many of the top open-source applications and creates an easily deployable server with a minimum amount of components to fully operate securely (`https://www.turnkeylinux.org/`). The C-SEC database server is a Turnkey distribution built specifically host a PostgreSQL database (`http://www.postgresql.org`). The database for C-SEC's online collaborative environment has transitioned through a couple of different database iterations, finally settling upon PostgreSQL. PostgreSQL is one of the most feature-rich open source relational databases available.

An overarching goal for C-SEC is the development of a repository of evaluations for reuse as a cost-saving feature. When a user wants to reuse an existing evaluation, they have two options. First, they can take another user's evaluation at face value; they are satisfied with the answers provided by the other user and accept the score as is. Alternatively, they can reuse existing evaluations while overlaying a set of weights based on individual needs using the built in wizard to establish their preferences. Weights based on individual prioritizations are overlayed onto the existing evaluation data. The user also sees a visualization consisting of a bar graph of existing scores over each of the Capabilities and an overlay "wave" of their weighted preferences.

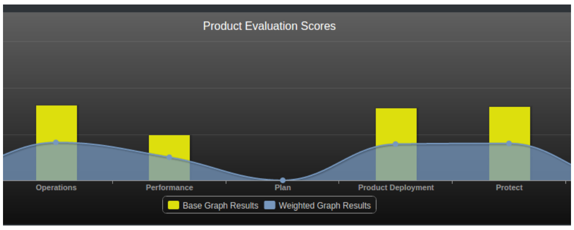


Figure 6: C-SEC "wave" overlay comparing user priorities with existing evaluations.

This allows users to directly compare their priorities to what the technology provides according to the existing evaluation, shown in Figure 6. (Hallman et al., 2014)

4 RELATED WORK

Much of the research into securing networks of IoT devices, particularly in the domain of ICS and SCADA, focuses on how to apply what we know about common devices and protocols to the IoT domain (Jing et al., 2014).

Yu discusses vulnerabilities introduced by collaborative implementations of IoT devices. Due to the vast diversity and computational simplicity of IoT devices, coupled with prolific unpatched vulnerabilities, a network-based solution, rather than a host-based solution, is the key to IoT security. The proposition is that of a software-based solution which dynamically adapts network security to a changing operational environment (Yu et al., 2015).

A simple IF-This-Then-That (IFTTT) rule commonly implemented in IoT tends to ignore the complexity of device interactions, and thus the security interpretation of a combination of known states. Yu, et.al acknowledge that threat signatures need to be collected in order to make intelligent security decisions, and so policy extraction must be implemented as a separate utility from firewalls and other IoT management protocols. The authors propose a brute force method of establishing all devices and possible device states, then deriving the environmental context and security posture from combinations of these states (Yu et al., 2015).

In order to build an attack signature library, the authors propose anonymous, incentivized crowd sourced signature collection of individual devices. To handle multiple device interactions, the authors propose a library built from the abstractions of classes of devices, with a particular focus on the key behaviors and environmental interactions (e.g. I/O) with other devices. Device interactions within this library of abstractions are fingerprinted by fuzzing the behavior space, and building a model for multi-stage attacks through graph analysis (Yu et al., 2015).

To properly utilize Software Defined Networking (SDN) to control of a network of IoT devices, the authors propose a hierarchical control structure with logical partitioning of devices based on only essential device interactions (Yu et al., 2015). In order to implement Network Functionality Virtualization (NFV) of essential devices, the authors propose a concept called micro-middleboxes (μmboxes) implemented as lightweight, rapidly configurable VMs running the lightweight Click OS. These μmboxes could be deployed and reconfigured to adapt to the changing security needs of their respective domains within the system (Yu et al., 2015).

4.1 Security Models

Axelrod discusses the observation that even though a number of studies cite the need to secure the IoT and which entities need to be secured, little is mentioned as to how to secure the IoT (Axelrod, 2015). Security takes a back seat to other concerns due to many cultural and economic factors, leading to the development of a security model based on various incentives and penalties to encourage vendors to build security into their offerings.

NIST SP800-82 R2 (Drias et al., 2015) and ISA/IEC 62443 (Stackowiak et al., 2015) standards have been developed to provide guidance to securing industrial systems, and are a first step in applying IT security designs, such as network segmentation and other fundamental security concepts, to complex industrial infrastructures. New security devices, which do not translate well to IT security, are being developed for IoT systems, such as one-way data communication diodes, “remote access devices, and protocol translating gateways” specifically for industrial protocols. A well-developed architecture with network segmentation and appropriate network gateway appliances allow monitoring of ICS processes without interfering with the time-sensitive nature of system availability (Meltzer, 2015).

Jing, et al. (Jing et al., 2014) break down the IoT domain into several layers of systems types, each of which can apply security solutions directly to the sub-types within each system. The top three layers of Jing’s security architecture are Perception, Transportation, and Application. The Perception layer is concerned with information communication and collection both in the nodes and the network, which would be protected by RFID, WSN, and RSN security solutions (to name a few). The Transportation layer involves access to the network, including core utilities and the LAN itself. The Application layer supports IoT applications, and security at this level would be targeted directly at weaknesses in IoT applications themselves (Jing et al., 2014).

Security solutions identified within Jing’s architecture address the technology itself, as well as devices which interact with that technology. For example, RFID technology security can be greatly affected by the devices that use it. Some security solutions include uniform coding, conflict resolution, trust management, and cryptographic algorithms, all of which need to be supported by devices which collect and interpret RFID data. Goals for these security implementations directly include maximizing information security during exchange, and limiting interference in the process (Jing et al., 2014).

5 CONCLUSIONS

In this paper we presented C-SEC, a tool to support cybersecurity decision making across new technologies by enabling streamlined, flexible, and repeatable evaluations. C-SEC has three components, a software evaluation tool a laboratory environment, and an online collaborative environment, and is designed to assist non-SCADA security personnel in addressing vulnerabilities in their networks. The C-SEC software Laboratory environment provides opportunities for testing security products on controlled SCADA networks as well as modeling how they will affect network vulnerabilities. We have also developed metrics for scoring security product capabilities, as well as algorithms for matching users to suites of products that address their individual needs.

Many of the security vulnerabilities that characterize SCADA networks are common to the IoT. The diversity of IoT devices, their resource limitations, and lifespans that will outlast vendor support mean that security technology cannot be broadly applied to smart devices. An *internet-oriented* approach to IoT security that takes devices into account is the only feasible strategy for addressing security concerns. C-SEC focuses on improving the cyber-security posture of SCADA networks that have long been used in machine-to-machine communication, and given the inherent difficulties of building secure smart devices, C-SEC is an ideal technology to integrate up-to-date security into the IoT.

REFERENCES

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Comput. Netw.*, 54(15):2787–2805.

Axelrod, C. W. (2015). Enforcing security, safety and privacy for the internet of things. In *Systems, Applications and Technology Conference (LISAT), 2015 IEEE Long Island*, pages 1–6. IEEE.

Curtis, S. and Wolfe, A. W. (2013). Energy-focused fusion information system integration, a nise funded capability investment project. year 1 fy 2013 report. Technical report, DTIC Document.

Drias, Z., Serhrouchni, A., and Vogel, O. (2015). Taxonomy of attacks on industrial control protocols. In *Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS), 2015 International Conference on*, pages 1–6. IEEE.

Ecosteer (2014). Open scada and the internet of things. Technical report.

Hallman, R., Romero-Mariona, J., Kline, M., and San Miguel, J. (2014). Ditec user priority designation (upd) algorithm: An approach to prioritizing technology evaluations. Technical report, DTIC Document.

Jajodia, S., Noel, S., Kalapa, P., Albanese, M., and Williams, J. (2011). Cauldron mission-centric cyber situational awareness with defense in depth. In *MILITARY COMMUNICATIONS CONFERENCE, 2011 - MILCOM 2011*, pages 1339–1344.

Jing, Q., Vasilakos, A. V., Wan, J., Lu, J., and Qiu, D. (2014). Security of the internet of things: perspectives and challenges. *Wireless Networks*, 20(8):2481–2501.

Kezunovic, M. (2002). Future trends in protective relaying, substation automation, testing and related standardization. In *Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES*, volume 1, pages 598–602 vol.1.

Meltzer, D. (2015). Securing the industrial internet of things. Technical report, Information Systems Security Association.

ONR (2012). Energize: Onr supports new energy partnership. Technical report.

Romero-Mariona, J. (2014). Ditec (dod-centric and independent technology evaluation capability): A process for testing security. In *Software Testing, Verification and Validation Workshops (ICSTW), 2014 IEEE Seventh International Conference on*, pages 24–25.

Russell, J. (2012). Scada history. Technical report, http://scadahistory.com.

Simões, P., Cruz, T., Proença, J., and Montiero, E. (2015). Specialized honeypots for scada systems. In *Cybersecurity: Analytics, Technology and Automation*.

Stackowiak, R., Licht, A., Mantha, V., and Nagode, L. (2015). Internet of things standards. In *Big Data and the Internet of Things*, pages 185–190. Springer.

Wilhoit, K. (2013). Who's really attacking your ics equipment? *Trend Micro*.

Yu, T., Sekar, V., Seshan, S., Agarwal, Y., and Xu, C. (2015). Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things. In *Proceedings of the 14th ACM Workshop on Hot Topics in Networks*, HotNets-XIV, pages 5:1–5:7, New York, NY, USA. ACM.

Zhu, B. and Sastry, S. (2010). Scada-specific intrusion detection/prevention systems: a survey and taxonomy. In *Proc. of the 1st Workshop on Secure Control Systems (SCS)*.