Pacific Northwest
NATIONAL LABORATORY

PNNL-SA-169054

VOLTTRON™ Threat Profile for VOLTTRON Version 8.0

Provided by Secure Software Central

November 2021

Catie Himes
Patrick O'Connell
Garret Seppala
Torri Simmons
Angela Steinmetz
Chance Younkin

U.S. DEPARTMENT OF ENERGY

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

DISCLAIMER

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

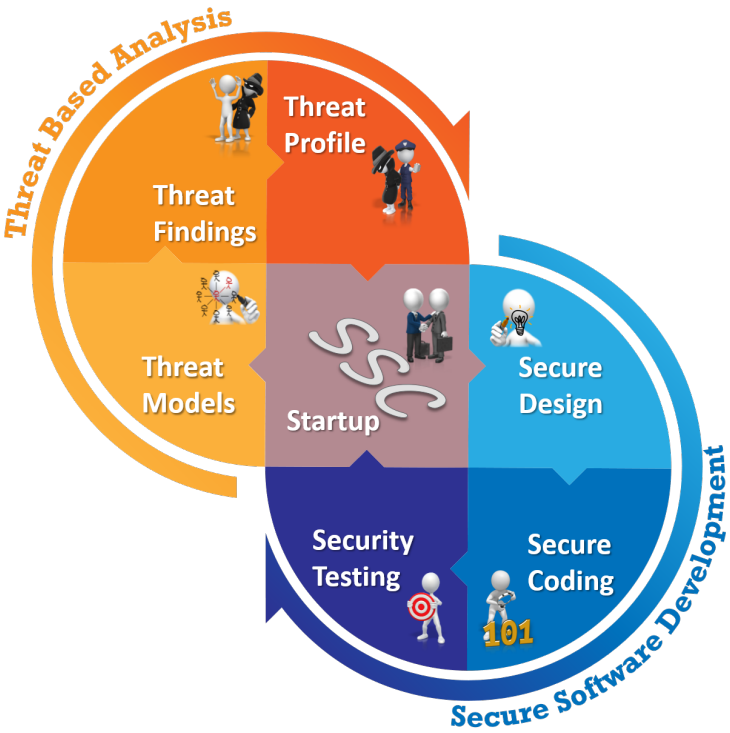
PNNL-SA-169054

VOLTTRON™ Threat Profile for VOLTTRON Version 8.0

Provided by Secure Software Central

November 2021

Catie Himes
Patrick O'Connell
Garret Seppala
Torri Simmons
Angela Steinmetz
Chance Younkin



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

Pacific Northwest National Laboratory
Richland, Washington 99354

Contents

Diagrams

Figures

Tables

Acronyms and Abbreviations

CIA	Confidentiality, Integrity, Availability
IDDIL-ATC	Identify Assets, Define the Attack Surface, Decompose the System, Identify Attack Vectors, List the Threat Actors, Analysis & Assessment, Triage, Controls
PNNL	Pacific Northwest National Laboratory
SSC	Secure Software Central
STRIDE	Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege
TMT	Threat Modeling Tool

Acronyms from NIST:

AC	Access Control
AT	Awareness and Training
AU	Audit and Accountability
CA	Security Assessment and Authorization
CM	Configuration Management
CP	Contingency Planning
IA	Identification and Authentication
IR	Incident Response
MA	Maintenance
MP	Media Protection
PE	Physical and Environmental Protection
PL	Planning
PS	Personnel Security
RA	Risk Assessment
SA	System and Services Acquisition
SC	System and Communications Protection
SI	System and Information Integrity

Summary

The VOLTTRON team at Pacific Northwest National Laboratory (PNNL) has engaged with PNNL's Secure Software Central (SSC) Team to produce this Threat Profile. The Threat Profile provides the foundation for a thorough understanding of threats for the development team, the testing team, management, stakeholders, and users of VOLTTRON. It can be used as is, or its content can be used to inform other reports tailored to a specific audience. As such, it is intended to enable decision makers at all levels to improve the security posture of the system.

For the Threat Profile, threats to the VOLTTRON system were categorized, prioritized, and mapped directly to affected system components. The table below shows the number of threats per category and per priority.

Threat Type	High Priority	Medium Priority	Low Priority	Totals
Spoofing	6	0	1	**7**
Tampering	3	5	0	**8**
Repudiation	2	3	2	**7**
Information Disclosure	4	2	3	**9**
Denial of Service	2	4	0	**6**
Elevation of Privilege	14	0	0	**14**
GRAND TOTALS	***31***	***14***	***6***	***51***

This Threat Profile provides critical information for making threat-based decisions to increase security at a reasonable cost and to reduce risk. Readers can use the Threat Profile to decide whether to implement the given mitigations or to accept threats based on their impact to the system. Not all threats must be mitigated, and not all threats can be addressed in a cost-effective way. The Threat Profile does not make these determinations, but rather provides the threats and mitigations so that others may make those determinations. The table below shows totals for completed mitigations, pending mitigations, and unneeded mitigations. Any ***unneeded*** mitigations have been deemed by VOLTTRON and SSC to be within acceptable risk tolerance to leave unmitigated.

Status	High Priority	Medium Priority	Low Priority	Totals
Completed	67	23	8	**98**
Pending	24	0	0	**24**
Unneeded	0	0	0	**0**
GRAND TOTALS	***91***	***23***	***8***	***122***

Note that there are 24 ***Pending*** mitigations (outlined in the Threat Profile) and 98 ***Completed*** mitigations. While it cannot be guaranteed that all threats, vulnerabilities, and risks will be found and mitigated, the Threat Profile shows the VOLTTRON team's due diligence in taking cybersecurity seriously. This effort leads to more secure software and better-understood security; the VOLTTRON team is to be commended for their rigorous approach to employing cybersecurity throughout the software development life cycle.

Completed: 98 of122 mitigations

1.0 Introduction

The VOLTTRON team is engaged with Pacific Northwest National Laboratory's (PNNL's) ***Secure Software Central (SSC)*** Team to provide cybersecurity analyses of the VOLTTRON software. SSC offers both threat-based analysis services and secure software development services, as defined in Figure 1. These services are ultimately used to understand and mitigate threats against software and to reduce vulnerabilities in software, thus improving overall cybersecurity and informing decision makers. SSC's threat-based analysis begins with ***Threat Models***, which are represented in a set of system diagrams. The next step is ***Threat Findings***, which consists of the threat models, use cases, and the threat findings. The final step is the ***Threat Profile*** (this document), which contains not only the Threat Findings, but also actionable mitigations that can be implemented against the threats, which is the ultimate objective of SSC threat-based analysis.

Threat-Based Software Analysis **–** determines and prioritizes threats against the software system and recommends mitigations. The result is a Threat Profile that contains a threat model, threat findings, and mitigations.

Secure Software Development **–** applies security best practices to the software development life cycle. This includes secure design, secure code review, vulnerability scanning, and security testing.

Figure 1. Secure Software Central services.

1.1 Purpose of the Threat Profile

The Threat Profile establishes security requirements, justifies security measures, yields actionable controls, and effectively communicates risk. To that end, it can be effectively used by development teams, software architects, managers, and stakeholders. For stakeholders and managers, the Threat Profile shows what has been mitigated and what has not been mitigated, thus enabling decision makers to assess priorities based on the actual system and the threats against it. For development teams and software architects, the Threat Profile provides direct and actionable tasking that boosts the cybersecurity of the software product. In addition to providing information, the format of the Threat Profile maps mitigations to threats and threats to the diagram, making it clear where and how the controls are affecting and benefiting the system. This is advantageous compared to controls and vulnerability assessments that are not threat based and do not stem from system diagrams.

1.2 Categorizing and Prioritizing Threats

Categorizing threats helps identify, organize, and prioritize threats in any system—this holds true for VOLTTRON. To optimize the analysis process, streamline the engagements, and aid in mitigation, SSC utilizes Microsoft's STRIDE model (see Figure 2). There are many categorization models, but STRIDE best lends itself to PNNL's processes, and tools are available to partially automate and expedite the initial analysis processes. SSC uses Microsoft's Threat Modeling Tool (TMT), which is based on the STRIDE model. The tool provides initial results, and the SSC team provides expertise to consolidate the threats.

Spoofing **–** when a process, file, website, network address, etc. is not what it claims to be
Tampering – the act of altering the bits in a running process, data in storage, or data in transit
Repudiation – involves an adversary denying that something happened
Information Disclosure – when the information can be read by an unauthorized party
Denial of Service – when the process or data store is unable to service incoming requests
Elevation of Privilege – when an adversary gains increased capability on a system or network

Figure 2. Microsoft's STRIDE model described.

Prioritizing threats is also critical to the process of developing a Threat Profile. With a list of mitigations, each with their own cost, level of effort, and consequences, it is necessary to understand which ones are most important and why. For a Threat Profile, priorities start with the standard CIA (Confidentiality, Integrity, and Availability) Triad, as used in Figure 3. The terms are defined simplistically as follows:

Confidentiality – keep the data secret.

Integrity – make sure the data is correct.

Availability – make the data available.

These terms are important considerations when prioritizing threats, but using the triad necessitates that one of the three must be ranked as the most important. Figure 3 shows the VOLTTRON priorities for this Threat Profile.

Figure 3. VOLTTRON priorities.

1.3 Types of Mitigation

Mitigations identified in this Threat Profile fall into three categories:

Physical – This is the traditional type of security in which valuable assets are guarded with guns, guards, and gates. However, physical security is becoming blended with cybersecurity in many ways because computers and networks are linked with gates, locks, and other access protection devices.

Technical – This refers to cybersecurity technology that is applied to typically (but not always) digital assets. Multi-factor authentication is a good example of a technical mitigation for access control.

Operational/Administrative – This is a method of following policy or procedural practices to implement security.

While these three types are not identified directly in the Threat Profile, it is important to note that most of the mitigations fall into the technical category, although both physical and operational do occur.

2.0 Threat Model

An SSC threat model is a set of use cases, a set of abuse cases, and a set of system diagrams. Use cases are descriptions of how the system operates from a user's viewpoint. They are invaluable for deriving system diagrams, which are the framework for Threat Findings and the Threat Profile. Abuse cases are just like use cases, but from the perspective of an adversary, abuse cases are used primarily to help derive and understand mitigations.

2.1 Use Cases

Use cases are descriptions of how the system operates from a user's viewpoint. They are invaluable for deriving system diagrams, which are the framework for Threat Findings and the Threat Profile. The following are typical users and the corresponding use case.

Building Controls Researcher: Developer, initial tester of building controls applications related to transactive energy, buildings-to-grid, energy efficiency etc. Building controls researchers interact with the system by developing and testing new agents in the VOLTTRON development hosts which have been set up for them.

System Administrator: Responsible for deployment and maintenance of the system. System administrator will administer deployments in actual deployment environments. For example, a system administrator is responsible for deploying VOLTTRON and its agents into all the buildings in the campus. This person will have access to full system.

Building Owner/Facility Operator: Building owners/operators who are interested that the deployed software is secure and does not hinder the operation of the building. Building owner/facility operators will have full access to the building and its assets. They may or may not know how to use the VOLTTRON platform.

Data Analysts: Engineers/analysts who are interested in the data collected by the platform and not the operation of the deployed software. Data engineers/analysts typically will not have access to the deployment setup. They will use the data from a database (or CSVs) and may have read only access to the database.

2.2 Abuse Cases

Abuse cases are ways in which a user can intentionally abuse the system to gain something they have no rights to. This could be things such as Spoofing, Tampering, Denial of Service, etc. The following are just brief descriptions of potential abuse cases. An actual abuse case requires an in-depth look at functional use elements, functional abuse elements, and technical abuse elements. SSC has a process for determining abuse cases at any level, but no abuse case engagements are reflected here.

Building Controls Researcher. There may be security loopholes in the code and config files developed by researchers. This could be intentionally done by the researcher, or simply an inadvertent weakness that could be intentionally exploited by someone else.

System Administrator. There may be security loopholes in the configuration (permission to the file system, devices need to be protected etc.) of the deployment can be abused. This could

be intentionally done by the researcher, or simply an inadvertent weakness that could be intentionally exploited by someone else.

Building Owner/Facility Operator. Facility operator typically provides limited access to use the building devices and is most interested that the deployed software is secure. An abuser can take full advantage of the vulnerabilities in VOLTTRON and the various building devices.

Data Analyst. If the data is accessed by a nefarious user, it could be manipulated for a wide variety of reasons, to falsify data, to distract an operator, to lead a researcher down the wrong path, etc.

2.3 Threat Diagrams

The diagram(s) in this section represent the VOLTTRON system and were derived through engagements between the SSC team and the VOLTTRON team. They contain some assumptions based on a mutual understanding about how the system will be designed and implemented.

2.3.1 Understanding Trust Boundaries

The most important aspect of performing threat-based analysis is knowing what trust boundaries are and where they are located. Interactions that cross trust boundaries are the most likely place for an adversary to inflict damage on a system. Figure 4 and Figure 5 show two sets of trust boundaries for the VOLTTRON system and explain what they are and where they are. The hierarchy of trust boundaries depicted in these figures are maintained throughout the threat diagrams.

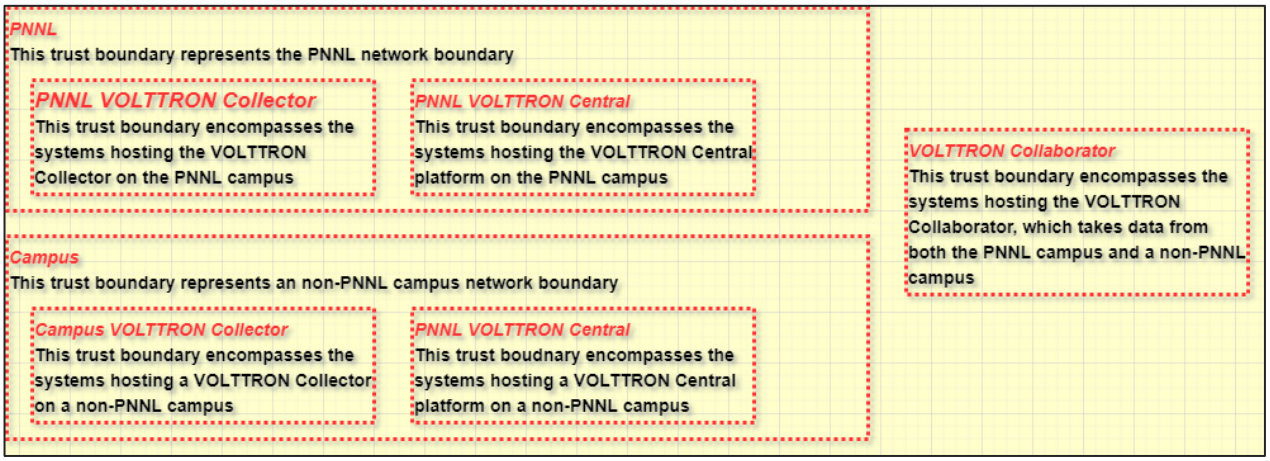


Figure 4. Trust boundaries for VOLTTRON intercampus deployment.

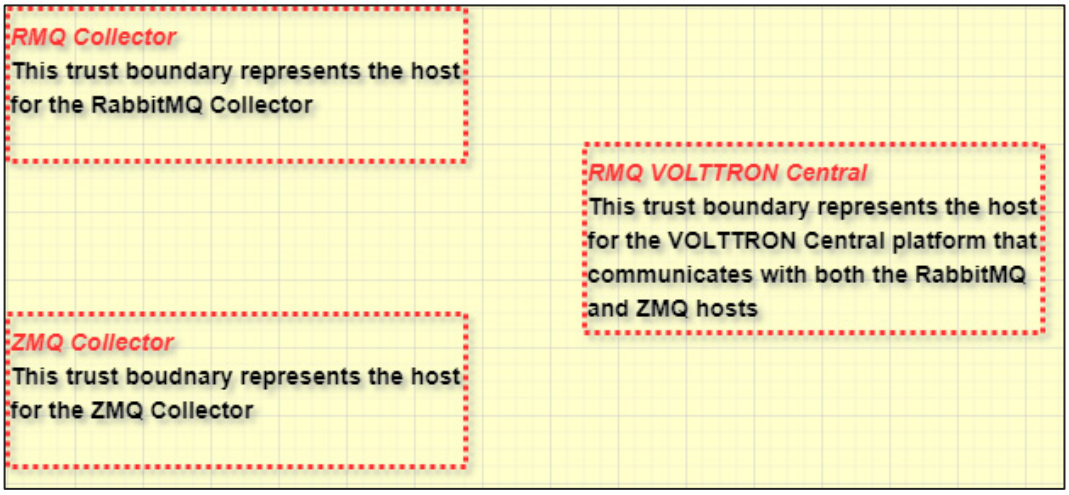


Figure 5 Trust boundaries for RabbitMQ and ZMQ legacy compatibility.

2.3.2 VOLTTRON Threat Diagrams

The conventions used in the threat diagrams below help distinguish and categorize the different components of the system as follows:

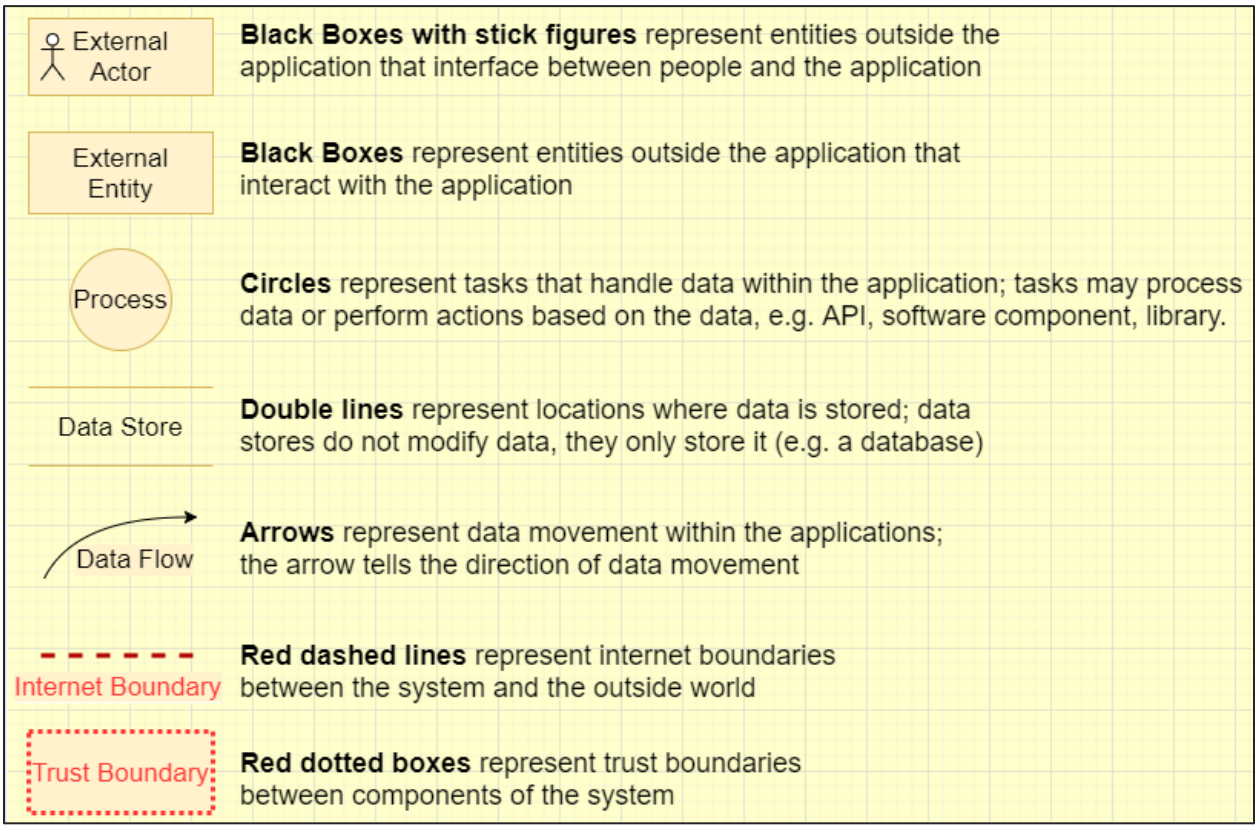


Figure 6. Legend for threat model diagrams.

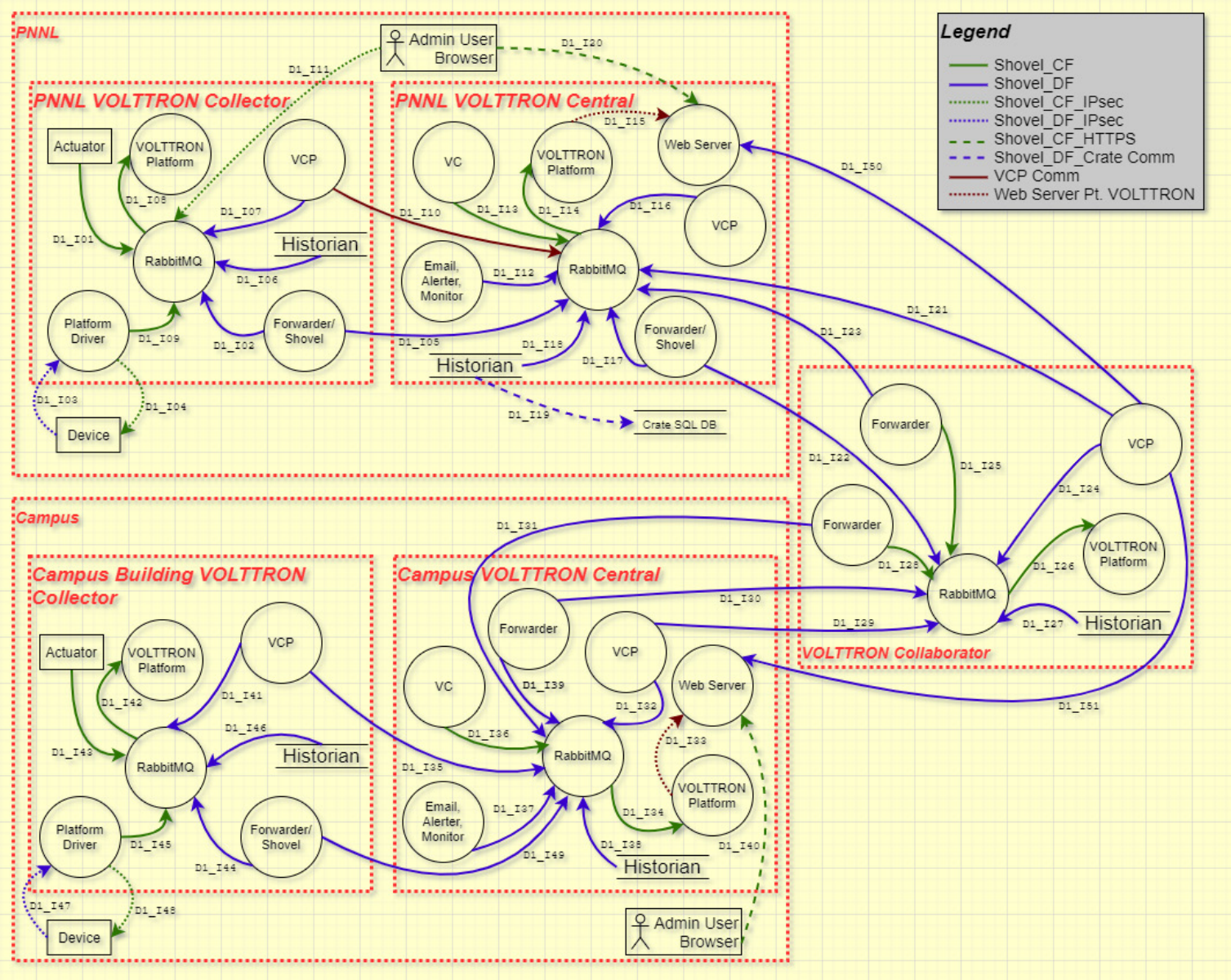


Diagram 1. RabbitMQ shovel system diagram.

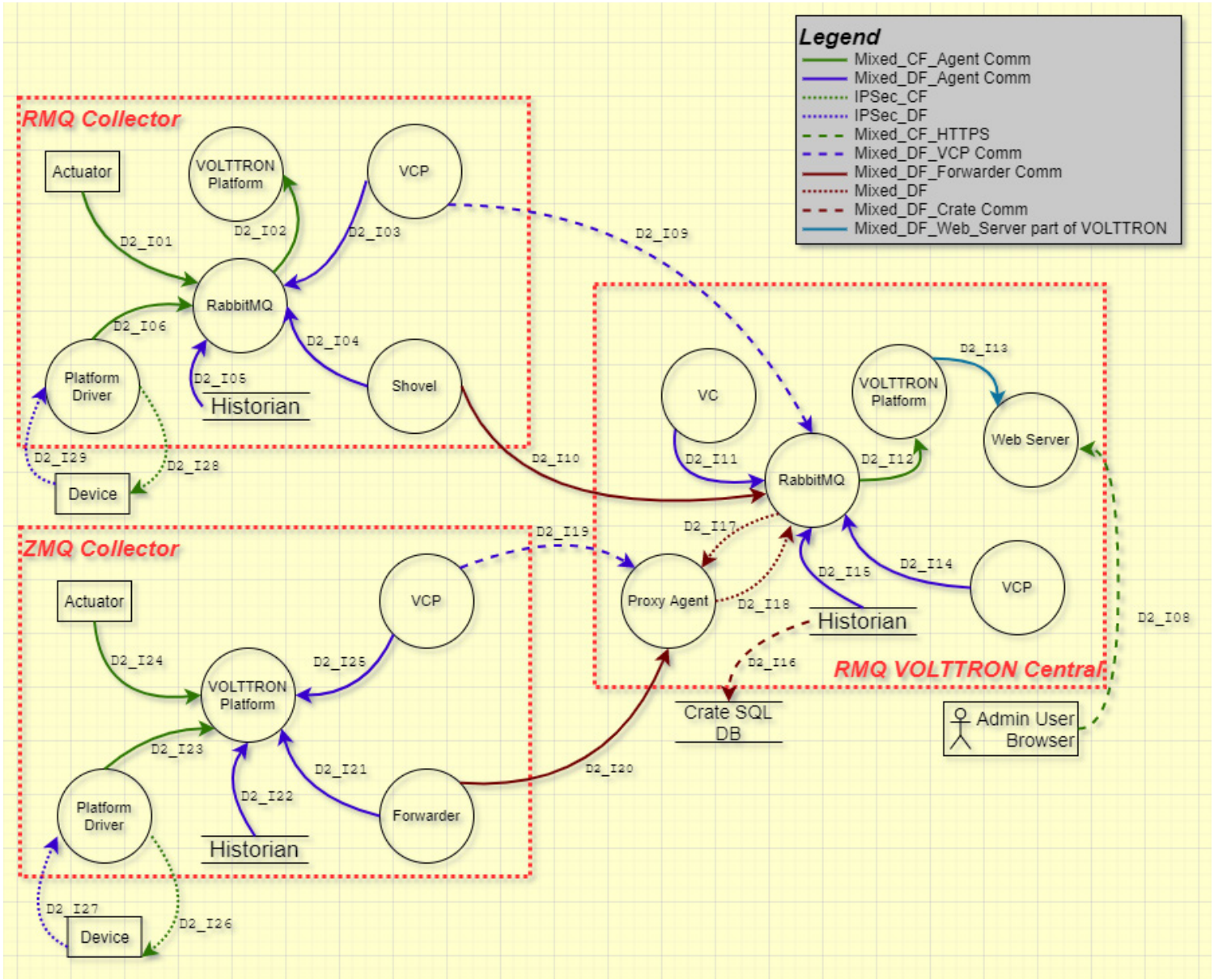


Diagram 2. Rabbit MQ/ZMQ mixed – legacy system diagram.

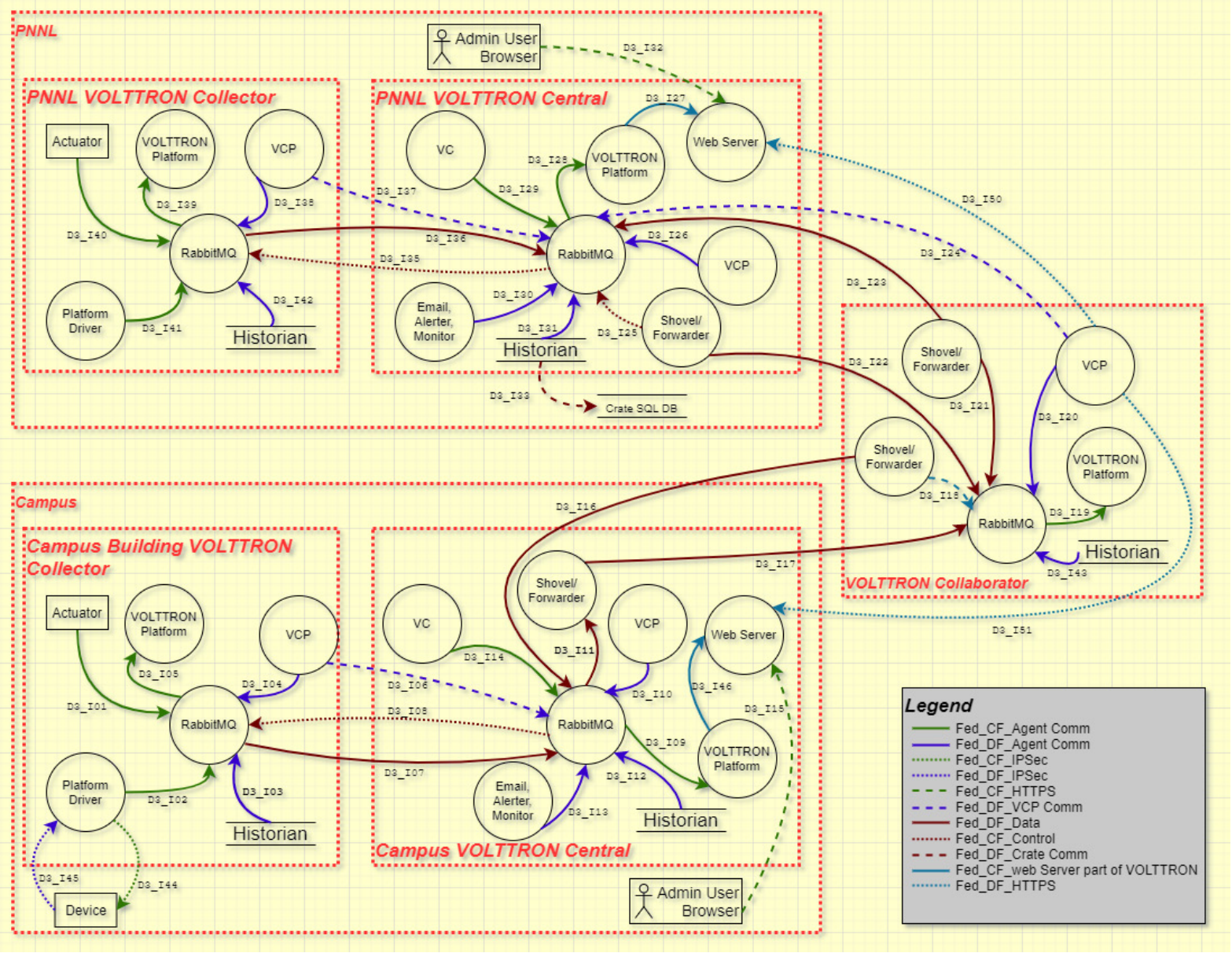


Diagram 3. Rabbit MQ federation system diagram.

3.0 Threat Profile Table

The details for all the threats, the mapping of those threats to categories, example threats, and associated mitigations are documented here. Mitigations are the main objective and describe what will be done to prevent, deter, or minimize the threat.

3.1 Interpreting the Labels

The labels captured in parentheses in the Threat column of the Threat Profile Table below refer to the diagrams above. The label refers to an interaction (arrow) in the diagram, thus showing which interaction and which components the threat corresponds to. For example, a label such as `D1_I15` refers to Diagram 1, Interaction 15. If you find Diagram 1 above, the arrow labeled `I15` will be the interaction corresponding to the threat. This strategy enables tracking of a mitigation, the threat it addresses, and the area of the diagram where the threat could occur. Thus, the table provides complete traceability from mitigation to threat to interactions between components.

3.2 Mitigation Status

Often, a Threat Profile reflects many mitigations that were already implemented to prevent the corresponding threat. These are still documented to give a complete picture of the threat landscape. In other cases, the mitigations listed are not implemented. This may be because the risk of accepting the threat is favorable to implementing the mitigation, or it could be that the mitigation is still under consideration and no decision has been made. To reflect these possibilities, the Threat Profile Table contains a ***Mitigation Status*** column. The possible values for mitigation status are:

- Completed – the mitigation is indeed implemented, and the threat is mitigated
- Pending – the implementation of the mitigation for the threat is still under consideration
- Unneeded – the risk of leaving the threat unmitigated is acceptable

Regardless of the mitigation status, the explanation and the mitigation description provide the detail to explain the situation for the purposes of due diligence, traceability, or risk management.

3.3 NIST Standards

The mitigations provided in this threat profile have been mapped to the ***Security and Privacy Controls for Federal Information Systems and Organizations, commonly referred to as SP 800-53 Rev. 4***[1] The publication was released by the National Institute of Standards and Technology (NIST). The SSC has mapped the mitigations in order to readily show compliance with NIST recommendations. For each mitigation in the threat profile table, the corresponding NIST standards are listed. Keep in mind that some mitigations map to more than one standard in the SP 800-53 document.

3.4 The Detailed Threat Profile Table

Table 1 below lists the threat type, threat, and mitigation. The table is arranged in order of priority.

[1] https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/final

Table 1 Threat Profile Table.

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
HIGH						
1	Spoofing	Crate SQL Database may be spoofed by an attacker, and this may lead to information disclosure by Historian. Such an attack is possible if the target's credentials become compromised.	D1_I19, D2_I16	1. Prevent using secure network policy. *Assumption A2	Completed	N/A
2	Spoofing	Forwarder may be spoofed by an attacker, and this may lead to unauthorized access to RabbitMQ.	D1_I31	2. Implement authentication of connection between RabbitMQ instances. 3. Only accept connections from instances that possess legitimate certificates.	Completed Completed	IA-5.2
3	Spoofing	Historian may be spoofed by an attacker, and this may lead to unauthorized access to Crate SQL Database. Consider using a standard authentication mechanism to identify the source process.	D1_I19	4. Prevent using secure network policy. *Assumption A2	Completed	N/A
4	Spoofing	RabbitMQ may be spoofed by an attacker, and this may lead to information disclosure by Forwarder. Consider using a standard authentication mechanism to identify the destination process.	D1_I31	5. Implement authentication of connection between RabbitMQ instances. 6. Only accept connections from instances that possess legitimate certificates.	Completed Completed	IA-5.2
5	Spoofing	RabbitMQ may be spoofed by an attacker, and this may lead to information disclosure by Shovel/Forwarder. Consider using a standard authentication mechanism to identify the destination process.	D3_I17, D3_I23, D3_I22, D3_I16	7. Implement authentication of connection between RabbitMQ instances. 8. Only accept connections from instances that possess legitimate certificates.	Completed Completed	IA-5.2
6	Spoofing	Shovel/Forwarder may be spoofed by an attacker, and this may lead to unauthorized access to RabbitMQ. Consider using a standard	D3_I17, D3_I23, D3_I22, D3_I16	9. Implement authentication of connection between RabbitMQ instances.	Completed Completed	IA 5.2

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
		authentication mechanism to identify the source process.		10. Only accept connections from instances that possess legitimate certificates.		
7	Tampering	Data flowing across Control Flow may be tampered with by an attacker. This may lead to a denial of service attack against RabbitMQ or an elevation of privilege attack against RabbitMQ or an information disclosure by RabbitMQ. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.	D1_I31, D3_I17, D2_I09, D3_I16	11. **Rate limit the messages allowed to be sent to RabbitMQ**. 12. Drop and do not process malformed JSON messages **13. Use threshold agents as a detection for potential data tampering**	**Pending** Completed **Pending**	**SC-5**
8	Tampering	Data flowing across Crate Comm may be tampered with by an attacker. This may lead to corruption of Crate SQL Database. Ensure the integrity of the data flow to the data store.	D1_I19	14. Ensure the integrity of the data flow to the data store. 15. Implement encryption when possible 16. Prevent using secure network policy for devices with unencrypted connections **17.** *Assumption A2	Completed Completed Completed Completed	SI-7.1 N/A
9	Tampering	SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. Any procedure that constructs SQL statements should be reviewed for injection vulnerabilities because SQL Server will execute all syntactically valid queries that it receives. Even parameterized data can be manipulated by a skilled and determined attacker. An adversary may read content stored in	D2_I16, D3_I33	18. Avoid using constructed queries in code and instead use validated and restricted stored procedures.	Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
		Crate SQL Database instances through SQL injection-based attacks				
10	Repudiation	An unlocked or unintended client device may be used to perform malicious activities	D1_I40, D1_I11, D3_I15	19. Recommend through guidance (or this document) to have a lockout policy for authenticated devices.	Completed	AC-7
11	Repudiation	Platform Driver claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	D1_I47, D3_I45	20. Alternate log specifically for auditing purposes. 21. Verify logging around relevant portions of code	Completed Completed	AU-6.1 AU-6.3
12	Information Disclosure	An adversary can gain access to sensitive data such as the following, through verbose error messages - Server names - Connection strings - Usernames - Passwords - SQL procedures - Details of dynamic SQL failures - Stack trace and lines of code - Variables stored in memory - Drive and folder locations - Application install points - Host configuration settings - Other internal application details	D1_I22, D1_I19, D1_I49, D1_I29, D1_I35, D1_I30, D1_I23, D1_I31, D1_I47, D1_I48, D1_I, D1_I32, D1_I40, D1_I11, D2_I08, D2_I20, D2_I09, D2_I19, D2_I10, D2_I29, D2_I28, D2_I27, D2_I16, D3_I17, D3_I22, D3_I07, D3_I08, D3_I24, D3_I16, D3_I06, D3_I23, D3_I45, D3_I50	22. Do not expose security details in error messages. 23. Detailed error messages are kept in logging system that requires elevated privileges to access.	Completed Completed	SA-17
13	Information Disclosure	An adversary can reverse weakly encrypted or hashed content	D1_I22, D1_I19, D1_I49, D1_I29, D1_I35, D1_I30, D1_I23, D1_I31, D1_I47, D1_I48, D1_I, D1_I32, D1_I40, D1_I11, D3_I17, D3_I22, D3_I15, D3_I07,	24. Use standard encryption algorithms AND implementations	Completed	SC-8

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
			D3_I08, D3_I24, D3_I16, D3_I06, D3_I23, D3_I45, D3_I44, D3_I50, D3_I33			
14	Information Disclosure	Custom authentication schemes are susceptible to common weaknesses such as weak credential change management, credential equivalence, easily guessable credentials, null credentials, downgrade authentication or a weak credential change management system. Consider the impact and potential mitigations for your custom authentication scheme.	D3_I08, D3_I07	25. Use Volttron Platform to generate root self-signed certificate; distribute client certificates to agents for authenticating to RabbitMQ. 26. RabbitMQ only accepts connections from valid certificates. 27. This is a standard technique for authenticating to RabbitMQ	Completed Completed Completed	SC-12
15	Information Disclosure	Secure system configuration information exposed	D1_I22, D1_I19, D1_I49, D1_I29, D1_I35, D1_I30, D1_I23, D1_I31, D1_I47, D1_I48, D1_I, D1_I32, D1_I40, D1_I11, D3_I17, D3_I22, D3_I15, D3_I07, D3_I08, D3_I24, D3_I16, D3_I06, D3_I23, D3_I45, D3_I44, D3_I50, D3_I33	28. Include a development standards rule showing config details in exception management outside development.	Completed	SA-17
16	Denial Of Service	An external agent interrupts data flowing across a trust boundary in either direction.	D1_I, D1_I32, D1_I40, D1_I11, D1_I19, D1_I49, D1_I30, D1_I23, D1_I29, D1_I35, D1_I22, D1_I31, D1_I47, D1_I48, D2_I28, D2_I26,	29. No third-party agents allowed on the central platform. 30. Historian has a persistent cache that matches storage on collection box. 31. PNNL supplies network security controls that	Completed Completed Completed	N/A N/A N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
			D2_I29, D2_I27, D2_I20, D2_I09, D2_I19, D2_I10, D3_I45, D3_I44, D3_I07, D3_I17, D3_I16, D3_I06, D3_I22, D3_I24, D3_I23, D3_I08, D3_I50, D3_I15, D3_I33	VOLTTRON deployment leverages. *Assumption A1	Completed	
				32. Secure socket shell (SSH) account access to	Completed	
				33. SSH access is not through keys, but through PNNL account via Kerberos.	Completed	
				34. Integrity of VOLTTRON agents is confirmed and verified before being registered.	Completed	
				35. Rate limit the messages allowed to be sent to the message bus.	Completed	**35-40** SC-5
				36. Recommend running VOLTTRON Platform as a service in the hardening guide.	**Pending** **Pending**	
				37. Set resource limits on agents.	Completed	
				38. **Set the limit core to zero (ulimit -c 0) for the startup shell for VOLTTRON Platform process.**	**Pending**	
				39. Retire log files by size; limit the size of log files.		
				40. **Set IP tables to rate limit new and/or unique connections.**		
17	Denial Of Service	RabbitMQ crashes, halts, stops or runs slowly; in all cases violating an availability metric.	D1_I32, D1_I11, D1_I49, D1_I30, D1_I23, D1_I29, D1_I35, D1_I22, D1_I31, D2_I09, D2_I10, D3_I07, D3_I17, D3_I16, D3_I06, D3_I22,	**41. Implement heartbeat monitoring for RabbitMQ.**	**Pending**	
				42. **Implement quick recovery from crash or low-responsive operation of RabbitMQ.**	**Pending**	

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
			D3_I24, D3_I23, D3_I08			
18	Elevation Of Privilege	Admin User Browser may be able to remotely execute code for Web Server.	D1_I40, D3_I15	43. In hardening guide, recommend using a standard and patchable web server as a proxy to the Volttron Platform web server. 44. Scan for default password hashes and request password changes or disable account. 45. Perform periodic vulnerability assessments on deployed web server 46. **Verify that public interface (vc.pnnl.gov) cannot inject JavaScript into the VOLTTRON Central process server and that server cannot execute JavaScript as a mediator.** 47. Monitor and log privileged activity on the VOLTTRON Platform by web services. 48. Implement protection against directory traversal.	Completed Completed Completed **Pending** Completed Completed	43. SI-7.1 SI-7.2 44. AC-2 45. RA-5 46. SI-3.2 SI-3.4 SI-3.5 47. AU-2 48- CM-6
19	Elevation Of Privilege	An adversary can gain long-term, persistent access to Crate SQL Database through the compromise of local user credentials	D1_I19, D3_I33	49. Rotate users' credentials such as account passwords (e.g., those used in connection strings) regularly.	Completed	
20	Elevation Of Privilege	An adversary may use unused features or services on Crate SQL Database such as UI, USB port etc. Unused features increase the attack surface and serve as additional entry points for the adversary	D1_I19, D2_I16, D3_I33	**50. Ensure that only the minimum services/features are enabled on devices.**	**Pending**	

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
21	Elevation Of Privilege	An adversary may use unused features or services on RabbitMQ such as UI, USB port etc. Unused features increase the attack surface and serve as additional entry points for the adversary	D1_I22, D1_I49, D1_I29, D1_I35, D1_I30, D1_I23, D1_I31, D1_I32, D1_I11, D2_I09, D2_I10, D3_I17, D3_I22, D3_I07, D3_I08, D3_I24, D3_I16, D3_I06, D3_I23	**51. Ensure that only the minimum services/features are enabled on devices.**	**Pending**	
22	Elevation Of Privilege	An adversary may use unused features or services on Web Server such as UI, USB port etc. Unused features increase the attack surface and serve as additional entry points for the adversary	D1_I, D1_I40, D2_I08, D3_I15, D3_I50	**52. Ensure that only the minimum services/features are enabled on devices.**	**Pending**	
23	Elevation Of Privilege	An attacker may pass data into Platform Driver in order to change the flow of program execution within Platform Driver to the attacker's choosing.	D1_I47, D2_I27, D3_I45	53. Agent processes run as a separate user than the VOLTTRON platform. **54. Only device driver can publish to the device topic (implemented but not currently deployed).** 55. **Limit RPC calls to the control agent by capability (implemented but not currently deployed).** 56. Agents run in a user space distinct from the VOLTTRON Platform. 57. Use discretionary controls for agents to use privileges. 58. Verify (actively check) agent and platform processes are not running as a privileged user.	Completed **Pending** **Pending** Completed Completed Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
24	Elevation Of Privilege	An attacker may pass data into Proxy Agent in order to change the flow of program execution within Proxy Agent to the attacker's choosing.	D2_I19, D2_I20	59. Agent processes run as a separate user than the VOLTTRON platform. **60. Only device driver can publish to the device topic (implemented but not currently deployed).** 61. **Limit RPC calls to the control agent by capability (implemented but not currently deployed).** 62. Agents run in a user space distinct from the VOLTTRON Platform. 63. Use discretionary controls for agents to use privileges. 64. Verify (actively check) agent and platform processes are not running as a privileged user.	Completed **Pending** **Pending** Completed Completed Completed	N/A
25	Elevation Of Privilege	An attacker may pass data into RabbitMQ in order to change the flow of program execution within RabbitMQ to the attacker's choosing.	D1_I32, D1_I11, D1_I49, D1_I30, D1_I23, D1_I29, D1_I35, D1_I22, D1_I31, D2_I09, D2_I10, D3_I07, D3_I17, D3_I16, D3_I06, D3_I24, D3_I23, D3_I22, D3_I08	65. Use Volttron Platform to generate root self-signed certificate; distribute client certificates to agents for authenticating to RabbitMQ. 66. RabbitMQ only accepts connections from valid certificates. 67. This is a standard technique for authenticating to RabbitMQ.	Completed Completed Completed	SC-12
26	Elevation Of Privilege	Device may be able to remotely execute code for Platform Driver.	D1_I47, D2_I27, D2_I29, D3_I45	**68. Monitor agents for detection of anomalous behavior.** **69. Prescribe limits on messages sent to agent.**	**Pending** **Pending** **Pending**	

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
				70. Only accept expected messages (input validation).	Completed	
				71. Perform periodic static code analysis on source code base for common vulnerabilities.		
27	Elevation Of Privilege	Forwarder may be able to remotely execute code for Proxy Agent.	D2_I20	**72. Monitor agents for detection of anomalous behavior.**	**Pending**	**75 SA-11**
				73. Prescribe limits on messages sent to agent.	**Pending**	
				74. Only accept expected messages (input validation).	**Pending**	
				75. Perform periodic static code analysis on source code base for common vulnerabilities.	Completed	
28	Elevation Of Privilege	Forwarder may be able to remotely execute code for RabbitMQ.	D1_I30, D1_I23, D1_I31	76. Configure RabbitMQ to only accept data from authenticated sources.	Completed	SI-10.5
				77. RabbitMQ does not process data. It just places messages (that are expected, from validated sources) on a bus.	Completed	
				78. Shovel agent has restricted permissions to RabbitMQ.	Completed	
				79. Only the topics that Volttron Central has configured can be forwarded by the Collector to RabbitMQ.	Completed	
29	Elevation Of Privilege	Forwarder/Shovel may be able to remotely execute code for RabbitMQ.	D1_I49, D1_I22	80. Configure RabbitMQ to only accept data from authenticated sources.	Completed	SI-10.5
				81. RabbitMQ does not process data. It just places messages (that are expected, from validated sources) on a bus.	Completed	
				82. Shovel agent has restricted permissions to RabbitMQ.	Completed	
					Completed	

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
				83. Only the topics that Volttron Central has configured can be forwarded by the Collector to RabbitMQ.		
30	Elevation Of Privilege	Shovel/Forwarder may be able to remotely execute code for RabbitMQ.	D3_I17, D3_I16, D3_I22, D3_I23	84. Configure RabbitMQ to only accept data from authenticated sources. 85. RabbitMQ does not process data. It just places messages (that are expected, from validated sources) on a bus. 86. Shovel agent has restricted permissions to RabbitMQ. 87. Only the topics that Volttron Central has configured can be forwarded by the Collector to RabbitMQ.	Completed Completed Completed Completed	SI-10.5
31	Elevation Of Privilege	VCP may be able to remotely execute code for Proxy Agent.	D2_I19	**88. Monitor agents for detection of anomalous behavior.** **89. Prescribe limits on messages sent to agent.** **90. Only accept expected messages (input validation).** 91. Perform periodic static code analysis on source code base for common vulnerabilities.	**Pending** **Pending** **Pending** Completed	**89 SC-5** **90 SI-10** 91 SA-11.1
MEDIUM						
32	Tampering	Attackers who can send a series of packets or messages may be able to overlap data. For example, packet 1 may be 100 bytes starting at offset 0. Packet 2 may be 100 bytes starting at offset 25. Packet 2 will overwrite 75 bytes of packet 1. Ensure you reassemble data before filtering it, and ensure you explicitly handle these sorts of cases.	D1_I19, D1_I28, D1_I31, D1_I, D1_I40, D3_I15, D3_I16, D3_I23, D3_I50, D3_I33	92. Ensure you reassemble data before filtering it, and ensure you explicitly handle these sorts of cases. 93. Implement IPSEC for communication crossing outside of Volttron trust boundary	Completed Completed Completed	IA-5.2

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
				94. Only accept connections from instances that possess legitimate certificates.		
33	Tampering	Data flowing across D2_I08_Mixed_CF_HTTPS may be tampered with by an attacker. This may lead to a denial of service attack against Web Server or an elevation of privilege attack against Web Server or an information disclosure by Web Server. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.	D2_I08	95. Prevent using secure network policy for devices with unencrypted connections 96. *Assumption A2	Completed Completed	N/A
34	Tampering	Data flowing across D2_I08_Mixed_CF_HTTPS may be tampered with by an attacker. This may lead to corruption of Web Server.	D2_I08	97. Prevent using secure network policy for devices with unencrypted connections 98. *Assumption A2	Completed Completed	N/A
35	Tampering	Data flowing across D2_I16_Mixed_DF_Crate Comm may be tampered with by an attacker. This may lead to a denial of service attack against Crate SQL Database or an elevation of privilege attack against Crate SQL Database or an information disclosure by Crate SQL Database. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.	D2_I16	99. Prevent using secure network policy. *Assumption A2	Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
36	Tampering	Data flowing across D2_I16_Mixed_DF_Crate Comm may be tampered with by an attacker. This may lead to corruption of Crate SQL Database.	D2_I16	100. Prevent using secure network policy. *Assumption A2	Completed	N/A
37	Repudiation	Device claims that it did not write data received from an entity on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	D1_I48, D2_I28, D2_I26, D3_I44	101. Accept risk of threat; no mitigation feasible as device logging is out of scope	Completed	N/A
38	Repudiation	Proxy Agent claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	D2_I20, D2_I19	102. Alternate log specifically for auditing purposes. 103. Verify logging around relevant portions of code.	Completed Completed	AU-6.1
39	Repudiation	RabbitMQ claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	D1_I49, D1_I30, D1_I23, D1_I29, D1_I35, D1_I22, D1_I31, D2_I09, D2_I10, D3_I07, D3_I06, D3_I22, D3_I24, D3_I16, D3_I08	104. Alternate log specifically for auditing purposes. 105. Verify logging around relevant portions of code.	Completed Completed	AU-6.1
40	Information Disclosure	Data flowing across D2_I16_Mixed_DF_Crate Comm may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.	D2_I16	106. Encrypt using non vulnerable version of TLS	Completed	N/A
41	Information Disclosure	Data flowing across D3_I33_Fed_DF_Crate Comm may be sniffed by an attacker. Depending on what type of data an attacker can read, it	D3_I33	107. Encrypt using non vulnerable version of TLS	Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
		may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.				
42	Denial Of Service	An external agent prevents access to a data store on the other side of the trust boundary.	D1_I19, D2_I29, D2_I26, D2_I27, D2_I28, D3_I45, D3_I44, D3_I33	108. No third-party agents allowed on the central platform. 109. PNNL supplies network security controls that VOLTTRON deployment leverages. *Assumption A1 110. Rate limit the messages allowed to be sent to the message bus. **111.** Set a cache size limit for the historian as a configuration.	Completed Completed Completed Completed	SC-5
43	Denial Of Service	Device might be phasing issues due to physical or cyber damage. That will cause the device to be unresponsive and therefore losing the control of all connected field level devices.	D1_I47, D1_I48, D3_I45, D3_I44	112. Consider the implementation and/or deployment of a network management system for monitoring the behavior of the devices. Unresponsive devices can be easily detected. If the Device is critical, consider deploying a stand-by twin.	Completed	N/A
44	Denial Of Service	Platform Driver crashes, halts, stops or runs slowly; in all cases violating an availability metric.	D1_I47, D3_I45	113. Configure automatic agent recovery.	Completed	N/A
45	Denial Of Service	Proxy Agent crashes, halts, stops or runs slowly; in all cases violating an availability metric.	D2_I20, D2_I19	114. Configure automatic agent recovery.	Completed	N/A
LOW						
46	Spoofing	Crate SQL Database may be spoofed by an attacker, and this may lead to data being written to the attacker's target	D1_I19, D2_I16	115. Prevent using secure network policy. *Assumption A2	Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
		instead of Crate SQL Database. Consider using a standard authentication mechanism to identify the destination data store.				
47	Repudiation	Crate SQL Database claims that it did not write data received from an entity on the other side of the trust boundary.	D2_I16, D3_I33	116. Alternate log specifically for auditing purposes. 117. Verify logging around relevant portions of code.	Completed Completed	AU-2 AU-6
48	Repudiation	Crate SQL Database claims that it did not write data received from an entity on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	D1_I19	118. Alternate log specifically for auditing purposes. 119. Verify logging around relevant portions of code.	Completed Completed	AU-2 AU-6
49	Information Disclosure	Data flowing across Control Flow may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.	D1_I31, D3_I17	120. Encrypt using non vulnerable version of TLS	Completed	N/A
50	Information Disclosure	Data flowing across Crate Comm may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.	D1_I19	121. Encrypt using non vulnerable version of TLS	Completed	N/A
51	Information Disclosure	Data flowing across Generic Data Flow may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or	D3_I16	122. Encrypt using non vulnerable version of TLS	Completed	N/A

#	Threat Type	Threat	Diagram location	Mitigation	Mitigation Status	NIST
		simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.				
INFO						
	N/A				N/A	

4.0 Conclusion

This VOLTTRON Threat Profile identifies threats that are mapped to specific system components. It also provides mitigations for each distinct threat–asset pairing. The outputs are actionable controls and facilitate an understanding of risk that informs decision makers who are most concerned with optimizing impact or cost. The contents of this Threat Profile inform threat-based decisions for increasing security at a reasonable cost and for reducing risk.

This threat-based software analysis illustrates the due diligence of the VOLTTRON team. In seeking an external analysis of their software, the team is assuring that VOLTTRON provides a secure and reliable capability in its operating environment.

The VOLTTRON Threat Profile provides a foundation for a thorough understanding of possible threats for the development team, the testing team, management, stakeholders, and partner stakeholders of VOLTTRON. It enables decision makers at all levels to improve the security posture of the system. This effort leads to more secure software and better-understood security. The VOLTTRON team is to be commended for their rigorous approach to employing cybersecurity throughout the development life cycle of their products.

Appendix A Brief on Threat-Based Analysis

The Secure Software Central (SSC) team combines three stages of Threat Based Analysis (TBA), as shown in Figure 7. TBA utilizes portions of Lockheed Martin's IDDIL-ATC methodology (Figure 8) to perform threat analysis. SSC optimizes IDDIL-ATC for more cost-effective, time-efficient results that lead to immediately actionable controls. Using the Lockheed Martin nomenclature, SSC actually begins with ***Decompose the System***. To accomplish this, SSC requests that ***Use Cases*** be written by members of the project team. These use cases **Error! Bookmark not defined.**provide the SSC team with valuable context in

Figure 7. The TBA half of SSC.

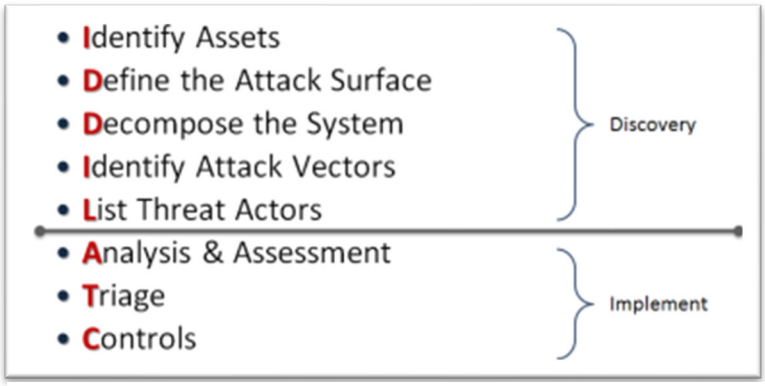


Figure 8. Lockheed Martin's methodology.

simple, non-jargon terms. With this context, the next step is to develop a set of use cases and data flow diagrams that represent the system. Generally, the assets and the attack surface can be identified using these diagrams, thus addressing the ***Identify Assets*** and ***Define the Attack Surface*** steps. From there, SSC attempts to ***List Threat Actors***, but this is not yet a rigorous exercise. The use cases, abuse cases, and data flow diagrams represent the ***SSC Threat Model***, which is the foundation for developing the Threat Profile.

SSC asks the project team to set an initial expectation of threat priority based on Confidentiality, Integrity, and Availability (CIA). The CIA Triad (see Figure 8) is a commonly used cybersecurity model.

Figure 9. The CIA triad.

The SSC team uses the data flow diagrams as input to Microsoft's Threat Modeling Tool (TMT). The TMT is a free download that comes with standard threat templates used by SSC. The TMT reads the diagrams and uses the templates to provide initial ***Analysis and Assessment*** as well as ***Triage*** results. The TMT also uses Microsoft's STRIDE model (outlined above in Figure 2) to categorize threats. The initial results from the TMT are then analyzed by SSC subject matter experts to complete the ***SSC Threat Findings*** for review by the project team.

With the Threat Findings in hand, SSC goes back to the project team to collaboratively analyze and determine mitigations (***Controls***). When this exercise is complete, the SSC team organizes the information into the final product, the ***SSC Threat Profile***.

Appendix B Brief on Secure Software Development

The Secure Software Central (SSC) Team is developing Secure Software development best practices in the areas depicted in Figure 9. While SSC will at some point offer ***Secure Design*** and ***Security Testing***, the current focus is on ***Secure Coding***. For SSC, secure coding combines Static Application Security Testing (SAST) and Open Source Analysis (OSA). The objective is to produce a Vulnerability Profile, which uses a SAST vulnerability scan of the code and an OSA scan to produce initial results. PNNL has adopted Checkmarx as the lab's vulnerability scanner, which does both SAST and OSA scans. SSC uses Checkmarx results to perform an analysis that eliminates false positives and condenses information into a simple report for use by the software development team. The full scan is also available in the Vulnerability Profile. The SSC process for creating a Vulnerability Profile is a straightforward set of steps:

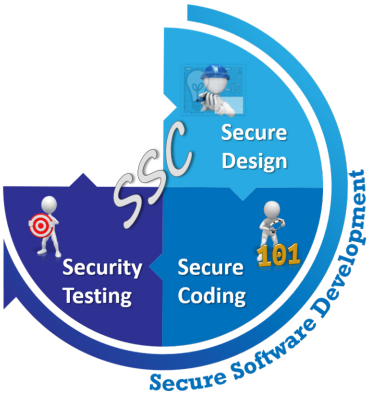


Figure 10. The SSD half of SSC

1. Receive source code from development team in the form of a zip file

   The zip file will be unzipped and used as input to the Checkmarx scanner.

2. Run Checkmarx SAST scan

   Every file contained in the zip file will be scanned with results, forming the foundation for SSC analysis.

3. Run Checkmarx OSA scan

   Dependency libraries will be checked by Checkmarx, and vulnerable libraries along with out-of-date libraries will be documented, forming the foundation for SSC analysis.

4. Analyze SAST scan results

   Results of SSC analysis are in the SAST Profile section of a Vulnerability Profile.

5. Analyze OSA scan results

   Results of SSC analysis are in the OSA Profile section of a Vulnerability Profile.

6. Deliver a Vulnerability Profile, often accompanied by a Threat Profile

Pacific Northwest National Laboratory

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

www.pnnl.gov