Pacific Northwest
NATIONAL LABORATORY

PNNL-XXXXX

Secure Software Central

The VOLTTRON™ Case

May 2019

Chance Younkin
Patrick O'Connell

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights**. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
email: orders@ntis.gov <https://www.ntis.gov/about>
Online ordering: http://www.ntis.gov

Secure Software Central

The VOLTTRON™ Case

May 2019

Chance Younkin
Patrick O'Connell

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Contents

Contents

Figures

Tables

1.0 Introduction

Our nation's government, corporations, critical infrastructure, and citizens are in harm's way in a very expensive, never-ending cyber arms race. In less than a decade, cybersecurity spending has risen by over $60 billion. In June 2015, the U.S. Office of Personnel Management saw 21.5 million social security numbers stolen, along with fingerprints, usernames, and passwords[1]—in just one of numerous incidents. The problem persists, the costs keep rising, and the bad guys keep getting gaining entry.

Most cybersecurity vulnerabilities lie in the weaknesses of software and its development. To address that truth, the VOLTTRON™ team partnered with the Pacific Northwest National Laboratory's (PNNL) Secure Software Central (SSC) team to provide a Threat Profile and a Secure Code Review of the VOLTTRON Platform. VOLTTRON is an open source agent platform for distributed sensing and control of building management systems and devices.

The SSC work arose from VOLTTRON's PNNL campus deployment and resulted in a Threat Profile containing critical assets, prioritized threats, and controls for mitigating those threats. The Threat Profile outlines controls already in place, controls not yet in place, and recommendations for implementing controls. As a companion to the software, the Threat Profile shows due diligence in cybersecurity risk management in terms of security requirements, security recommendations, security budgeting, justification, and prioritization. This report describes the SSC offerings in general and the specific offerings applied to VOLTTRON.

[1] https://www.opm.gov/cybersecurity/cybersecurity-incidents

2.0 What is Secure Software Central?

2.1 Background

Software architects, developers, and testers are directly on the path to a system compromise that could do severe damage, whether it is reputation, financial, or physical harm to people. Therefore, attention to cybersecurity is critical at all stages of the software development life cycle. The SSC helps development teams stand on that path and minimize such compromises. Whether to define security requirements, design security into the software, or assess risk, SSC has offerings for any stage of the development life cycle. SSC shows why cybersecurity in software matters and how it benefits the software architect, developer, and tester.

2.2 History

The SSC origins date back to 2014 when a research prototype, when delivered, was a big hit for its sponsor, who wanted it deployed immediately. Given the prototype nature of the effort, the two researchers had not been concerned with cybersecurity and did not have the funds, requirements, or time to build in security. At nearly the same time, the PNNL Cybersecurity Operations Center received an internally developed business application for mandatory vulnerability scans—the results of which indicated the application could not be released. The angst was high because the development cycle had ended, and the system needed to be online. What could be done? Thus, SSC was born. It was well under way by 2017 and has grown steadily ever since.

2.3 Overview

Today, SSC comprises software engineers and cybersecurity experts from PNNL's National Security Directorate (researchers and software engineers) and Information Technology Directorate (software engineers and cybersecurity experts). The goal is to establish SSC as an institutional capability offering software security services to research projects, internal business projects, and external sponsors. VOLTTRON benefits from this due diligence in the form of more secure software and a greater understanding of risk.

The SSC has seven offerings divided into two focus areas and one ***Startup*** phase (see Figure 1):

1. ***Threat Based Software Analysis (TBSA)*** assesses the threats against the software in the context of its environment. The outcome is a Threat Profile that addresses the threats and describes appropriate mitigations and controls.

2. ***Secure Software Development (SSD)*** is the cycle of designing, implementing, and testing the software with a security mind set.

As shown in Figure 1, creating the Threat Profile prior to software development is ideal, as the ideal work flow progresses from Startup to TBSA to SSD. However, SSC strives to engage a project team with the offering best suited to the project budget or development stage. SSC has adopted a "get in, provide value, get out, and come back when needed" philosophy that provides maximum flexibility. Employing a "teach-as-we-go" approach over time will shift the culture to naturally build SSC offerings into the development cycle.

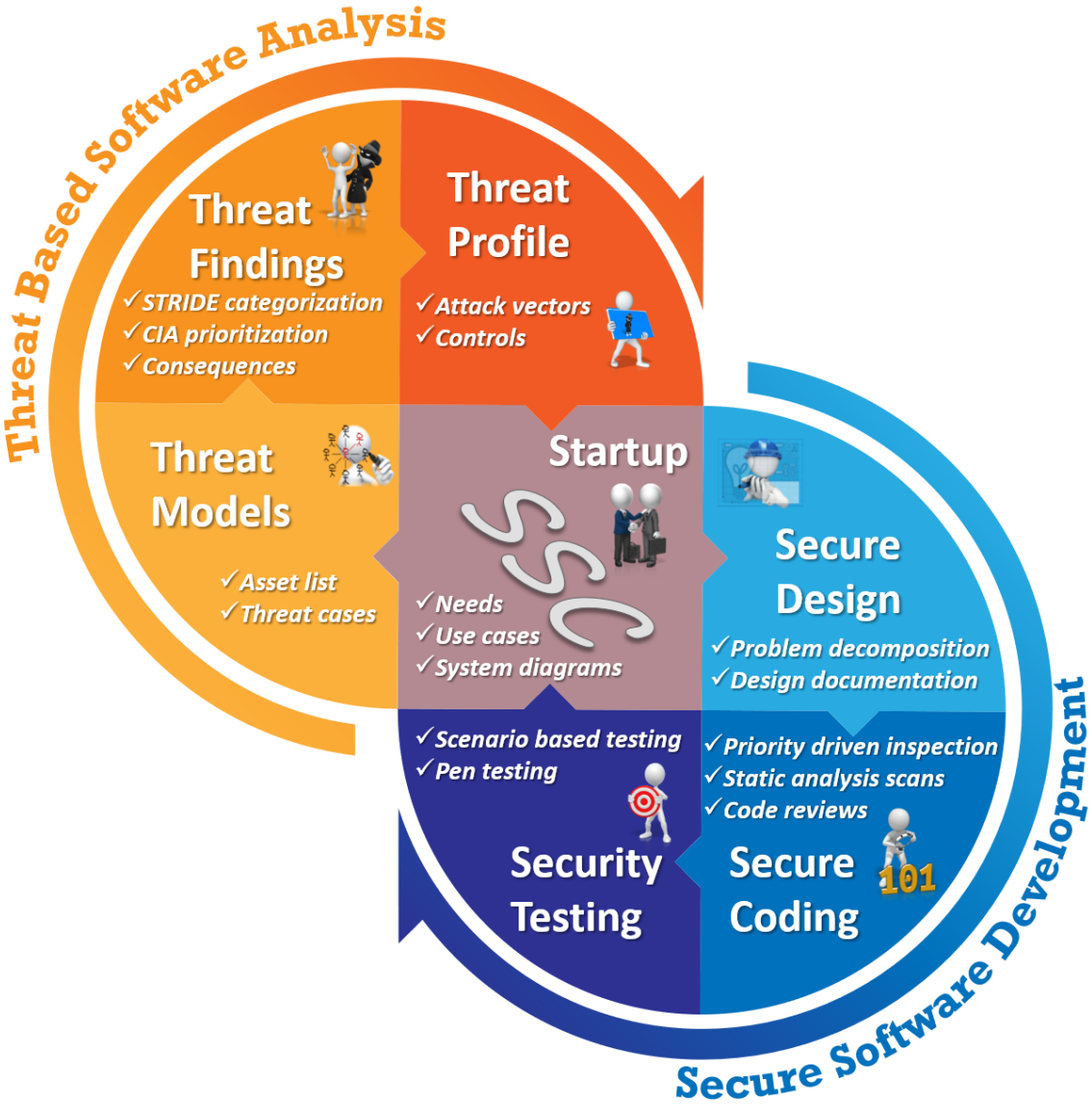

Figure 1 SSC offerings

2.4 SSC Offerings

SSC's seven offerings provide full coverage to bring cybersecurity into the complete software development life cycle. The TBSA offerings build on each other, but a project can choose to exit the process at any time with a valuable product. The SSD offerings are iterative and relatively independent. A project can again opt for a single offering or any combination. The VOLTTRON team chose to engage SSC for all TBSA offerings and for the secure code review portion of the SSD Secure Coding offering.

2.4.1 Startup

Needs, Use Cases, System Diagrams

Regardless of the development stage, the project budget, or even the most pressing need, all engagements begin with Startup. The SSC team must understand the needs of the

Development (Dev) Team and must develop use cases to gain insight into the software system. Whether the software is an age-old legacy system or an initial design, the process starts with gathering requirements, developing use cases, and creating flow diagrams.

SSC works with the Dev Team to draw an accurate flow diagram. The diagrams are based on use case discussions that guide the "drawer" who creates the diagram on a white board. Next, SSC creates the flow diagram in Microsoft's Threat Modeling Tool (TMT). The Dev Team and SSC iterate on that diagram until it is accurate—without an accurate diagram, none of the other offerings will be in the proper context, and the SSC team will not have the necessary understanding to proceed. Figure 2 shows the diagram for the VOLTTRON Campus Deployment.

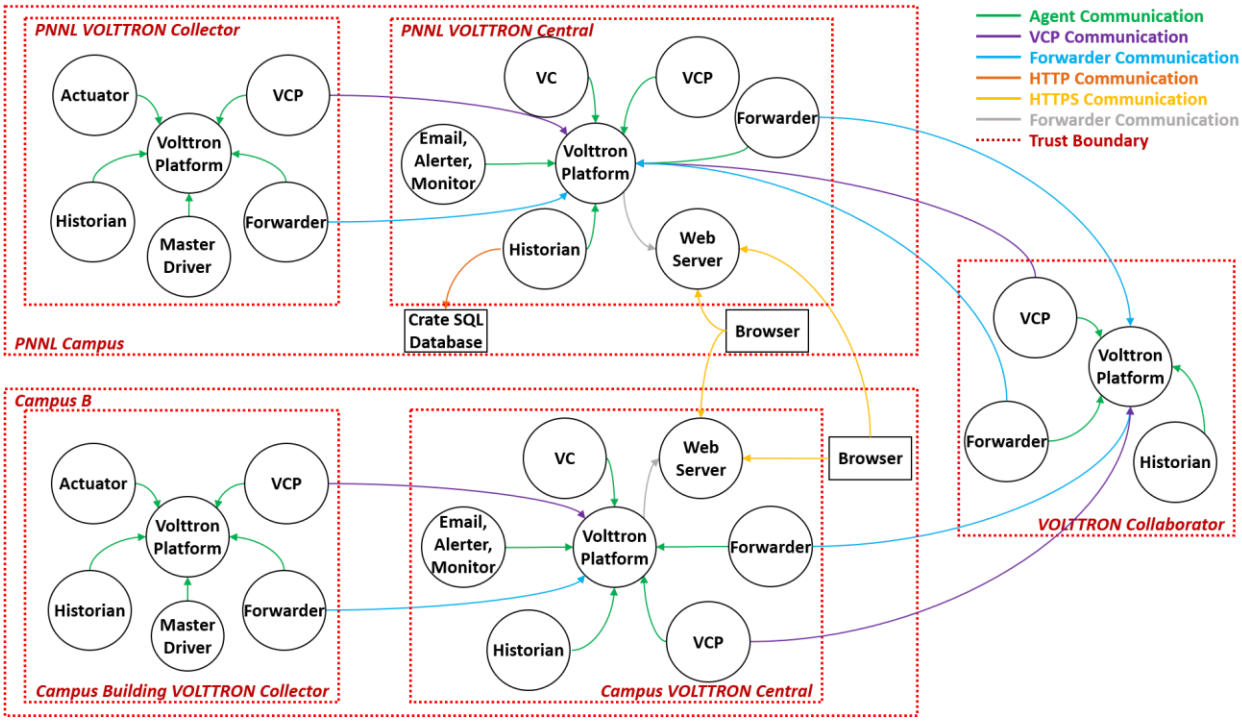

Figure 2 VOLTTRON Campus Deployment dataflow diagram

For TBSA, the diagrams lead to Threat Modeling, which is the basis for the remaining TBSA offerings. For SSD offerings, the diagrams provide context and insight for both SSC and the Dev Team. Often, the Dev Team does not have system diagrams in this format and creating them with SSC builds a valuable understanding of the system being developed. When the diagrams are complete, any of the other SSC offerings can be launched.

2.4.2 Threat Based Software Analysis (TBSA)

Threat Based Software Analysis (TBSA) determines and prioritizes threats against a software system's assets and recommends possible mitigations. The services include Threat Models, Threat Findings, and Threat Profiles. Each service builds on the previous but adds value independently.

For TBSA, SSC adopted, modified, and adapted portions of Lockheed Martin's IDDIL-ATC (see Figure 3) methodology, which uses the mnemonic: "There are no Idle (IDDIL) threats – they

attack (ATC)."[2] SSC uses adaptations of IDDIL-ATC throughout the three offerings of Threat Modeling, Threat Findings, and finally, the Threat Profile.

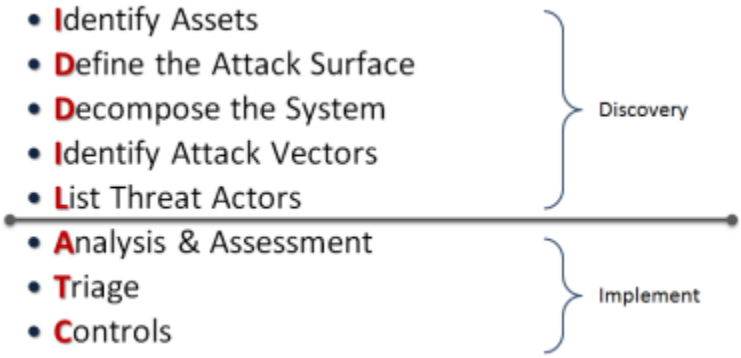

Figure 3 Lockheed Martin's IDDIL-ATC

2.4.2.1 TBSA: Threat Models (covers I, D, D, L)

Asset List, Threat Cases

SSC develops Threat Models using the flow diagrams from Startup. The list of assets to protect is derived from the nodes in the diagram, along with any other assets the Dev Team considers important. These threat cases are determined by revisiting the use cases from a threat perspective. The asset list and threat cases are documented with the diagram to produce a Threat Model that becomes the springboard to the Threat Findings.

Finalizing the Threat Model is straight forward: The SSC team takes the diagram, lists the assets, and determines an initial list of threat cases, all of which are reviewed with the Dev Team, iteratively or together, until completion.

2.4.2.2 TBSA: Threat Findings (covers A, T)

STRIDE Categorization, CIA Prioritization, Consequences

Equipped with a list of important assets and threat cases in hand, SSC uses Microsoft's TMT to begin developing the Threat Findings. The Startup diagrams are plugged into the tool, which has built-in rules to perform the initial analysis. The tool automatically assigns threat types to arrows in the diagram and categorizes them based on the STRIDE model (see Table 1). The more accurate the diagram, the better the initial results.

[2] https://pdfs.semanticscholar.org/be09/f7a16eb4a379e698d8f42100fd8a91943a0c.pdf

Threat Type	Definition	Example
Spoofing	Impersonating something or someone else	Pretending to be an administrator, enterprise, or file
Tampering	Modifying the data or code	Modifying a Dynamic Link Library on disk or DVD, or a packet as it traverses a network
Repudiation	Claiming to have not performed an action	"I didn't send that email." OR "I didn't modify that file."
Information Disclosure	Exposing information to someone not authorized to see it	Allowing someone to read the Windows source code; publishing a list of customers to a web site
Denial of Service	Denying or degrading service to users	Crashing windows or a web site; sending a packet and absorbing seconds of CPU time
Elevation of Privilege	Gain capabilities without proper authorization	Allowing a remote internet user to run commands; going from a limited user to admin

Table 1 STRIDE Model[3]

With these initial results, SSC performs additional analysis based on expertise and information from the Dev Team. The Dev Team can be involved as little or as much as they wish, with some level of consultation required to ensure objectives are met. This effort results in the Threat Findings document. A brief example of the VOLTTRON Threat Profile is shown in Table 2.

Asset	Threat Type	Priority	Consequences
VOLTTRON Platform	Elevation of Privilege	High	• An attacker may pass data into the VOLTTRON Platform to change the program execution flow to the attacker's choosing. (D1-D5, D8-D16, D19-D23, D25-D29, D31-D39) • Agents may be able to remotely execute code for VOLTTRON Platform. (D1-D5, D8-D16, D19-D23, D25-D29, D31-D39)

Table 2 VOLTTRON Threat Findings snippet

[3] Adapted from https://www.microsoft.com/security/blog/2007/09/11/stride-chart/

Each column in the Threat Findings document was derived by a specific exercise, tool, or process:

- Asset – derived by selecting important nodes from the diagrams
- Threat Type – assigned by the TMT, adjusted as needed
- Priority – derived from the Confidentiality, Integrity, Availability (CIA) Triad (see Figure 4)
- Consequences – derived through the TMT and SSC analysis

Figure 4 VOLTTRON priorities based on CIA Triad

Threat priorities are derived from the CIA Triad as shown in Figure 4, which shows the VOLTTRON priorities. SSC and the Dev Team determined these priorities as part of the Threat Findings exercise.

Another important step in Threat Findings is labeling the diagram to show the entry points for specific consequences of attack. For example, the consequences column in Table 2 has “D<n>” references, which refer to the arrows in the diagram.

The SSC team consults the Dev Team to refine assets and priorities, and to adjust the diagram as needed.

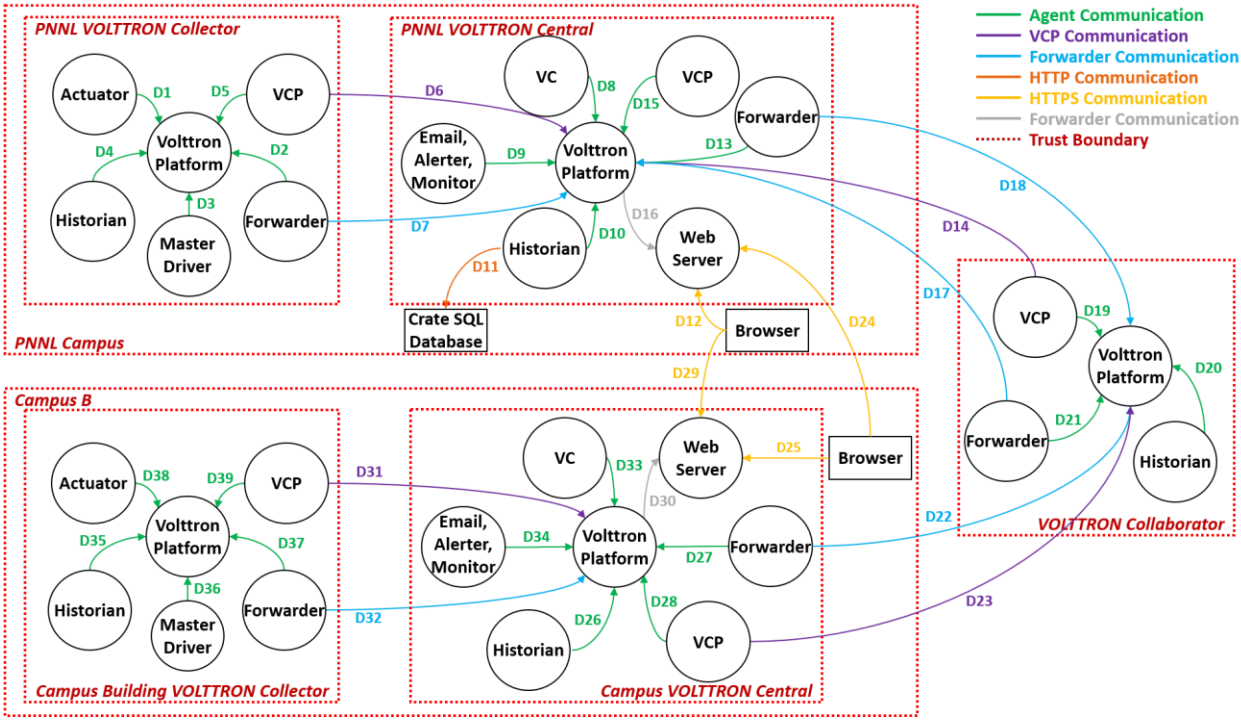

Figure 5 Diagram updated with labels to consequences in the Threat Findings

2.4.2.3 TBSA: Threat Profile (I, C)

Attack Vectors, Controls

Upon completion of the Threat Findings, the Threat Profile can be developed. This step seems simple: add the ***Attack Vector*** and the ***Controls*** columns to the Threat Findings and out comes a Threat Profile. In reality this is the most complex step, perhaps with the

exception of creating an accurate diagram. To add these two columns, SSC engages the Dev Team for several hours to determine how an attacker can actualize the consequences (the attack vector) and how to mitigate against those vectors (controls). This is done in the context of each threat and asset in the Threat Findings, so having accurate Threat Findings is key to a fully prioritized, well-understood Threat Profile.

Advancing from Threat Findings to a Threat Profile requires analyzing the findings and the software system to identify attack vectors and controls that mitigate the consequences. As controls are determined, many of the controls may already be in place or perhaps mitigated by virtue of the operating environment. This is a positive thing to discover and to document, and it is a crucial reason for developing Threat Profiles. In other cases, controls are not yet in place, which is also crucial. Whether implemented or not, the Threat Profile's controls list provides valuable information to the Dev Team, management, and stakeholders regarding the software system's cybersecurity posture.

Table 3 shows the same snippet as Table 2 with the two new columns added, thus turning the Threat Findings into a Threat Profile. For clarity, labels in the controls column refer to items in the attack vector column and labels in the consequences column refer to the diagram. Everything ties together to create the full context of the software system's security posture and its deployment environment. In Table 3, controls in bold text are those yet to be addressed.

Asset	Threat Type	Priority	Consequences	Attack Vector	Controls
VOLTTRON Platform	Elevation of Privilege	High	• An attacker may pass data into the VOLTTRON Platform to change the program execution flow to the attacker's choosing. (D1-D5, D8-D16, D19-D23, D25-D29, D31-D39) • Agents may be able to remotely execute code for VOLTTRON Platform. (D1-D5, D8-D16, D19-D23, D25-D29, D31-D39)	AV7. Actor with publish access to message bus passes data that change calculations. AV8. Actor with publish access to message bus passes Remote Procedure Call (RPC) data to cause the VOLTTRON Platform to call agent RPC functions (for agents with that functionality exposed). AV9. Actor with publish access to message bus passes RPC data to cause the VOLTTRON Platform to call VOLTTRON platform control agent service functions. AV10. Actor with access to message bus could issue command to VOLTTRON platform control agent to shut down. AV11. Actor with control of malicious agent can modify VOLTTRON home, which contains sensitive and privileged files. AV12. Actor with control of malicious agent has agent spawn a shell.	3. **Agent processes run as a separate user than the VOLTTRON platform. (AV11)** a. Possible implementation: Whitelist commands using pattern matching that can be executed using sudo. **4. Only device driver can publish to the device topic (implemented but not currently deployed). (AV7)** 5. **Limit Remote Procedure Calls to the control agent by capability (implemented but not currently deployed). (AV8, AV9, AV10)** 6. Agents run in a user space distinct from the VOLTTRON Platform. **(AV12)** 7. **Verify (actively check) agent and platform processes are not running as a privileged user. (AV12)**

Table 3 VOLTTRON Threat Profile snippet

The full Threat Profile used for the Table 3 snippet was delivered to the Dev Team, the VOLTTRON sponsor, and to the general VOLTTRON community. The intent is to provide awareness of the state of VOLTTRON™ security and to enable further security exercises using SSC offerings to bolster the security of external deployments of VOLTTRON.

2.4.3 Secure Software Development

SSD provides tools and practices that enable development teams to incorporate cybersecurity into all phases of the software development life cycle. While the Threat Profile provides controls against threats, it also serves as a launching point for gathering other requirements, initiating design, establishing test procedures with built-in security, and security testing practices. The SSD services include Secure Design, Secure Coding, and Security Testing. Ideally these services are performed in that order but can readily be executed independently and in any order.

2.4.3.1 SSD: Secure Design

Problem Decomposition, Design Documentation

SSC is developing practices and procedures for Secure Design, largely based on problem decomposition, traditional design methods, and Microsoft's Secure Software Development Lifecycle. While SSC performed a pilot on a small project, research and development efforts continue. VOLTTRON was not involved in this effort and currently has no plans to do so.

2.4.3.2 SSD: Secure Coding

Priority Driven Inspection, Static Analysis Scans, Secure Code Review

Because VOLTTRON is mature and has been in production for six years, the Dev Team opted for a ***secure code review***, an element of the Secure Coding offering. For this review, SSC made use of Checkmarx, a commercial off-the-shelf static analysis tool licensed for laboratory-wide use at PNNL. SSC met with the Dev Team to review the Checkmarx scan report and inspect the details, and it was determined that no further action was needed. Had any vulnerabilities been found, the SSC team would have engaged the Dev Team to prioritize and map the issues to the Threat Profile, if appropriate.

2.4.3.3 SSD: Security Testing

Scenario Based Testing, Pen Testing

SSC is developing practices and procedures to conduct software security testing. The effort will be based on collaborations with PNNL's IT directorate and the Cybersecurity Assessments team on the research side. This will ensure that business systems and sponsor-bound systems will benefit from SSC Security Testing. VOLTTRON has not engaged in security testing at this point, but it is possible in the future.

3.0 SSC Impact on VOLTTRON

3.1 Benefits of a Threat Profile

Threat Profiles are ultimately intended to protect an organization's reputation. SSC enables software security due diligence by providing these assessments. The benefits include:

- Creating knowledgeable security requirements – controls listed in the Threat Profile become security requirements for future software revisions. They are prioritized based on threats and assets and lead directly to tasks. If already implemented, the document serves as a set of requirements to ensure no controls are undone by development work.
- Yielding actionable controls – for controls not already in place, action can be taken to implement those controls deemed necessary based on need and priority.
- Providing justification for taking security measures – if funding is needed to implement controls, the Threat Profile becomes a mechanism for justifying the need for funds to implement security measures. Because the controls are mapped directly to prioritized threats, it translates readily into a budget request.
- Communicating risk to customer – for controls that a stakeholder chooses not to implement, the Threat Profile shows the risk of inaction. This Profile provides the stakeholders the situation awareness and the knowledge to base decisions on actual consequences.

The VOLTTRON Dev Team is tasked to implement the controls from the PNNL Campus Deployment Threat Profile. There is also a task to do a follow-on Threat Profile, which will benefit the team and the sponsor in determining future security-minded development efforts. It will also be beneficial to have a baseline Threat Profile to compare against for cost estimation and reusability.

3.2 Creation of a Security Working Group

Based on the PNNL Campus Deployment Threat Profile, the VOLTTRON User Community took interest in SSC work. This led to the development of two more use cases showcased in a Community Security Report" that was delivered back to that community. The report was well received and sparked a Security Working Group (SWG). This group meets via teleconference as needed to discuss new use cases, community security priorities, and potential tasks for SSC and the Dev Team for future VOLTTRON releases.

The SWG aims to bring awareness to the community and to deliver Threat Profile solutions to new industry use cases. Currently, one industry partner has requested diagrams for use in presentations to potential buyers of their product.

4.0 Future SSC Impact on VOLTTRON

4.1 External VOLTTRON Contributions

VOLTTRON recently joined the Eclipse Foundation and will likely accept code contributions from external developers. This will expand VOLTTRON development beyond PNNL to developers with unknown backgrounds. Because of this, a contribution vetting process is a necessity for maintaining quality of software and trust within the community. The security processes of SSC need to be part of this vetting. The Checkmarx scan can be an initial gate for acceptance and the Threat Profile can be a more extensive assessment of the overall security of the contributed software. Creating a Threat Profile can answer questions such as:

- Does this contribution add new threat findings to the list?
- Does this contribution circumvent existing controls?
- Does this contribution undo existing controls?

The contribution rules for accepting a piece of software must be easy to use, accurate, and non-exclusionary; therefore, additional consideration will need to be put into these processes.

4.2 Ongoing R&D at PNNL

SSC continues to improve its process by providing services to other PNNL projects. SSC is collaborating with PNNL's software engineering group to develop common processes and tool suites both for internal PNNL business development and sponsor-funded software products to be deployed outside the laboratory. By working across laboratory capabilities, sponsors will benefit directly from improved processes and increased expertise.

SSC will also work with PNNL's Cybersecurity Assessments team to develop services for SSC's Security Testing offering. The assessment team brings experience in penetration testing, requirements-based testing, and vulnerability assessments. This enables SSC to ensure that security policies, requirements, or Threat Profile controls are implemented and function as intended.

5.0 Conclusion

VOLTTRON has become an enabling technology for numerous commercial companies. The platform's flexibility and open-source nature have empowered companies to build products on top of the platform and make it a key component in their business strategy. For example, several companies are using VOLTTRON for data collection and transport to their cloud analytics services. One company is building large-scale solar power to battery solutions, and a subcontractor started a new company based on assisting other companies deploying VOLTTRON, which is a first for VOLTTRON. Other companies pre-install VOLTTRON on their custom hardware solutions to offer additional data collection and application environments to their customers. These companies have been open with each other and are committed to improving the platform to increase its utility for all.

Outside the commercial realm, international researchers are using VOLTTRON as the platform for their community-level energy management system. They engaged PNNL with a subcontract for workshops held at the laboratory and for continued assistance as they perform their research.

Given this active environment, cybersecurity practices throughout the software development life cycle will be crucial in ensuring that security is built into the VOLTTRON software whether it originates at PNNL, in the open-source community, or in industry. The VOLTTRON Dev Team and SSC recognize this need and are actively integrating security into VOLTTRON development.

Appendix A – Acronyms and Terms of Reference

SSC	Secure Software Central, PNNL's capability to bring cybersecurity to the full life cycle of software development
CIA	Confidentiality, Integrity, Availability – the three main objectives in conducting cybersecurity
CIA Triad	a triangle where each corner represents an element of Confidentiality, Integrity, and Availability
Consequence	the result of a successfully exploited vulnerability with impact internal or external to the organization
Control	the policy, technology, practice, or tool that mitigates a threat
Dev Team	the software development team using SSC offerings to serve a cybersecurity purpose
Penetration Testing	testing a software system's security by attacking a system to exploit its vulnerabilities
Security Requirements	specific needs from a security perspective that the software must satisfy
Threat	the joining of intent (from a malicious actor), means (weapon), and opportunity to inflict harm
Threat Findings	an initial threat assessment that contains a Threat Model diagram, a list of critical assets, a list of categorized and prioritized threats, and a list of consequences of compromise (i.e., the "what could go wrong" information
Threat Model	use cases, threat cases, and a diagram representing a software system's flow
Threat Profile	A Threat Findings table with an attack vector column and a controls column added to the Threat Findings table (i.e., the "what to do about it" information)
Vulnerability	a flaw that can be affected by a threat or hazard

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354
1-888-375-PNNL (7665)

www.pnnl.gov