CESER
PUBLIC REPORTS

U.S. DEPARTMENT OF ENERGY | Office of Cybersecurity, Energy Security, and Emergency Response

Clean Energy Cybersecurity Accelerator: Cohort 2

Asimily Public Report

DECEMBER 2024 • NREL/TP-5T00-90846

CLEAN ENERGY CYBERSECURITY ACCELERATOR

This document was prepared by the National Renewable Energy Laboratory (NREL) under an agreement with and funded by the U.S. Department of Energy.

Clean Energy Cybersecurity Accelerator™ (CECA) Team

Technical Team

Amoresano, Katherine
Balamurugan, Sivasathya Pradha
Blair, Nicholas
Christensen, Dane
Davis, Max
Gonzalez, Paulie Jo
Guerra, Jennifer
Hasandka, Adarsh
Howard, Brian
Koul, Neil
Neely, Chelsea
Pailing, Courtney
Urlaub, Nik
Wallace, Anthony
Williams, Gareth

Patria Security LLC

Richardson, Bryan
Schwalm, Keith

Advice and Assistance

Abbondanza, Michael
Castellano, Anthony
Cox, Mariah
Glatter, Casey
Granda, Steve
Henry, Jordan
Lacoste, Jorge
Mujumdar, Monali
Roberts, Cari

Acknowledgments

The authors thank the U.S. Department of Energy Office of Cybersecurity, Energy Security, and Emergency Response and the Office of Energy Efficiency and Renewable Energy for supporting this effort. In addition, we thank utility industry partners Berkshire Hathaway Energy and Duke Energy for sponsoring the technical assessment.

Solution provider:

asimily

Sponsors:

Berkshire Hathaway Energy

Duke Energy

U.S. Department of Energy | Office of Cybersecurity, Energy Security, and Emergency Response

U.S. Department of Energy | Office of Energy Efficiency & Renewable Energy

Managed by:

NREL Transforming ENERGY

Notice

This work was authored by the National Renewable Energy Laboratory (NREL), operated by Alliance for Sustainable Energy LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the U.S. Department of Energy Office of Cybersecurity, Energy Security, and Emergency Response. The views expressed in the article do not necessarily represent the views of DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

National Renewable Energy Laboratory

15013 Denver West Parkway, Golden, CO 80401

303-275-3000 • www.nrel.gov

NREL/TP-5T00-90846 • December 2024

NREL prints on paper that contains recycled content.

Disclaimer of Endorsement

Reference herein to any specific commercial products, 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 Alliance. The views and opinions of authors expressed in the available or referenced documents do not necessarily state or reflect those of the United States Government or Alliance.

Acronyms

AaC	assessment as code
AD	Asimily anomaly detection server
AMI	advanced metering infrastructure
API	application programming interface
ARIES	Advanced Research on Integrated Energy Systems
BESS	battery energy storage system
BHE	Berkshire Hathaway Energy
BOE	baseline operating environment
CECA	Clean Energy Cybersecurity Accelerator™
CESER	Office of Cybersecurity, Energy Security, and Emergency Response
CIDR	classless inter-domain routing
CLI	command line interface
CPU	central processing unit
CVE	Common Vulnerability and Exposure
DB	Asimily database server
DC	direct current
DERs	distributed energy resources
DHCP	dynamic host configuration protocol
DMZ	demilitarized zone
DNAT	destination network address translation
DNP3	Distributed Network Protocol, Version 3
DNS	Domain Name System
DOE	U.S. Department of Energy
ELM	ELM MicroGrid
ERSPAN	Encapsulated Remote Switch port Analyzer
GRE	Generic Routing Encapsulation
GUI	graphical user interface
HMI	Human-Machine Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
ICS	industrial control system
IoT	Internet of Things
IP	Internet Protocol
IPSec	Internet Protocol Security
ISP	internet service provider
IT	information technology
JSON	javascript object notation
KVM	kernel-based virtual machine
LAN	local area network

MAC	Media Access Control
MSSQL	Microsoft SQL Server
NREL	National Renewable Energy Laboratory
NTP	Network Time Protocol
OS	operating system
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First
OT	operational technology
PV	photovoltaic
QEMU	quick emulator
RS232	Recommended Standard 232
RS485	Recommended Standard 485
RSPAN	Remote Switch port Analyzer
RTAC	Real-Time Automation Controller
RTU	remote terminal unit
SCADA	supervisory control and data acquisition
SDN	software defined networking
SEL	Schweitzer Engineering Laboratories
SMA	System, Mess and Anlagentechnik Solar Technology AG
SMB	Server Message Block
SNAT	source network address translation
SoH	state of health
SPAN	Switch port Analyzer
SSH	Secure Shell Protocol
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
USB	Universal Serial Bus
VLAN	virtual local area network
VM	virtual machine
VPN	virtual private network
WAN	wide area network

Executive Summary

The National Renewable Energy Laboratory (NREL)'s Clean Energy Cybersecurity Accelerator™ (CECA) program expedites the deployment of emerging operational technology (OT) security technologies to address the most urgent security concerns facing the modern electric grid. The U.S. Department of Energy (DOE) Office of Cybersecurity, Energy Security, and Emergency Response (CESER) and participating utilities sponsor the program. By working directly with utility sponsors to prioritize cybersecurity gaps and to test the ability of solutions under development to address those gaps effectively, CECA helps to both reduce the time to market for developing solutions and assure prospective adopters of the efficacy of new solutions or approaches for solving industry-wide problems.

The second cohort (Cohort 2) of the CECA program sought to address the long-standing challenge of OT asset management. Industrial control system (ICS) networks often "grow organically," so they contain a rich mix of devices developed by multiple vendors over a substantial range of time. This wide variation often impedes asset owners' ability to accurately appraise all devices (known and unknown) connected to their systems at any time. This limited visibility inherently prevents system owners from understanding the risks in their systems. CECA defined the Cohort 2 theme as "hidden risks due to incomplete system visibility and device security and configuration." Cohort 2 evaluated solutions designed to identify risks posed by the lack of asset owner visibility into ICS networks and tested the ability of solutions to do this without impact to devices or processes. The latter specifically addresses lingering industry concerns about the potential for active scanning to impact ICS processes and a subsequent reliance on limited passive discovery.

This report presents the outcomes of CECA's evaluation of the Asimily solution. The Asimily solution is a high-throughput suite of server and client machines that ingest mirrored network traffic at Asimily's edge processors. The solution then sends that traffic to a set of servers that extracts insights about the system configuration and the risks present in the system. Asimily's solution is one of a class of products designed to improve an asset owner's visibility into their environment without impeding system operations. This visibility improves the understanding of risks in the system. The CECA evaluations of the Asimily solution showed the following findings. Each instance is a known limitation of passive sampling methods:

- Important asset information about most devices in the environment was identified.
- All devices that were not identified were subject to known constraints of passive solutions, such as devices connected serially behind a remote terminal unit (RTU) or devices that did not generate network traffic that traversed one of the sampling points.

CECA tested the Asimily solution in a virtual environment deployed through NREL's Advanced Research on Integrated Energy Systems (ARIES) Cyber Range. The Cyber Range provides researchers a virtual platform in which to evaluate interdependencies among power systems and digital communication devices and networks. The ARIES cyber-physical modeling and simulation platform supports both virtual and physical deployments of variable-scale environments (NREL 2024). NREL's scalable testing environment allows for products to be tested against a large number of devices, and potentially against larger loads, depending on the tests. NREL's ARIES Cyber Range afforded CECA tools with which to emulate a multilayer, modern electric grid; visualize the effects of disruptions to the grid; and evaluate the performance of the Asimily solution and its impact on the performance and resilience of the system.

CECA also tested new capabilities of Asimily's solution for targeted active scanning, which builds on its existing passive network sampling to find additional information about the system. The targeted active scanning works by using information already collected from passive sampling to build a picture of which devices exist in the environment and which ports and protocols they are using to communicate. Based on this information, Asimily's edge processors send targeted queries to collect additional information about the hosts and protocols in the network that have already been identified. CECA found that Asimily's approach of using targeted active scanning was successful in collecting more detailed information about the assets in the environment, but it did cause temporary interruptions in the communications between one specific device and the supervisory control and data acquisition (SCADA) platform.

The CECA testing demonstrated the benefits of Asimily's hybrid methodology of passive and targeted active scanning to enhance visibility of networked devices. CECA's testing also revealed key improvements for Asimily's solution to identify more assets in an energy system environment and ensure continuous availability of all resources

when performing targeted active scanning. The identified issues and areas for improvement, in decreasing order of importance, include the following:

- Reduce temporary interruptions in the communications between devices and the SCADA platform.
- Identify a wider variety of devices and protocols.

Asimily's solution is a novel approach to addressing the limitations of passive scanning while moderating impacts on system performance that could limit the incorporation of active scanning. As the energy sector undergoes significant transformations, further refinement of such solutions is crucial for enhancing system visibility and protecting against emerging threats. The CECA program provides these insights as tangible and applicable evidence of benefits that may allow the industry to become comfortable with this class of products.

Table of Contents

List of Figures

List of Tables

1 Introduction

Market forces drive the continuous evolution of the electric sector to become more diverse, interconnected, distributed, and intelligent, with increasing integration and interconnection of nonutility systems—such as distributed energy resources (DERs)s, public communication systems, independent power producers, and cloud environments—to utility networks. Increased data exchanges between diverse assets introduce new cybersecurity challenges and complicate visibility among interconnected devices.

Cyberattacks that disrupt the critical assets, systems, and networks managed by electric utilities can pose significant, negative impacts on the economy, the environment, and public health and safety. Mitigating utility risks posed by cyber threats demands increasingly nuanced insights into the technology systems—both information technology (IT) and operational technology (OT); therefore, improving visibility into utility environments is critical to improving the cybersecurity of the evolving electric systems. Today, many energy systems' operations manage OT assets using manual processes, which can be time-intensive, error prone, and fraught with delay, making it difficult to respond to cyber incidents quickly and efficiently (NIST 2020). Further, many asset management processes are static and capture only specific points in time or are not repeatable, and they lack real-time visibility into asset status.

Future clean energy systems will feature more complex system designs and an increasing number of advanced digital components that are geographically dispersed and have more diverse operators, owners, and stakeholders. The electric sector needs more automated, dynamically responsive tools to improve asset identification and asset management as these diverse, distributed technologies are integrated into the existing electric grid.

1.1 CECA Program Overview

The U.S. Department of Energy (DOE) Office of Cybersecurity, Energy Security, and Emergency Response (CESER) sponsors the Clean Energy Cybersecurity Accelerator™ (CECA) to expedite the deployment of emerging security technologies that address the most urgent security concerns facing modern and future electric grids. Utility partners provide CECA with strategic direction and cost-sharing. The Cohort 2 utility partners include Berkshire Hathaway Energy (BHE) and Duke Energy.

Each CECA cohort focuses on a theme that is defined by CECA's utility partners. This theme is then used to select the solutions tested in the defined evaluation scenarios. The cohort participants' solutions are tested on the National Renewable Energy Laboratory (NREL) Advanced Research on Integrated Energy Systems (ARIES) Cyber Range, which provides a platform for controlled emulations in a realistic and scalable cyber-physical environment (NREL 2024). The prioritized risk for CECA Cohort 2 is "hidden risks due to incomplete system visibility and device security and configuration." Cohort 2 includes clean energy components of future energy systems that will help the electric sector assess and gain confidence in adopting new cybersecurity solutions for their evolving electric distribution systems.

1.2 Cohort 2 Theme

The solutions assessed in CECA Cohort 2 focused on identifying risks that might escape detection by asset owners due to incomplete visibility of systems or device configurations. The Cohort 2 solutions aim to improve OT system visibility to shed light on OT networks and assets and to elucidate risks. Capabilities like asset identification, attack surface enumeration, and configuration management can help OT asset owners better understand their risk posture.

CECA Cohort 2 evaluated the active and passive asset discovery capabilities of market-ready solutions, documented and analyzed results, and identified gaps in functionalities or capabilities. This report describes these results to help accelerate the adoption and improvement of these and similar solutions in the electric sector to mitigate risks. The Cohort 2 evaluations focused on testing the solutions' abilities to illuminate characteristics about the environment. Red-teaming activities, such as penetration testing the solution itself, were out of scope.

2 Solution Under Test: Asimily

Asimily is an industry-leading risk management platform that secures Internet of Things (IoT) devices for healthcare, manufacturing, the public sector, and other industries that depend on their numerous connected devices. Asimily developed the solution tested by CECA to improve visibility into connected devices by enhancing capabilities related to inventorying devices, mitigating device vulnerabilities, modeling risks, detecting threats, and responding to incidents (Asimily 2024a).

2.1 Asset Identification

Asimily's solution identifies devices by examining network traffic and parsing protocols to aid with inventory management, vulnerability mitigation, and threat detection and investigations. With a protocol analyzer, deep packet inspection, and machine learning-based analysis, Asimily classifies devices, applications, services, and connections into families. The solution is deployed locally to provide visibility into all IT, OT, and IoT traffic. This real-time traffic analysis extracts insights and then reports to a centralized server. Asimily can provide some protocol-aware analysis that can identify connected serial devices in a parent-child relationship in some specific environments and configurations (Asimily 2024b).

2.2 Deployment

CECA used two servers to deploy the Asimily services, and it installed the solution to conduct both passive monitoring and targeted active scanning. In each test, the Asimily solution was exposed to the environment to sample network traffic for 30 minutes. Each asset in the environment was configured to communicate several times per minute, so the 30-minute window was chosen to allow each connected device several cycles to respond to the received communications and for any impacts from the subsequent changes to be observed. This resulted in an environment with a combined average throughput of approximately 385 KBps, or approximately 760 MB of the total bytes transferred across the experiment network for each 30-minute test.

2.2.1 Components

The Asimily solution consists of several services that run on a server cluster and at least one edge processor. Each service can be deployed on separate hardware or in a virtual machine (VM). As shown in Figure 1, in Cohort 2, a two-server cluster configuration was deployed with one server designated as the main server and the other server designated for additional services.

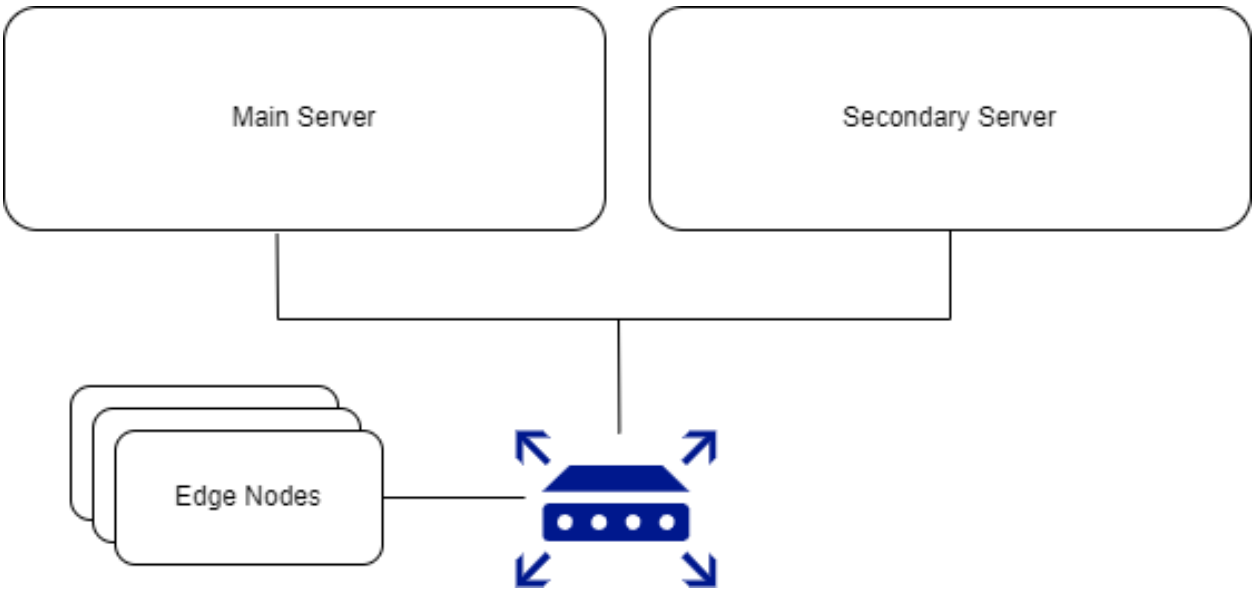


Figure 1. Asimily deployment architecture

Asimily Main Server

The Asimily main server is the focal point for both the data collection and the user interaction. Each edge processor connects to the main server, which ingests and stores the information collected by the edge processors. The other services connect to the main server to get the information reported by the edge processors and to update the main server with their outputs. Users interact with the main server through a web portal.All Asimily services can be deployed on this server or distributed across a cluster of servers, with this server acting as the head.

Edge Processor

The Asimily edge packet collector and processor—also called the edge collector, edge processor, or simply the edge–is the primary aggregation machine for network and asset discovery. Asimily edge processors are capable of performing both passive sampling and targeted active scanning methods of asset identification. To provide full system visibility, edge processors are deployed to observe the traffic across an organization's network within all the different subnetworks and across the security boundaries. Edge processors can ingest network traffic data using a variety of protocols (e.g., Switch port Analyzer (SPAN), Remote Switch port Analyzer (RSPAN), Encapsulated Remote Switch port Analyzer (ERSPAN), or Generic Routing Encapsulation (GRE) tunnels). To forward the data extracted from the traffic, the edge processors connect to the Asimily main server on Transmission Control Protocol (TCP) ports 5568 and 5574.

2.2.2 CECA Integration

CECA deployed the Asimily solution with a self-hosted, air-gapped Main/CE server and AD/DB server, and an Asimily edge processor hosted in each subnet of interest.[1] CECA evaluated the Cohort 2 solutions in two separate environments: 1) a smaller-scale generation and distribution system modeled with a photovoltaic (PV) plant, substation, and control center; and 2) an advanced metering infrastructure (AMI) environment modeled with simulated meters served by a larger substation.

PV Plant and Substation Environment

Most tests were conducted in an environment that featured a basic utility control center, a clean energy-generating PV plant (i.e., solar plant), and a substation. This environment is represented in Figure 2. This system was built to simulate the complexities that a solution could be expected to encounter when identifying assets in an industrial control system (ICS) network containing both modern clean energy components and legacy OT devices. The environment featured 13 different OT devices communicating over a variety of media, protocols, and firmware versions, as detailed in Appendix A.

The Asimily servers were integrated into this environment according to the documentation provided by Asimily, with the help of and recommendations from the Asimily team. The Asimily servers were installed in a demilitarized zone (DMZ) in the control center, and the only security change required for this integration was to allow TCP network traffic destined for ports 5568 or 5574 to reach the Asimily main server. Five Asimily edge processors were deployed in each relevant subnet that contained devices of interest.

[1] CECA tested Asimily version 5.5.2 Asimily regularly updates its software with additional functionality.

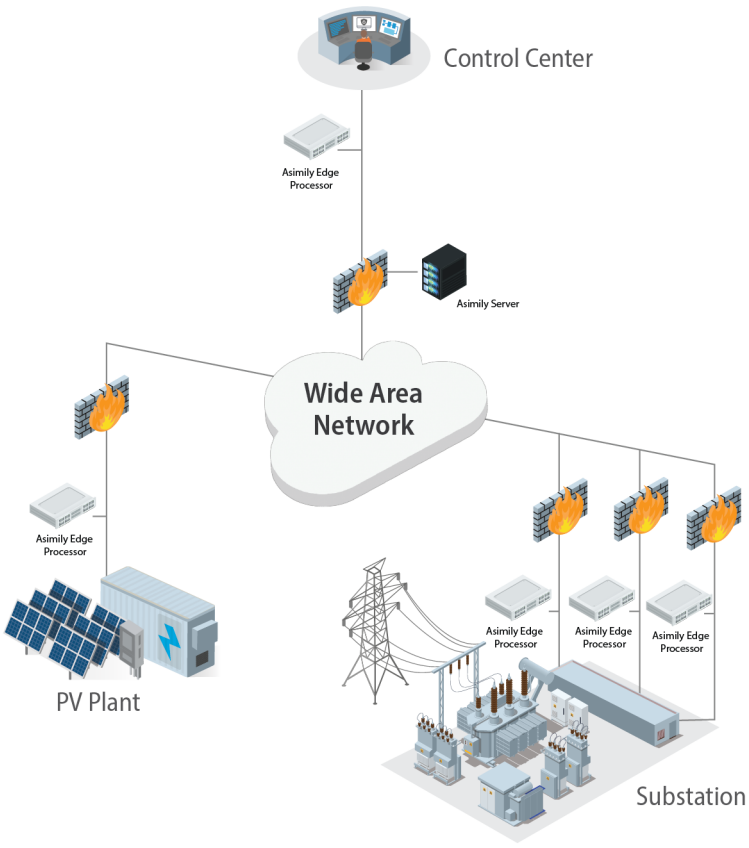


Figure 2. High-level overview of the PV plant and substation environment integrated with Asimily

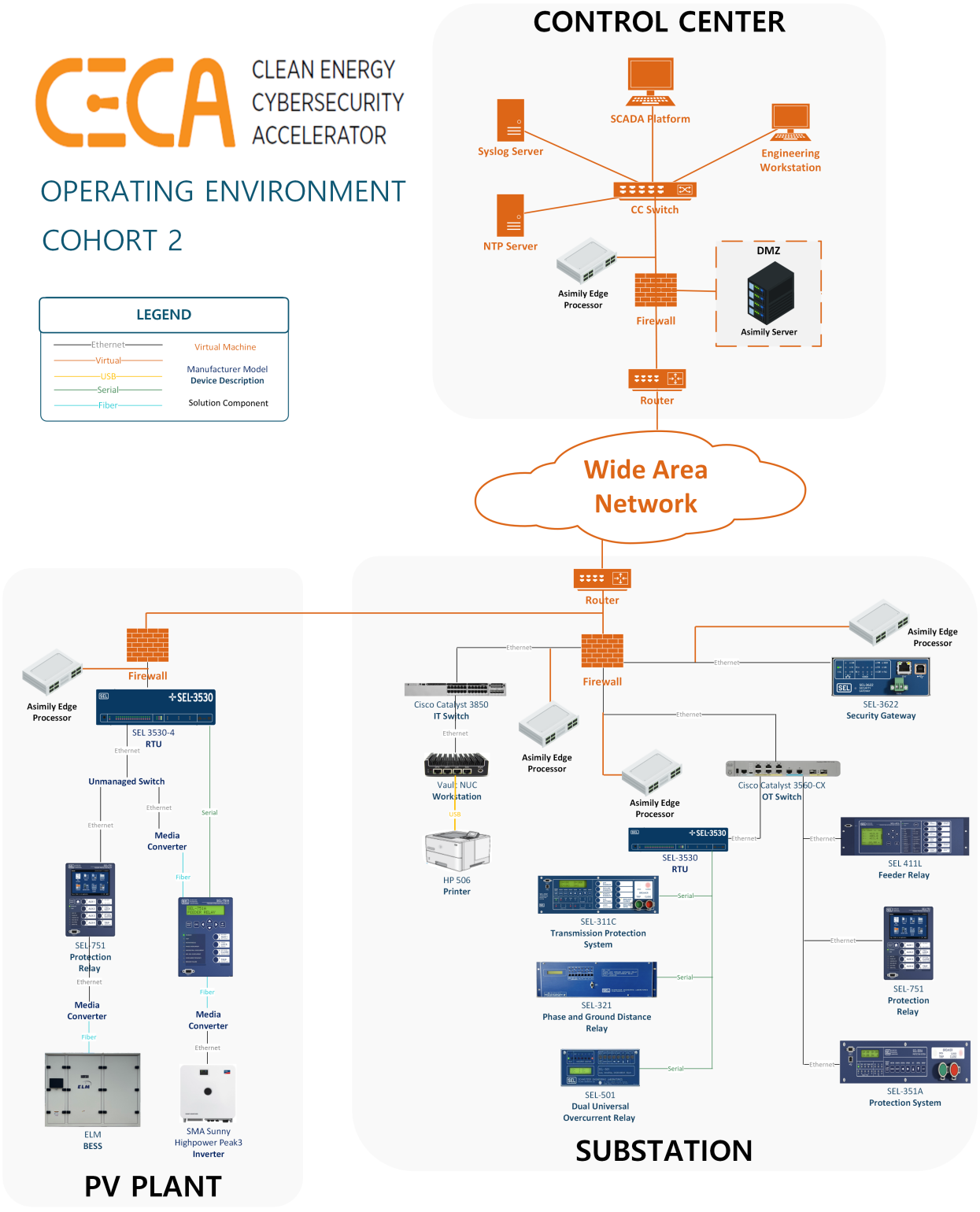


Figure 3. Diagram of the PV plant, substation, and control center integrated with Asimily

AMI Environment

To evaluate the solution at scale, CECA also integrated Asimily into a separate, larger environment featuring 2,014 advanced metering infrastructure (AMI) devices on a single flat network within the subnet `10.200.0.0/21`.

The network shown in Figure 4 represents the number of customers that could be served by a distribution substation. Used only for evaluation in Scenario 4: Scale Discovery, the AMI environment included a single Asimily edge processor and the Asimily main and AD servers were added on a separate subnet.

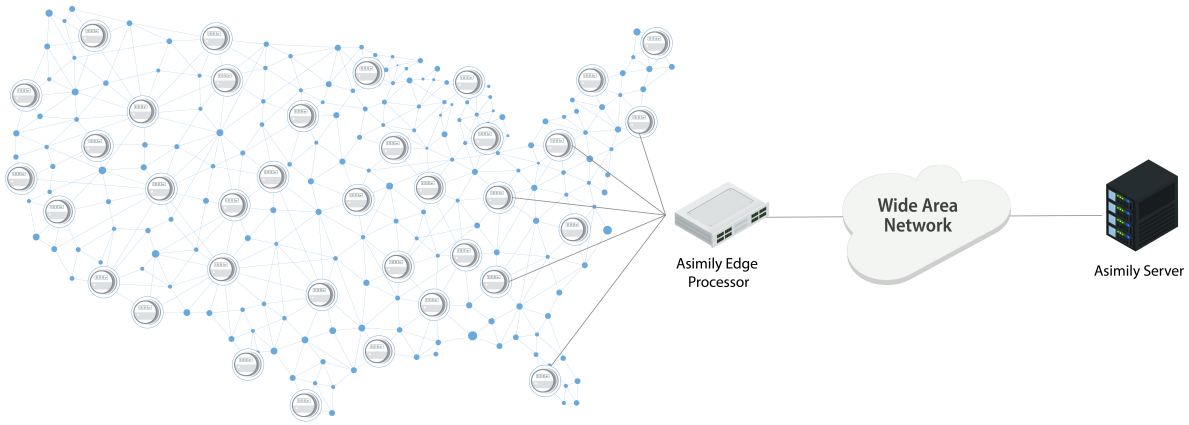


Figure 4. High-level overview of the AMI environment integrated with Asimily

3 Evaluations and Results

CECA developed an evaluation plan based on the Cohort 2 prioritized risk—hidden risks due to incomplete system visibility—to test the capabilities of each solution selected for the cohort. The evaluation plan detailed four scenarios that each tested several characteristics of the solution. Each individual scenario is a scientific, repeatable set of procedures and data collection methods. Table 1 shows which characteristics were tested in each scenario. Following the table are short descriptions of each characteristic.

Table 1. Testing matrix

	Scenario 1 Initial Discovery	**Scenario 2** Change Discovery	**Scenario 3** Alternate Discovery	**Scenario 4** Scale Discovery
Timing[2]				
Inventory accuracy	✓		✓	✓
Data richness	✓		✓	
Additional network traffic	✓		✓	✓
Change detection		✓		
Alert		✓		✓
Disruption of operations			✓	

- **Timing**: How long does it take to identify all the assets in the environment?
- **Inventory accuracy**: How many assets in the environment did the solution successfully identify?
- **Data richness**: For each identified asset, how detailed are the data collected by the solution?
- **Additional network traffic**: How much additional network traffic does the solution add to the ICS network?[3]
- **Change detection**: How does the solution track changes to assets over time?
- **Alerts**: Does the solution notify users of unexpected devices on the network?
- **Disruption of operations**: Does the solution affect any normal operations of the ICS system?

The following sections describe each test's objective and results. Details about the exact procedures for each test can be found in Appendix D. Each test was conducted three times to ensure that the data were consistent.[4] Each iteration is called a run.

3.1 Scenario 1: Initial Discovery

The objective of this scenario was to examine how a solution performs in an environment that it has never been exposed to before. Scenario 1 evaluated inventory accuracy, data richness, and additional network traffic. This scenario was run in the PV and substation environment.

The Asimily solution was exposed to the environment to sample network traffic for 30 minutes. Each asset in the environment was configured to communicate at least five times per minute, so the 30-minute window was chosen to allow each connected device several cycles to respond to received communications and for any impacts from the subsequent changes to be observed.

---

[2]The Asimily solution uses passive sampling methods or hybrid passive sampling with targeted active probing based on the results found via passive sampling. Passive solutions inherently do not have a time when the identification is "completed" because they can only make inferences based on traffic ingested. In each test, the Asimily solution was allowed to run for 30 minutes.

[3]Additional network traffic measures the total amount of data that is added to the network by the solution, not the rate at which it is added. The Asimily solution is configurable, and the rate can be constrained so that the additional network traffic can be spread out over long periods of time to achieve whatever rate an operator desires.

[4]CECA reduced the number of runs for each test from five, as used for runZero, to three because the testing encountered no discrepancies across any of the runs for any test.

3.1.1 Results

Inventory Accuracy

Inventory accuracy evaluates how many assets the solution successfully identifies. The Asimily solution successfully identified 26 of 33 assets in the environment. It did not identify the assets that did not have traffic traverse one of the sampling points. There are three reasons for this lack of identification:

- The assets were connected via serial:
  - Schweitzer Engineering Laboratories (SEL) 311C
  - SEL 321
  - SEL 501.
- The assets did not generate or receive any application layer traffic:
  - OT switch
  - IT switch
  - Sub-IT admin VM.
- The assets had local network traffic that did not traverse the sampling point:
  - SEL 751 connected to the SMA inverter.

Each asset identification limitation is inherent to passive sampling methods, which are subject to limitations based on the traffic into which they have visibility.[5]

Data Richness

Data richness evaluates the amount of detail the solution collects about each device. For the devices found, the solution identified the Media Access Control (MAC) address and MAC vendor for each device and the Internet Protocol (IP) address for 23 of 27 devices. The Asimily solution also identified other attributes about a device and its communications at several layers in the Open Systems Interconnection (OSI) model. Asimily identified the virtual local area network (VLAN) on which devices were operating, the services and ports over which devices communicated, and which other devices that the assets communicated with. Examples of services identified include Server Message Block (SMB), Distributed Network Protocol, Version 3 (DNP3), Modbus, Secure Shell Protocol (SSH), Network Time Protocol (NTP), Microsoft SQL Server (MSSQL), and Domain Name System (DNS). Figure 5 shows the Asimily portal view of the assets identified in the PV plant, and Figure 6 shows the specific details identified for theELM MicroGrid (ELM) battery energy storage system (BESS).

Table 2 summarizes which attributes Asimily identified or did not identify, including each device's hostname, MAC address, MAC vendor, operating system (OS), and OS version.

[5]Passive sampling results depend on the system configuration and which sources of traffic are visible to the solution. Additional sampling points and traffic flows could have provided deeper visibility.

[6]Persistent internal IP addresses are redacted throughout the report. This pertains to just one subnet, which is always redacted as XX.XX.XX.0/24, and the last octet is left to identify the specific device.

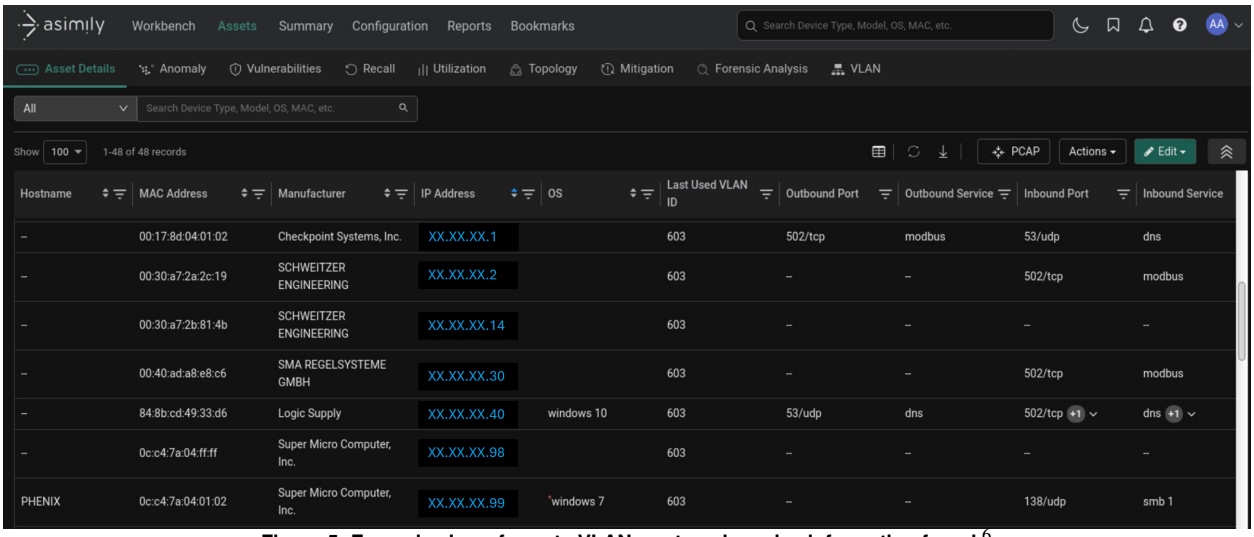


Figure 5. Example view of assets VLAN, port, and service information found.[6]

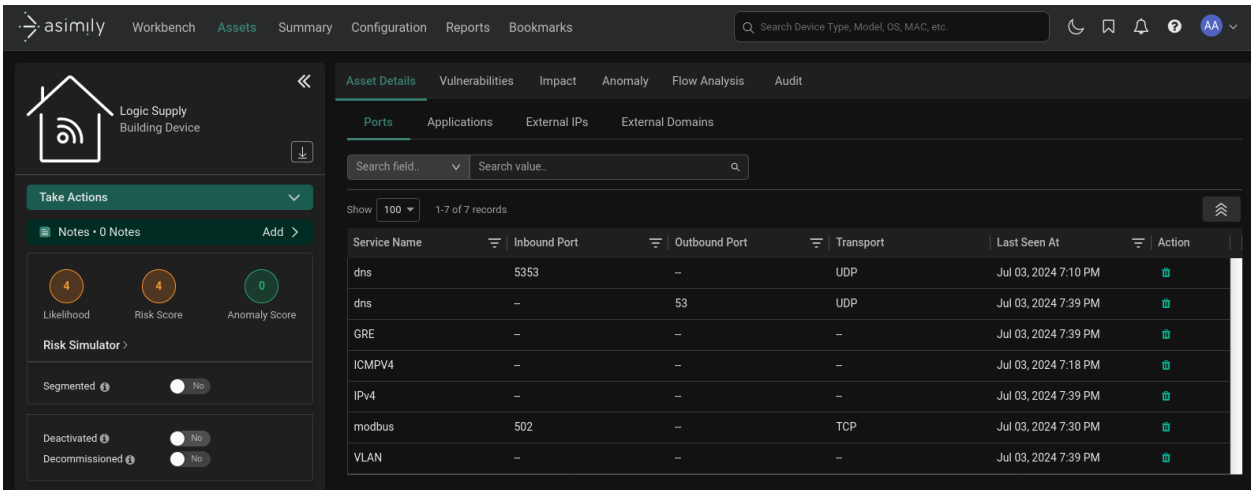


Figure 6. Example view of services found for a specific asset

Table 2. Scenario 1 data richness

Device	Hostname	IP Address	MAC Address	MAC Vendor	OS	OS Version
Control center (7 devices)						
cc firewall			✓	✓		
Asimily edge processor 1		✓	✓	✓		
†cc admin vm	✓	✓	✓	✓	X	
SCADA platform		✓	✓	✓		
Engineering workstation	✓	✓	✓	✓	✓	
†NTP server			✓	✓		
†Syslog server		✓	✓	✓		
Substation OT gateway (3 devices)						
†Sub-ot-gateway admin vm	✓	✓	✓	✓	X	
Asimily edge processor 2		✓	✓	✓		
SEL 3622		✓	✓	✓		
Substation OT (11 devices)						
Sub firewall			✓	✓		
Asimily edge processor 3		✓	✓	✓		
***OT switch**						
†Sub-ot admin vm	✓	✓	✓	✓	X	
SEL 3530 RTAC		✓	✓	✓		
†SEL 411L		✓	✓	✓		
SEL 751		✓	✓	✓		
SEL 351A		✓	✓	✓		
***SEL 311C**						
***SEL 321**						
***SEL 501**						
Substation IT (4 devices)						
***IT switch**						
Asimily edge processor 4		✓	✓	✓		
***†Sub-it admin vm**						
†Workstation		✓	✓	✓	✓	
PV plant (8 devices)						
PV firewall		✓	✓	✓		
Asimily edge processor 5		✓	✓	✓		
†PV admin vm	✓	✓	✓	✓	X	
SEL 3530 RTAC		✓	✓	✓		
***SEL 751 to SMA**						
SEL 751 to ELM		✓	✓	✓		
SMA Sunny Highpower		✓	✓	✓		
ELM BESS		✓	✓	✓	✓	
Total (of 33 devices)	5	23	26	26	3	0

* Device not identified
† No application traffic
✓ Attribute correctly identified
X Attribute incorrectly reported

The Asimily solution also extracted inferences about vulnerabilities and risks from the network traffic that it parsed. Two example vulnerabilities identified during testing included the passage of credentials over an insecure protocol—Hypertext Transfer Protocol (HTTP)—shown in Figure 7, and the presence of a device manufactured by a company that is banned for sale in the United States, shown in Figure 8. The Asimily solution also identified all Common Vulnerability and Exposure (CVE)s that a device *might* be subject to, based on its OS and services, shown in Figure 9.

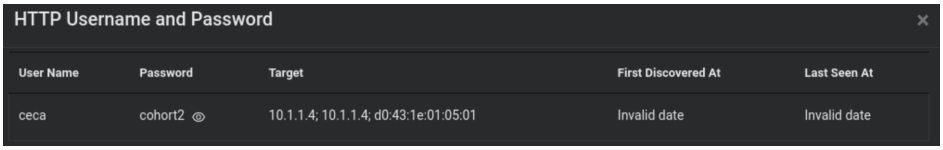


Figure 7. Asimily identifies a log-in over insecure protocol and captures credentials.

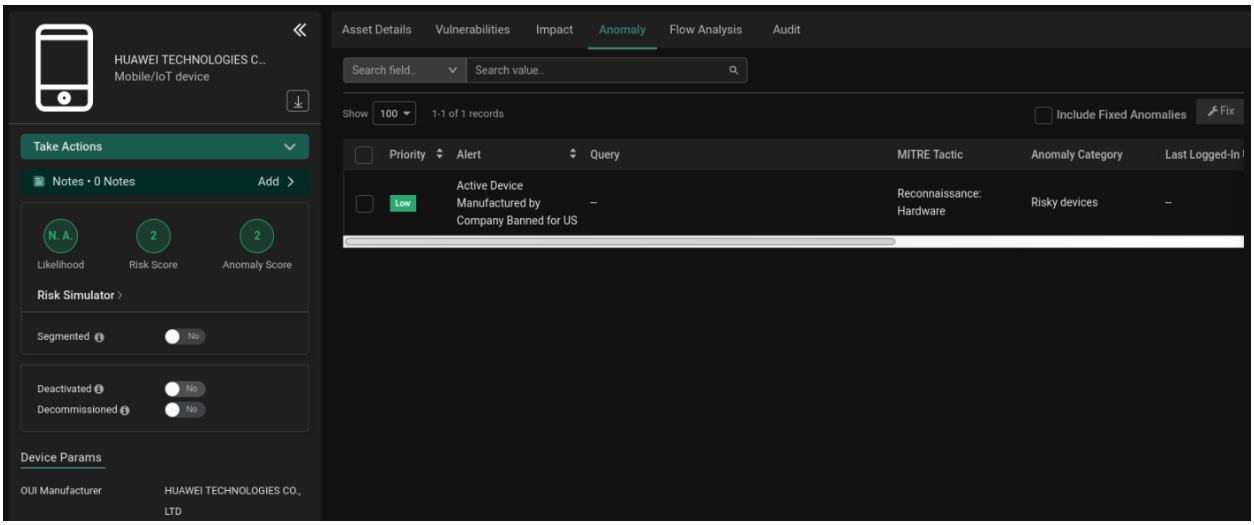


Figure 8. Asimily identifies a substation router manufactured by a sanctioned company.

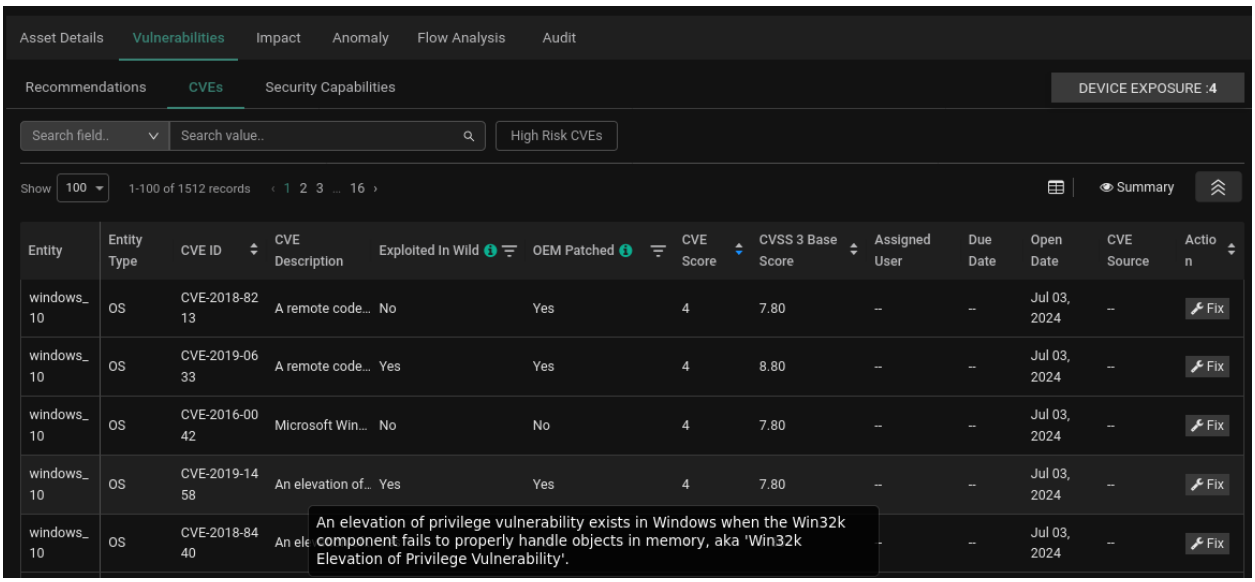


Figure 9. Example of CVEs identified for a device

Additional Network Traffic

Additional network traffic evaluates how much traffic the solution adds to the network above baseline levels. This is examined in two ways: (1) How much traffic does the solution add to each local subnet? (2) How much traffic does the solution add to the backhaul network that carries communications from each location to the control center DMZ where the solution's central components are located?

The Asimily solution was configured with one edge processor in each subnet. These edge processors added negligible traffic to their local subnets, as shown in Table 3. This is expected with passive sampling solutions.

Table 3. Asimily edge processor traffic on subnets in Scenario 1

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	5 kB	0.5 kB
Substation OT gateway	3	1.4 kB	0.3 kB
Substation OT	11	1.6 kB	0.3 kB
Substation IT	4	1.8 kB	0 kB
PV plant	8	1.8 kB	0 kB

The edge processor in each location sends data back to the Asimily main server in the control center DMZ. The edge processors communicating with the control center added approximately 33% of additional network traffic above baseline levels, as shown in Table 4.

Table 4. Asimily edge processor traffic to main server in Scenario 1

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	16,959 kB	4,081 kB
Substation OT gateway	3	4,979 kB	651 kB
Substation OT	11	8,713 kB	1,047 kB
Substation IT	4	5,140 kB	512 kB
PV plant	8	12,695 kB	503 kB

3.2 Scenario 2: Change Discovery

The objective of this scenario was to examine how a solution tracks changes to an environment. Scenario 2 evaluated change detection and alerts. Scenario 2 was a follow-on test to Scenario 1, and it started with the solution in an "onboarded" state, like it was at the end of Scenario 1. This scenario was run in the PV and substation environment with several changes:

- **New connections:**
  - **Rogue device:** A RaspberryPi was added to the substation OT network, simulating an unauthorized user or rogue device.
  - **Misconfigured device:** The substation IT printer was connected to the network via Ethernet in violation of security policy.
- **Changes to existing devices:**
  - The IP address of the engineering workstation in the control center was changed to `10.1.1.10`, while all other attributes were held constant.
  - The MAC address of the syslog server in the control center was changed to `10:c5:95:ff:04:ff`, while all other attributes were held constant.

These changes are visually depicted in Figure 10.

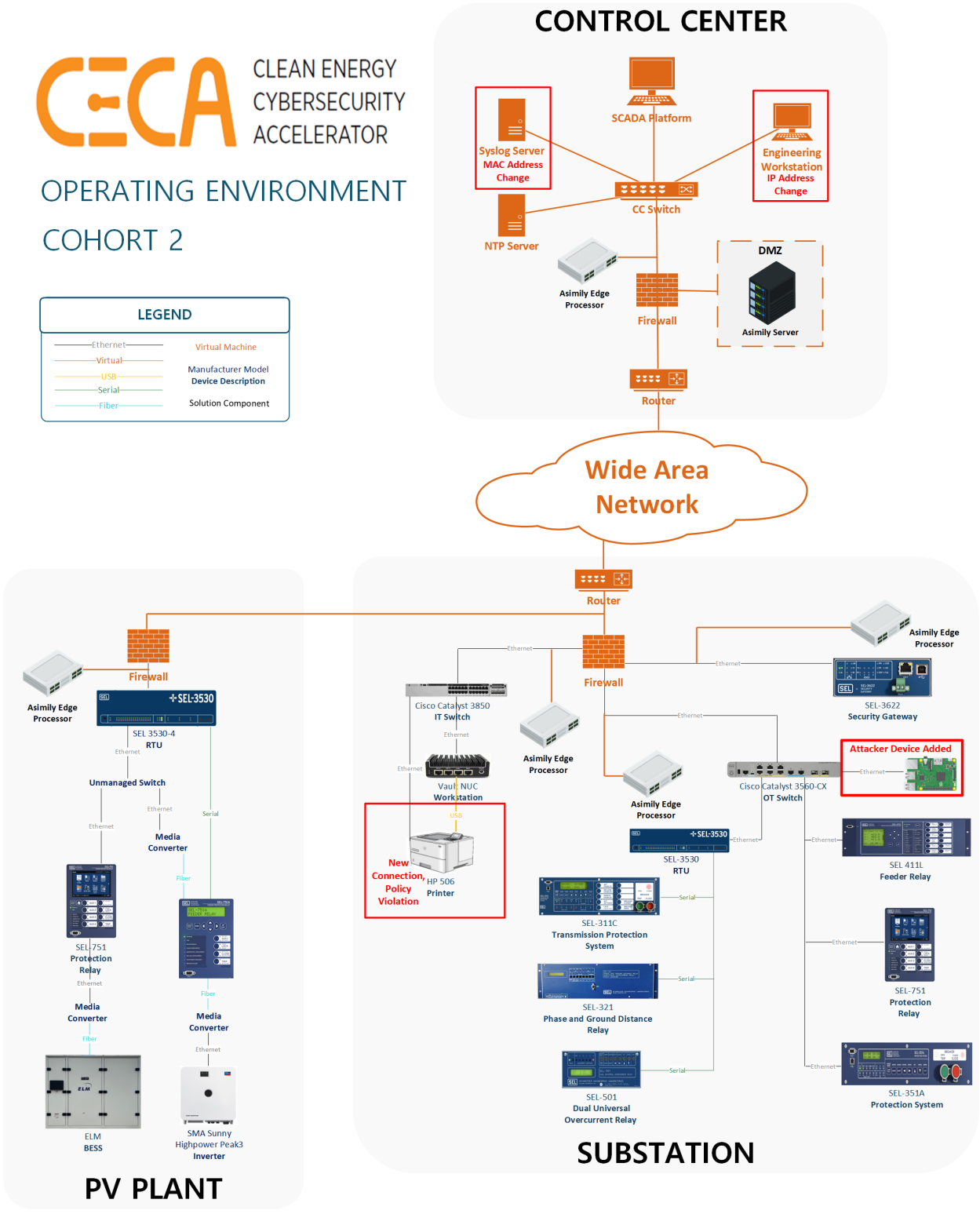


Figure 10. Scenario 2 Asimily AOE

3.2.1 Results

Change Detection

Change detection evaluates how well the solution tracks changes to assets over time. The Asimily solution successfully identified each of the four changes introduced into the environment. It identified the new attacker device and the newly connected printer. It also tracked the changed MAC address, as shown in Figure 11, and it updated the device entry for the engineering workstation with the new IP, as shown in Figure 12. The red dots indicate that a recent change has occurred in that field, so it serves as a form of a visual alert. The user does not need to filter those changes specifically, however, they are able to do so if needed. Asimily also provides other views which focus on the anomalies specifically if that is the focus of the user.

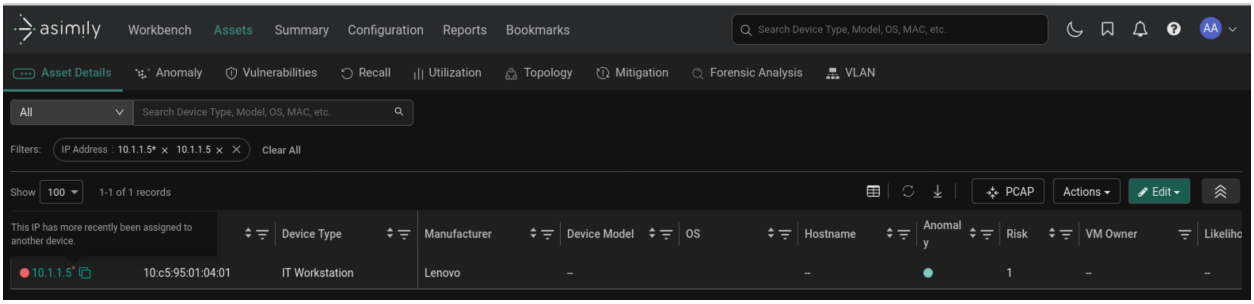


Figure 11. Asimily inventory showing changed MAC

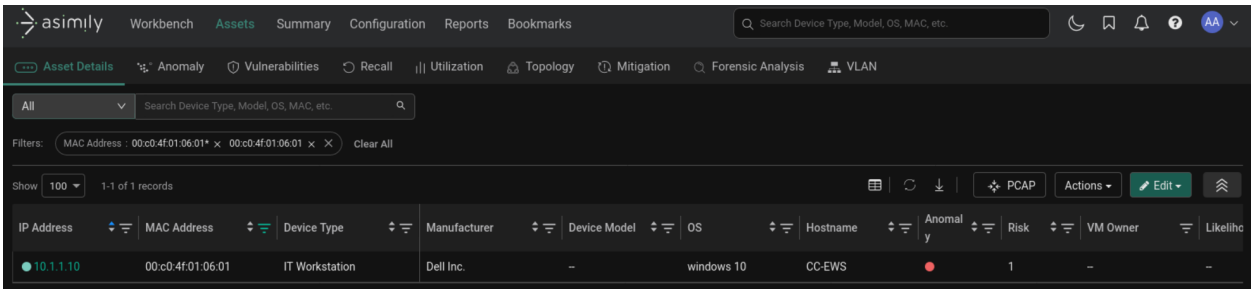


Figure 12. Asimily inventory showing changed IP

Alerts

Alerts evaluates how the solution notifies users of new or unexpected devices on the network. CECA configured the Asimily solution with a custom anomaly rule for any new devices detected after the start of the test. In each test, the Asimily solution created "high-risk" anomalies for the newly discovered device. Figure 13 shows this alert.

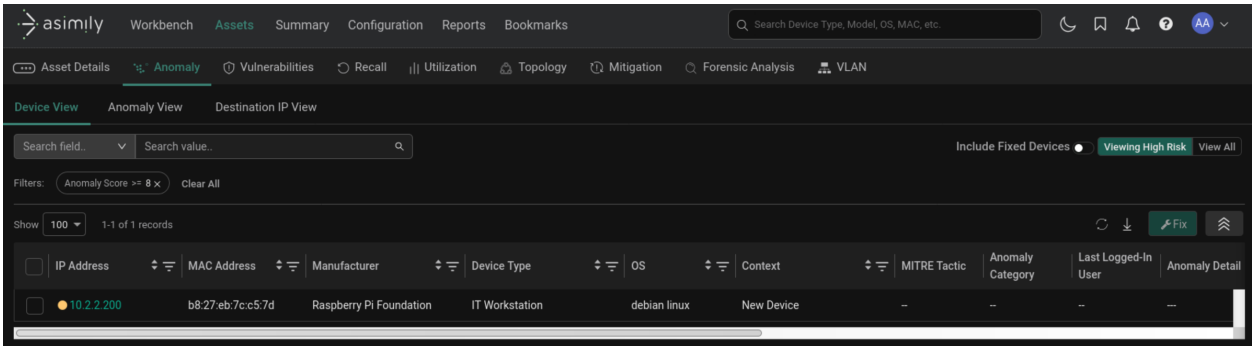


Figure 13. Example of high-risk anomaly for the newly discovered device

Asimily monitored each device's communication flows and presented that data to users in a "Flow Analysis" view. An example is shown in Figure 14, which highlights traffic to external IPs.

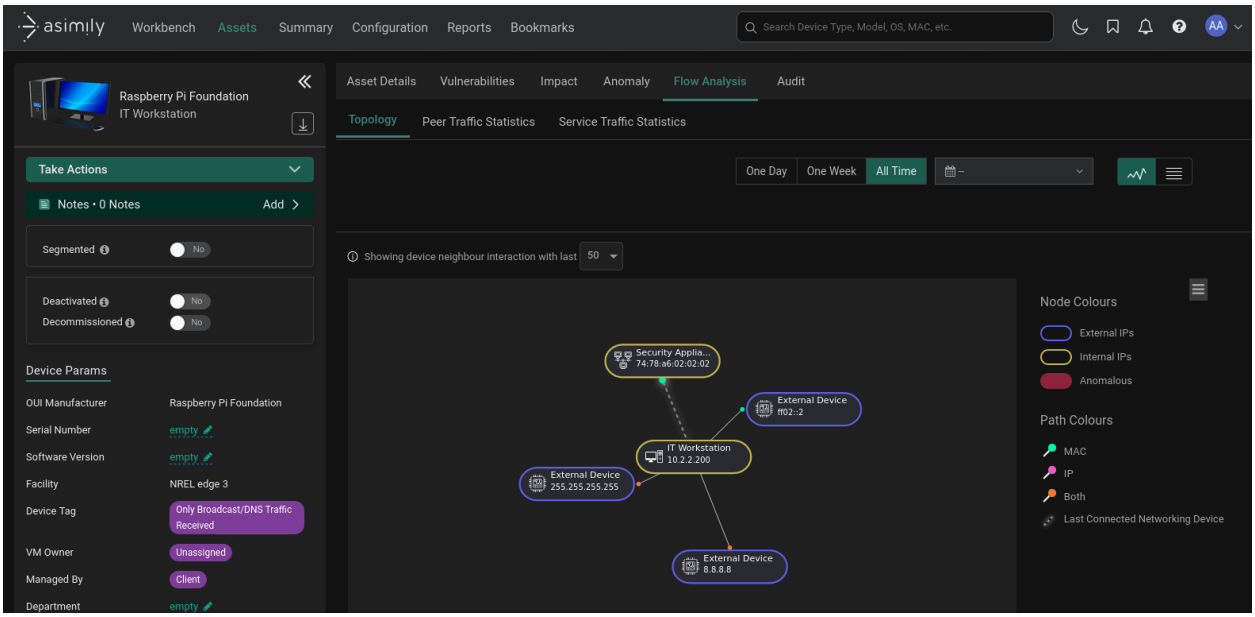


Figure 14. Asset view of a newly added device showing traffic flow analysis and including external IPs

3.2.2 Measurements Not Used for Evaluation

The following criteria were not part of the objectives for Scenario 2, but they were measured during the tests. They provide additional data points for Scenario 1.

Data Richness

Data richness evaluates the amount of detail the solution collects about each device. Scenario 2 data richness only differs from Scenario 1 with the identification of the two devices added in this scenario.

Table 5. Scenario 2 data richness

Device	Hostname	IP Address	MAC Address	MAC Vendor	OS	OS Version
Control center (7 devices)						
cc firewall			✓	✓		
Asimily edge processor 1		✓	✓	✓		
†**cc admin vm**	✓	✓	✓	✓	X	
SCADA platform		✓	✓	✓		
Engineering workstation	✓	✓	✓	✓	✓	
†**NTP server**			✓	✓		
†**Syslog server**		✓	✓	✓		
Substation OT gateway (3 devices)						
†**Sub-ot-gateway admin vm**	✓	✓	✓	✓	X	
Asimily edge processor 2		✓	✓	✓		
SEL 3622		✓	✓	✓		
Substation OT (12 devices)						
Sub firewall			✓	✓		
Asimily edge processor3		✓	✓	✓		
*__OT switch__						
†**Sub-ot admin vm**	✓	✓	✓	✓	X	
†**Attacker RPi**	✓	✓	✓	✓	✓	
SEL 3530 RTAC		✓	✓	✓		
†**SEL 411L**		✓	✓	✓		
SEL 751		✓	✓	✓		
SEL 351A		✓	✓	✓		
*__SEL 311C__						
*__SEL 321__						
*__SEL 501__						
Substation IT (5 devices)						
*__IT switch__						
Asimily edge processor 4		✓	✓	✓		
†**Sub-it admin vm**						
†**Workstation**		✓	✓	✓	✓	
†**printer**		✓	✓	✓		
PV plant (8 devices)						
PV firewall		✓	✓	✓		
Asimily edge processor 5		✓	✓	✓		
†**PV admin vm**	✓	✓	✓	✓	X	
SEL 3530 RTAC		✓	✓	✓		
*__SEL 751 to SMA__						
SEL 751 to ELM		✓	✓	✓		
SMA Sunny Highpower		✓	✓	✓		
ELM BESS		✓	✓	✓	✓	
Total (of 35 devices)	6	25	28	28	4	0

* Device not identified
† No application traffic
✓ Attribute correctly identified
X Attribute incorrectly reported

Additional Network Traffic

Additional network traffic evaluates how much traffic the solution adds to the network above baseline levels. This is examined in two ways: (1) How much traffic does the solution add to each local subnet? (2) How much traffic does

the solution add to the backhaul network that carries communications from each location to the control center DMZ where the solution's central components are located?

The Asimily solution was configured with one edge processor in each subnet. These edge processors added negligible traffic to their local subnets, as shown in Table 6. This is expected with passive sampling solutions.

Table 6. Asimily edge processor traffic on subnets in Scenario 2

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	5 kB	.3 kB
Substation OT gateway	3	0.8 kB	0.3 kB
Substation OT	12	1.6 kB	0.9 kB
Substation IT	5	1.2 kB	0.6 kB
PV plant	8	1.2 kB	1 kB

The edge processor in each location sends data back to the Asimily main server in the control center DMZ. The edge processors communicating with the control center added approximately 33% additional network traffic above baseline levels, as shown in Table 7.

Table 7. Asimily edge processor traffic to main server in Scenario 2

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	22,289 kB	7,388 kB
Substation OT gateway	3	5,605 kB	29 kB
Substation OT	12	11.516 kB	34 kB
Substation IT	5	5,719 kB	61 kB
PV plant	8	13,264 kB	32 kB

3.3 Scenario 3: Alternate Discovery

The objective of this scenario was to examine how a solution performs using alternative methods for asset discovery. Scenario 3 evaluated inventory accuracy, data richness, additional network traffic, and disruption of operations. This scenario was run in the PV and substation environment.

Asimily's alternate discovery method combines passive sampling with periodic, targeted active scanning. Each edge processor uses information collected from passive sampling to build a picture of which devices exist in the environment and which ports and protocols they are using to communicate. Based on this information, the edge processors send targeted queries to collect additional information about the hosts and protocols in the network that have already been identified.

CECA configured the Asimily solution to enable targeted active scanning by setting `ENABLE_ACTIVE_SCANNER=1` in each Asimily edge processor configuration file. CECA also updated Asimily's main server cronjob settings to enable periodic active scanning. Asimily's default configuration is to perform targeted active scans every 10 minutes, so CECA maintained the 30-minute sampling period for the Scenario 3 tests, guaranteeing at least two targeted active scans. CECA's configuration of the Asimily solution used `nmap` to perform targeted active scanning.

3.3.1 Results

Inventory Accuracy

Inventory accuracy evaluates how many assets the solution successfully identifies. The Asimily solution successfully identified 27 of 33 assets in the environment. The only change in inventory accuracy in Scenario 3, compared to Scenario 1's exclusively passive sampling, is that Asimily's solution successfully identified the SEL 751 connected to the SMA inverter located in the PV plant. The solution achieved this increased visibility because Scenario 3's disruption to operations criteria used an Internet Control Message Protocol (ICMP) polling apparatus, which generated traffic to the device.

Data Richness

Data richness evaluates the amount of detail the solution collects about each device. Compared to Scenario 1's exclusively passive sampling, the data gathered during Scenario 3 were the same across all fields evaluated in Table 8, except for the addition of the identification of the SEL 751 connected to the System, Mess and Anlagentechnik Solar Technology AG (SMA) inverter; however, targeted active scanning identified many more services and open ports.

Table 8. Scenario 3 data richness

Device	Hostname	IP Address	MAC Address	MAC Vendor	OS	OS Version
Control center (7 devices)						
cc firewall			✓	✓		
Asimily edge processor 1		✓	✓	✓		
†cc admin vm	✓	✓	✓	✓	X	
SCADA platform		✓	✓	✓		
Engineering workstation	✓	✓	✓	✓	✓	
†NTP server			✓	✓		
†Syslog server		✓	✓	✓		
Substation OT gateway (3 devices)						
†Sub-ot-gateway admin vm	✓	✓	✓	✓	X	
Asimily edge processor 2		✓	✓	✓		
SEL 3622		✓	✓	✓		
Substation OT (11 devices)						
Sub firewall			✓	✓		
Asimily edge processor 3		✓	✓	✓		
***OT switch**						
†Sub-ot admin vm	✓	✓	✓	✓	X	
SEL 3530 RTAC		✓	✓	✓		
†SEL 411L		✓	✓	✓		
SEL 751		✓	✓	✓		
SEL 351A		✓	✓	✓		
***SEL 311C**						
***SEL 321**						
***SEL 501**						
Substation IT (4 devices)						
***IT switch**						
Asimily edge processor 4		✓	✓	✓		
***†Sub-it admin vm**						
†Workstation		✓	✓	✓	✓	
PV plant 8 devices)						
PV firewall		✓	✓	✓		
Asimily edge processor 5		✓	✓	✓		
†PV admin vm	✓	✓	✓	✓	X	
SEL 3530 RTAC		✓	✓	✓		
SEL 751 to SMA		✓	✓	✓		
SEL 751 to ELM		✓	✓	✓		
SMA Sunny Highpower		✓	✓	✓		
ELM BESS		✓	✓	✓	✓	
Total (of 33 devices)	5	24	27	27	3	0

* Device not identified
† No application traffic
✓Attribute correctly identified
X Attribute incorrectly reported

Figure 15 shows how active scanning was able to identify HTTP and Hypertext Transfer Protocol Secure (HTTPS) services on ports `80` and `443`, respectively, including specific Secure Sockets Layer (SSL) and Transport Layer Security (TLS) versions. Both ports were open but remained unused during testing.

Service Name	Inbound Port	Outbound Port	Transport
GRE	--	--	--
ICMPV4	--	--	--
IPv4	--	--	--
modbus	502	--	TCP
VLAN	--	--	--

Service Name	Inbound Port	Outbound Port	Transport
GRE	--	--	--
http	80	--	TCP
ICMPV4	--	--	--
IPv4	--	--	--
modbus	502	--	TCP
ssl_tls	443	--	TCP
ssl_tls SSL 3	443	--	TCP
ssl_tls TLS 1.2	443	--	TCP
VLAN	--	--	--

Figure 15. Comparison of ports and services found with passive sampling (top) vs. targeted active scanning (bottom) for the SMA inverter

Additional Network Traffic

Additional network traffic evaluates how much traffic the solution adds to the network above baseline levels. This is examined in two ways: (1) How much traffic does the solution add to each local subnet? (2) How much traffic does the solution add to the backhaul network that carries communications from each location to the control center DMZ where the solution's central components are located?

The Asimily solution was configured with one edge processor in each subnet. In scenario 3, these edge processors added noticeable amounts of traffic to their respective subnets, as shown in Table 9. This is expected with passive sampling solutions. This change from previous scenarios is expected because the edge processors are conducting targeted active scanning to collect information from local devices.

Table 9. Asimily edge processor traffic on subnets in Scenario 3

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	3,774 kB	593 kB
Substation OT gateway	3	2,499 kB	389 kB
Substation OT	11	15,859 kB	2,368 kB
Substation IT	4	8,345 kB	1.123 kB
PV plant	8	7,219 kB	1,199 kB

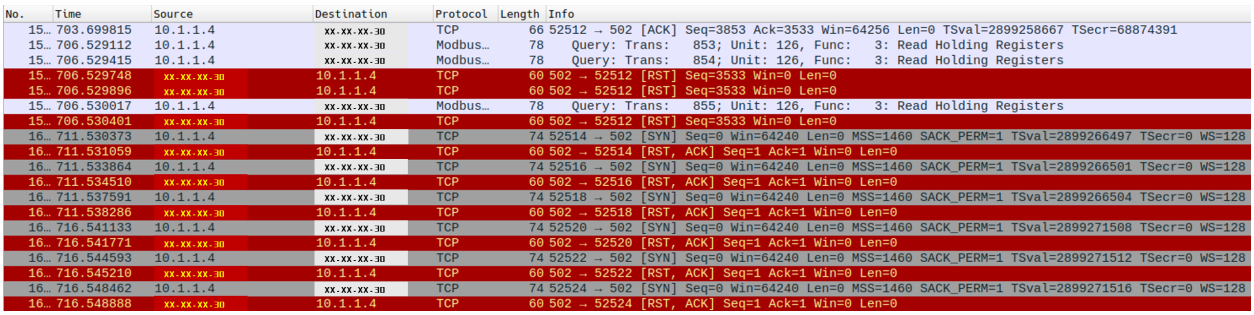
The edge processor in each location sends data back to the Asimily main server in the control center DMZ. The edge processors communicating with the control center added approximately 33% additional network traffic above baseline levels, as shown in Table 10.

Table 10. Asimily edge processor traffic to main server in Scenario 3

Subnet	Number of Hosts	Average	Standard Deviation
Control center	7	53,808 kB	7,748 kB
Substation OT gateway	3	55,863 kB	7,142 kB
Substation OT	11	132,953 kB	13,971 kB
Substation IT	4	78,240 kB	8,954 kB
PV plant	8	99,896 kB	12,218 kB

Disruption of Operations

CECA observed intermittent loss in connectivity between the supervisory control and data acquisition (SCADA) platform and the SMA inverter across each of the three runs for Scenario 3. Upon inspection, CECA found that these temporary losses in availability coincided with periods during which targeted active scanning was occurring. A Wireshark screenshot in Figure 16 shows the SCADA platform failing to establish a TCP handshake with the inverter on port 502.



No.	Time	Source	Destination	Protocol	Length	Info
15…	703.699815	10.1.1.4	xx.xx.xx.30	TCP	66	52512 → 502 [ACK] Seq=3853 Ack=3533 Win=64256 Len=0 TSval=2899258667 TSecr=68874391
15…	706.529112	10.1.1.4	xx.xx.xx.30	Modbus…	78	Query: Trans: 853; Unit: 126, Func: 3: Read Holding Registers
15…	706.529415	10.1.1.4	xx.xx.xx.30	Modbus…	78	Query: Trans: 854; Unit: 126, Func: 3: Read Holding Registers
15…	706.529748	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52512 [RST] Seq=3533 Win=0 Len=0
15…	706.529896	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52512 [RST] Seq=3533 Win=0 Len=0
15…	706.530017	10.1.1.4	xx.xx.xx.30	Modbus…	78	Query: Trans: 855; Unit: 126, Func: 3: Read Holding Registers
15…	706.530401	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52512 [RST] Seq=3533 Win=0 Len=0
16…	711.530373	10.1.1.4	xx.xx.xx.30	TCP	74	52514 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899266497 TSecr=0 WS=128
16…	711.531059	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52514 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16…	711.533864	10.1.1.4	xx.xx.xx.30	TCP	74	52516 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899266501 TSecr=0 WS=128
16…	711.534510	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52516 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16…	711.537591	10.1.1.4	xx.xx.xx.30	TCP	74	52518 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899266504 TSecr=0 WS=128
16…	711.538286	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52518 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16…	716.541133	10.1.1.4	xx.xx.xx.30	TCP	74	52520 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899271508 TSecr=0 WS=128
16…	716.541771	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52520 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16…	716.544593	10.1.1.4	xx.xx.xx.30	TCP	74	52522 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899271512 TSecr=0 WS=128
16…	716.545210	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52522 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16…	716.548462	10.1.1.4	xx.xx.xx.30	TCP	74	52524 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2899271516 TSecr=0 WS=128
16…	716.548888	xx.xx.xx.30	10.1.1.4	TCP	60	502 → 52524 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Figure 16. Wireshark screenshot showing the SCADA platform unable to perform a TCP handshake with the SMA inverter

The Asimily solution was configured to perform targeted active scanning every 10 minutes (i.e., starting at minutes 10, 20, 30, etc., after the hour). At the start of each scan, the Asimily edge processor conducted a `SYN` sweep to find open TCP ports on the inverter. Figure 17 shows the beginning of this traffic.

No.	Time	Source	Destination	Protocol	Length	Info
1388…	652.617777	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 554 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1388…	652.619985	xx-xx-xx-30	xx-xx-xx-98	TCP	60	554 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1388…	652.627446	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 3389 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1388…	652.628127	xx-xx-xx-30	xx-xx-xx-98	TCP	60	3389 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1388…	652.628521	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 3306 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1388…	652.629261	xx-xx-xx-30	xx-xx-xx-98	TCP	60	3306 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1389…	652.643621	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1389…	652.644460	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 5900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1389…	652.644819	xx-xx-xx-30	xx-xx-xx-98	TCP	60	135 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1389…	652.645415	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 8080 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1389…	652.645714	xx-xx-xx-30	xx-xx-xx-98	TCP	60	5900 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1389…	652.646126	xx-xx-xx-30	xx-xx-xx-98	TCP	60	8080 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1389…	652.646417	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 1025 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1389…	652.647423	xx-xx-xx-30	xx-xx-xx-98	TCP	60	1025 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1389…	652.664964	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1390…	652.665934	xx-xx-xx-98	xx-xx-xx-30	TCP	58	65082 → 23 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1390…	652.665953	xx-xx-xx-30	xx-xx-xx-98	TCP	60	80 → 65082 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
1390…	652.666039	xx-xx-xx-98	xx-xx-xx-30	TCP	54	65082 → 80 [RST] Seq=1 Win=0 Len=0
1390…	652.666727	xx-xx-xx-30	xx-xx-xx-98	TCP	60	23 → 65082 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Figure 17. Wireshark screenshot showing the Asimily SYN sweep

CECA verified that each period where the SCADA platform was unable to poll the SMA inverter coincided with Asimily's targeted active scanning. In each instance, the inverter failed to perform a TCP handshake with the SCADA platform, replying to each attempt with `RST` packets. This behavior continued for exactly 1 minute before the SCADA platform was able to establish normal communications with the inverter. During each subsequent 10-minute interval when the Asimily edge processor began its targeted active scanning, the inverter would again become unresponsive for 1 minute.

Although the Asimily solution's targeted active scanning disrupted the SMA inverter in the Scenario 3 testing, the interruption was minimal, and the asset recovered relatively quickly. The inverter was available throughout the Scenario 3 tests, responding to ICMP pings and sending `RST` responses to the SCADA platform's attempt to establish a TCP connection. The SMA maintained 90% availability during the targeted active scanning (i.e., unresponsive for exactly 1 minute out of every 10). Further, the SMA was the only device impacted. All other devices and SCADA protocols remained available throughout the targeted active scanning. Targeted active scanning is a new capability in Asimily's solution, meaning future refinements could address the availability of assets during active scanning.

3.4 Scenario 4: Scale Discovery

The objective of this scenario was to examine how a solution performs at scale. Scenario 4 evaluated inventory accuracy, additional network traffic, and alerts. The previous scenarios were all run in the same PV and substation environment with several tens of devices. To stress the solution and test how it performs at scale, CECA created the AMI environment with 2,014 AMI devices in a single "flat" subnet (`/21` in classless inter-domain routing (CIDR) notation). In addition to these AMI devices, the environment featured a VM to host an Asimily edge processor, an administrative VM, and a router, for a total of 2,017 devices. Scenario 4 magnifies both the time that the solution takes to identify a single device and the amount of additional network traffic that the solution adds to the infrastructure when identifying a single asset. In addition, Scenario 4 provides an opportunity to test a solution's ability to identify a new device in a much larger environment.

Scenario 4 consisted of two consecutive phases. First, the solution was activated to identify all the existing assets in the environment. Second, a single additional device was added to the network, and the solution was again activated to identify all the assets in the environment, including the new device.

The AMI environment for the Scenario 4 testing of Asimily was configured with background traffic between an administrative VM and each of the AMI devices.

3.4.1 Results

Inventory Accuracy

Inventory accuracy evaluates how many assets the solution successfully identifies. The Asimily solution identified all devices in both the first phase (2,017 devices) and the second phase (2,018 devices) of the test.

Alert

Alerts evaluates how the solution notifies users of new or unexpected devices on the network. CECA configured the Asimily solution with a custom anomaly rule for any new devices detected after the end of Phase 1. The Asimily solution generated alerts within 5 minutes of the new machine being turned on during each test run.

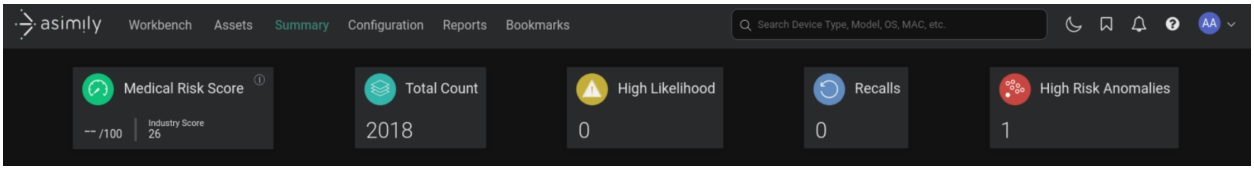


Figure 18. Asimily high-risk anomaly at the end of Scenario 4

Additional Network Traffic

Additional network traffic evaluates how much traffic the solution adds to the network above baseline levels. This is examined in two ways: (1) How much traffic does the solution add to each local subnet? (2) How much traffic does the solution add to the backhaul network that carries communications from each location to the control center DMZ where the solution's central components are located?

The Asimily solution was configured with one edge processor in the AMI subnet. This edge processor added negligible traffic to its local subnets, averaging 3 kB across both phases. This is expected with passive sampling solutions.

The edge processor sent data back to the Asimily main server. The edge processors communicating with the main server added approximately 33% additional network traffic above baseline levels, averaging 99,166 kB, with a standard deviation of 18,164 kB across both phases.

3.4.2 Additional Notes

Variable Processing Time

The initial test phases produced variable times to process and settle to a consistent baseline in the number of devices. The Asimily solution generally reported the correct number of devices (2,017) within the first 30 minutes of the initial phase of this scenario; however, the reported number of devices occasionally oscillated to either greater than 2,017 or fewer than 2,017 before eventually settling on 2,017. In all test runs, these oscillations approached and settled on the correct number of devices within 60 minutes.

4 Conclusion

Utility ICS networks can be vast, geographically dispersed systems that comprise a heterogeneous set of devices and protocols. These characteristics compound the ability of asset owners to accurately appraise which devices are connected to their network, which risks they face, and how those risks emerge and evolve within their environment. Asimily represents one product in a class of solutions designed to help organizations enumerate their assets and understand potential risks while maintaining normal business operations.

The CECA evaluations in Cohort 2 tested the Asimily solution across a range of scenarios. In these tests, the Asimily solution consistently and quickly identified all assets for which it was able to sample traffic. The visibility into network assets provided by the Asimily solution could help operators evaluate risks that they might otherwise miss. Across tests, CECA observed that the Asimily edge processor increased the amount of traffic traversing each firewall and traveling back to the control center by approximately 1–2 MB per active host in the edge processor's subnet.

All tests of Asimily's solution found that it was able to quickly identify assets in the environment. Tests were run for 30 minutes in the smaller PV substation environments, but Asimily only needed between 1 and 5 minutes to complete its identification of assets in the environment, after which the total number of identified assets remained consistent until the end of the test. In the larger AMI environment, Asimily identified more than 1,000 devices within the first 10 minutes, but it needed 30 to 60 minutes to define a baseline number of assets in the environment. In this AMI environment, once the baseline was established, the Asimily solution performed effectively and was able to identify new devices quickly and reliably.

The CECA testing demonstrated how Asimily's hybrid methodology of passive and targeted active scanning enhances visibility into networked devices without causing substantial impacts to system availability. The CECA tests also demonstrated that Asimily's hybrid method of passive sampling with targeted active scanning developed a rich understanding of protocols and services;however, the solution temporarily affected the availability of information to the SCADA platform. CECA tested Asimily's solution against varied ICS protocols and devices to validate the conclusions to the greatest degree possible, and these results represent the analysis conducted during the four scenarios for Cohort 2.

CECA's testing revealed key areas for improvement for Asimily's solution. One area was in expanding capabilities related to asset identification. Specifically, although Asimily supports some common ICS protocols, such as Modbus and DNP3, additional protocol and device support might be needed to identify the diverse range of assets in many systems. Another area for improvement identified in the testing was in the active scanning methodology and the potential for disruption of availability of some resources when performing active scanning.

Challenges still exist for the broader class of asset identification technologies in the ICS space, including the visibility of assets connected via legacy media like serial, the identification of assets that are not IP addressable, and the visibility into assets connected behind a remote terminal unit (RTU) that do not forward traffic to subordinate devices. Solving each problem will likely require a combination of various data collection techniques, including vendor-specific methods for credentialed identification, manual operator actions, ingestion of procurement documents, etc.

Cybersecurity is a complex and shifting field full of unique challenges. Threats, risks, architectures, and technologies will continue to evolve as the energy sector undergoes significant transformations. Innovations of solutions should be enabled to evolve as well. There will always be widespread challenges in industry that solution providers are aiming to solve. Using solutions such as those offered by Asimily to identify control system assets and to monitor changes in that equipment is expected to improve the security of the industry as a whole.

References

Asimily. 2024a. “About Us.” https://asimily.com/product/.

Asimily. 2024b. “Reduce Vulnerabilities 10x Faster with Half the Resources.” https://asimily.com/product/.

Modbus Organization. 2006a. “MODBUS Messaging on TCP/IP Implementation Guidea,” October. https://modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf.

Modbus Organization. 2006b. “MODBUS over Serial Line Specification and Implementation Guide,” December. https://modbus.org/docs/Modbus_over_serial_line_V1_02.pdf.

Modbus Organization. 2012. “MODBUS Application Protocol Specification,” April. https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf.

National Institute of Standards and Technology. 2020. “Energy Sector Asset Management for electric utilities, oil ...” https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-23.pdf.

National Renewable Energy Laboratory. 2024. “ARIES Cyber Range.” https://www.nrel.gov/security-resilience/cyber-range.html.

OpenJS Foundation. 2024. “Node-RED.” https://nodered.org/.

Patria Security, LLC. 2023. “Operational Technology (OT) Simulator.” https://ot-sim.patsec.dev/.

Patria Security, LLC. 2024a. “phenix documentation.” https://phenix.sceptre.dev/latest/scorch/.

Patria Security, LLC. 2024b. “phenix documentation.” https://phenix.sceptre.dev/latest/state-of-health/.

Patria Security, LLC. 2024c. “phenix documentation.” https://phenix.sceptre.dev/latest/apps/#vrouter-app.

Sandia National Laboratories. 2023. “minimega.” https://www.sandia.gov/minimega/.

Sandia National Laboratories. 2024a. “minimega github.” https://github.com/sandia-minimega/minimega.

Sandia National Laboratories. 2024b. “phenix.” https://phenix.sceptre.dev/latest/.

Sandia National Laboratories. 2024c. “sceptre phenix github.” https://github.com/sandialabs/sceptre-phenix.

Wallace, A., A. Liao, D. Rager, A. Hasandka, A. Sahu, N. Ryan, S. Drake, et al. 2024. *Cloud Zero Phase 2 Technical Report.* Technical report. Under submission. NREL. https://www.nrel.gov/docs/fy24osti/xxxx.pdf.

Appendix A. Baseline Operating Environment

A.1 Architecture Overview

The Cohort 2 solutions were tested in solution-specific operating environments built within a common baseline operating environment (BOE). The BOE describes the environment prior to the inclusion and configuration of the solution under test. The BOE included a control center, substation, and utility-owned PV plant. The Cohort 2 BOE is represented as a combination of VMs and hardware devices. The control center comprised only VMs, whereas the substation and PV plant consisted of hardware components and virtual firewalls.

The BOE was deployed through NREL's ARIES Cyber Range, a cyber-physical modeling and simulation platform that supports both virtual and physical deployments of variable-scale environments (NREL 2024). The ARIES Cyber Range leverages multiple open-source software packages to facilitate the design and deployment of experiments, networking, and VMs. Minimega is a VM manager that oversees the creation and startup of kernel-based virtual machine (KVM)s and software defined networking (SDN) used within the emulated environment (SNL 2023). Phenix sits above minimega in the software stack and orchestrates the organization and deployment of experiments and scenario executions from structured markup configuration files (SNL 2024b). Details about these tools can be found in Appendix B.

Through the deployment of an experiment on the ARIES Cyber Range, the requisite configurations and networking were set up to allow for repeatable evaluations and analyses of the generated data. The experiment BOE was designed to emulate a simple distribution system topology used by a utility or municipality to deliver power or grid services. To orient the BOE toward the cohort theme of device discovery, different asset types were used, and configurations were diversified to provide a clear understanding of the capabilities of each solution.

Several SEL power hardware assets were deployed, including relays, protection systems, communication devices, and control equipment (e.g., Real-Time Automation Controller (RTAC)). The power assets connected were an SMA inverter and an ELM BESS. The substation also included IT elements, such as a workstation and printer, to represent such devices that are often present for workers to perform administrative tasks on-site. The OT devices were configured to use various protocols commonly seen in such environments for management and control. Each of the three sites—the control center, substation, and PV plant—were connected virtually through the ARIES Cyber Range using a representative wide area network (WAN) built on top of several emulated routers participating in a common Open Shortest Path First (OSPF) area, similar to real-world WANs.

A.1.1 Control Center

The control center was designed with the minimal elements required to represent a basic set of services run by the emulated utility. All systems were VMs and ran either Windows or Linux operating systems. The elements included were a SCADA platform running a Human-Machine Interface (HMI), an engineering workstation, and application servers. A DMZ was configured to allow for the isolated deployment of the solution providers' components within that space as needed.

A.1.2 Substation

The substation was designed to represent a geographically separated area from the PV utility-owned site. The substation included several SEL power hardware devices. There was also a Protectli Vault workstation computer and a printer connected to it via Universal Serial Bus (USB) to represent on-site IT resources available at the substation. The only virtualized element of this site was the edge firewall that served as the connection point to the experiment environment.

A.1.3 PV Plant

The PV plant was designed to emulate a small utility-owned solar generation plant, and it included SEL power system devices, a BESS, and a commercial-grade inverter. A TerraSAS Module was connected to the inverter to provide an active direct current (DC) power source as well as input power and output demand set according to a predefined curve in the TerraSAS software. This connection enabled testing with the SMA device, but it only provided power values, so this device was not included as an identifiable asset in the testing environment. The only virtualized element of this site was the edge firewall that served as the connection point to the experiment environment.

CLEAN ENERGY CYBERSECURITY ACCELERATOR

BASELINE OPERATING ENVIRONMENT COHORT 2

Application Layer

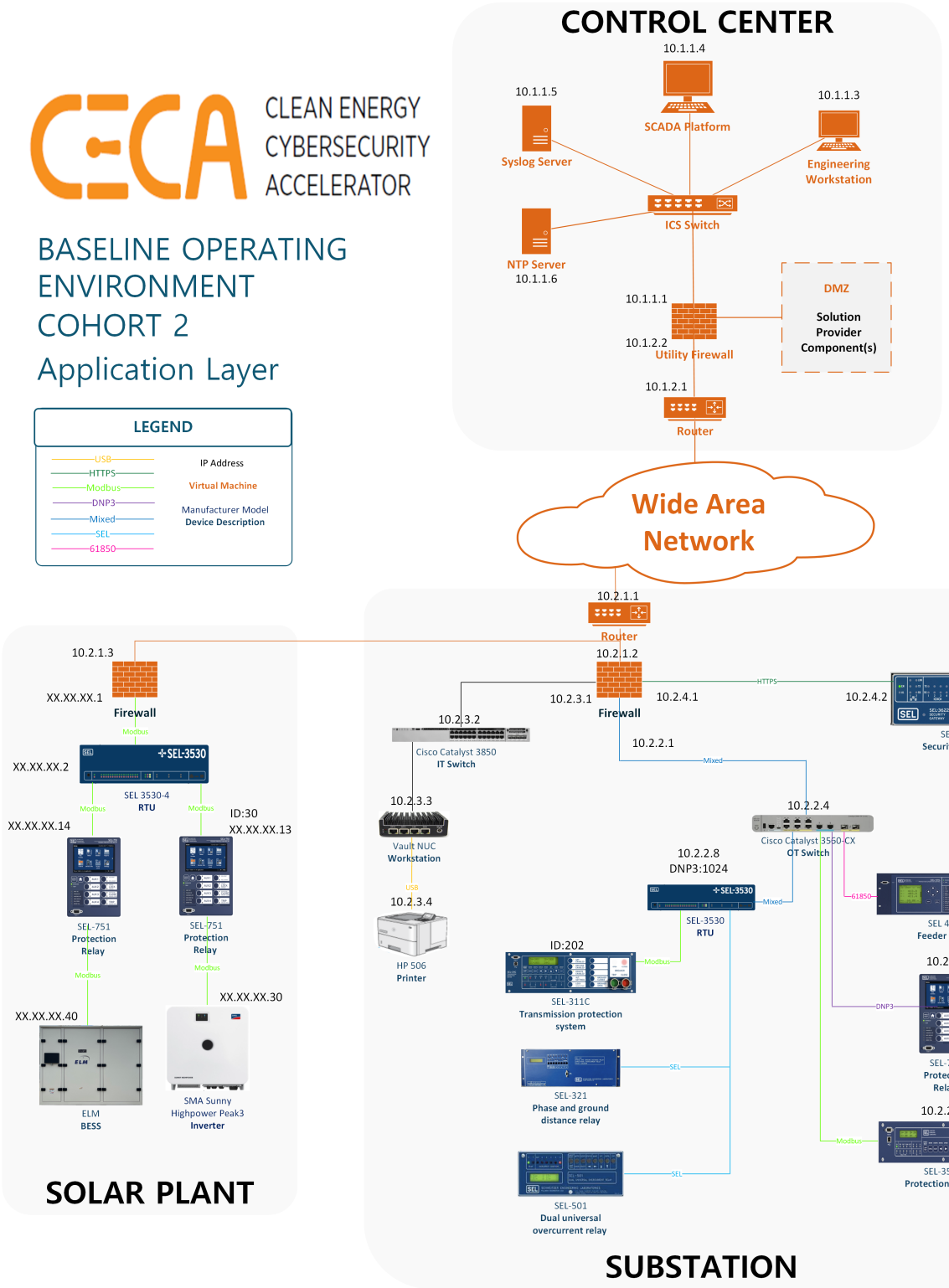


Figure A.1. Cohort 2 application layer BOE[6]

A.2 Network

Seven subnets were used to segment the network. In the virtual environment, all subnets were connected with Internet Protocol Security (IPSec) tunnels over a virtual WAN with redundant internet service provider (ISP) routers.

A.2.1 Subnets

Table A.1. Subnets

Subnet Name	Subnet	Description
CC-LAN	10.1.1.0/24	Dedicated to the control center network
CC-DMZ	10.1.2.0/24	Dedicated to the solution's components to restrict third-party access to the utility's control center network
SUB-DMZ	10.2.1.0/24	Covers the routers to the PV site and substation
SUB-OT	10.2.2.0/24	Covers the substation site's OT components, such as the relays RTU, and serial devices
SUB-OT-GATEWAY	10.2.4.0/24	Dedicated to the security gateway
SUB-IT	10.2.3.0/24	Covers the IT components in the substation
PV	XX.XX.XX.0/24 [6]	Covers the entire PV plant site

A.2.2 Communication Protocols

Table A.2. Protocols

Type	Protocol	Description
OT	DNP3	Commonly used for automation of various industrial systems and components. Examples of devices that are commonly seen to communicate using DNP3 are RTUs, inverters, smart meters, and other such devices related to electrical systems.
OT	Modbus/TCP	IP-based OT protocol that runs on top of the TCP protocol (Modbus 2006a). This protocol is a variant of the Modbus protocol specification (Modbus 2012) maintained by the Modbus Organization.
OT	Modbus/RTU	Serial OT protocol designed to communicate with devices connected over Recommended Standard 232 (RS232) or Recommended Standard 485 (RS485) interfaces (Modbus 2006b). It is a variant of the Modbus protocol specification (Modbus 2012) maintained by the Modbus Organization.
OT	SEL	Proprietary serial OT protocol developed by SEL for communication with devices manufactured by SEL
IT	HTTP	HTTP and its more secure counterpart, HTTPS, are IT protocols commonly used for web applications and general-purpose browsing. HTTPS provides enhanced security through the usage of certificates to encrypt and secure connections.
IT	SMB	Commonly used for sharing files on local or networked data storage
NET	IPSec	IT security protocol commonly used to set up virtual private network (VPN)s or secure tunnels between two remote networks
NET	SNMPv2	Commonly used network protocol to send messages related to the management of networked devices. Community public version used.
NET	OSPF	Routing protocol commonly used in local area network (LAN)s or involved in internet routing. It allows packets to traverse multiple interconnected networks to communicate across large areas and geographic regions.

A.2.3 Firewall Rules

The firewalls specified in the BOE were configured with a default drop policy, and the following ingress allow rules:

Table A.3. Firewall rules

Device	Description	Source	Destination	Port	Protocol
cc-firewall	Allow established				all
sub-firewall					
(to OT LAN)	Allow SCADA platform DNP3 to all OT	10.1.1.4	10.2.2.0/24	20000	TCP
	Allow SCADA platform Modbus to all OT	10.1.1.4	10.2.2.0/24	502	TCP
	Allow SCADA platform SEL protocol to all OT	10.1.1.4	10.2.2.0/24	23	TCP
	Allow established				all
(to IT LAN)	Allow established				all
pv-firewall	Allow SCADA platform Modbus to all OT	10.1.1.4	XX.XX.XX.0/24 [6]	502	TCP
	Allow established				all

A.3 Assets

Table A.4. Asset list

Manufacturer	Model	Function	OS Type
N/A (VM)	Vyatta	Firewalls	VyOS
N/A (VM)	qcow2	SCADA platform	Windows
N/A (VM)	qcow2	Engineering workstation	Windows
N/A (VM)	qcow2	Syslog server	Linux
N/A (VM)	qcow2	NTP server	Linux
Cisco Systems	3850	PV switch	Cisco IOS
Cisco Systems	3560-CX	Substation switch	Cisco IOS
Hewlett Packard	506 Laserjet	Printer	Laserjet Enterprise
Protectli	FW4B	Workstation	Windows
SEL	311C	Transmission protection system	SEL Embedded Linux
SEL	321	Phase and ground distance relay	SEL Embedded Linux
SEL	351A	Protection system	SEL Embedded Linux
SEL	3530	RTAC	SEL Embedded Linux
SEL	3530-4	RTAC	SEL Embedded Linux
SEL	3622	Security gateway	SEL Embedded Linux
SEL	411L	Feeder relay	SEL Embedded Linux
SEL	501	Dual universal overcurrent relay	SEL Embedded Linux
SEL	751	Protection relay	SEL Embedded Linux
ELM	501	BESS	Windows
SMA	SHP 125-US-20	PV inverter	Linux

A.4 Monitoring

A.4.1 Pretesting

Before each test, the phenix state of health (SoH) app[7] was used to validate the environment configuration and to ensure that all assets were available and communicating.

A.4.2 During Testing

To verify that each OT device continued to perform its intended function throughout each active scanning scenario, and to ensure no underlying ICS processes were affected by active scanning, CECA used two different techniques. First, assets that were reachable via ICMP were polled every second, and response latency was monitored.

Second, the data acquisition portion of the environment's SCADA processes was monitored to ensure that no communications were unavailable throughout the duration of each test. The OT devices in the PV plant and substation

[7] For more information on phenix apps, see Appendix B

each had several registers monitored by the SCADA platform in the control center. These values were polled via DNP3, Modbus, or the SEL protocol (in some instances forwarded or aggregated by an intermediate RTU), at a frequency of once every 5 seconds. During normal environment operations, these values were monitored by the SCADA platform and displayed on the accompanying control center Node-RED HMI. At the end of each test, the logs from the SCADA platform were retrieved and checked for any failures to read from an outstation device.

Appendix B. Evaluation Tools

The assessment was conducted in the NREL ARIES Cyber Range, which uses a collection of open-source and custom tools to emulate complex ICS systems.

B.1 Minimega

Minimega is an open-source tool for starting and managing multiple VMs (SNL 2023) (SNL 2024a). Minimega is based on the quick emulator (QEMU) hypervisor.

B.2 Phenix

Phenix is an open-source application wrapping multiple tools that orchestrates the definition, configuration, deployment, and management of VMs, scenarios, and hardware into various experiments. Phenix flexibly integrates virtualized and hardware components into environments and can be customized using the phenix application framework. Several built-in and custom phenix applications used by CECA are detailed in the following (SNL 2024b) (SNL 2024c).

B.2.1 Scorch (Core App)

Scorch (SCenario ORCHestration) is an automated scenario orchestration framework within phenix. Scorch provides the ability to create customizable attack and data collection pipelines for efficient and repeatable assessment (Patria 2024a). These pipelines can launch Atomic Red Team and other command-based attacks in addition to instrumenting assessment data collection during the scenario to enable subsequent analysis. Using Scorch allows for entire evaluation scenarios to be documented in a file using an assessment as code (AaC) methodology.

B.2.2 State of Health (Core App)

The SoH app continuously collects the state of components in the virtual environment (Patria 2024b). It visually renders the state using a network graph for quick overview. A set of predefined measurements—central processing unit (CPU) load, open ports, running processes, reachability, etc.—can be gathered on the VMs as part of the SoH test. Custom tests specific to the environment can also be configured. SoH simplifies the monitoring of the complex experiment with both virtual and hardware devices connected through the ARIES Cyber Range infrastructure.

B.2.3 Vrouter (Core App)

The virtual router (vrouter) is a phenix app that enables the automated configuration of routers and firewalls in phenix experiments (Patria 2024c). The vrouter app can configure IPSec tunnels, firewall rules, dynamic host configuration protocol (DHCP) settings, DNS entries, and source network address translation (SNAT) or destination network address translation (DNAT), and it can add custom traffic profile emulations to any supported router running in a phenix experiment.

B.2.4 AMI (Custom App)

The custom AMI phenix app allows researchers to instantiate thousands of small IoT-like devices in minutes. Each device is a minimega container possessing a TCP/IP networking stack for switching and routing like any networking device would require to function on Ethernet. The application reads a configuration from an underlying power model and determines a set number of containers required to represent the devices within it (Wallace et al. 2024). It creates a number of VMs on which to initiate the containers. For example, in Scenario 4, the app was used to deploy 14 VMs with 150 containers, totaling exactly 2,014 AMI-representative devices in the experiment. Each device functioned as a smart meter with networking capability. This virtualization allowed for many more networked end points to be included in the environment than in previous tests.

B.3 OT-sim

The Operational Technology Simulator (OT-sim) is a software tool developed by Patria Security LLC to simulate various components of an OT device using a module-based approach (Patria 2023). OT-sim is deployed as a set of binaries, each for a module that can be configured to run a simulated component of an OT device within VMs or containers. Through the deployment of this tool, along with any necessary configurations in an automated manner through phenix, the ARIES Cyber Range allows researchers to represent a physical system, at scale, in a

co-simulation environment. The specific OT-sim modules used in this experiment were: CPU, DNP3, Modbus, and Node-RED. Together, they were deployed in a configuration serving as a HMI for the physical devices in the experiment.

B.4 Node-RED

Node-RED is a flow-based programming tool developed by the OpenJS Foundation (OpenJS 2024) that provides a browser-based editor for developing applications that have a user-interfacing dashboard. The tool works with hardware devices, application programming interface (API)s, and other peripheral interfaces using a software plug-in framework. Node-RED provides both a graphical user interface (GUI) for the development of applications and dashboards and simple user access control for deployed applications using passwords and configurable user credentials. Configurations can also be exported and imported as javascript object notation (JSON) files. Doing so allows for easy modification of templates because the modified JSON file can be injected into the VM running Node-RED each time an experiment is started. For the CECA Cohort 2 experiments, Node-RED was deployed through an OT-sim module and configured using the JSON import method.

Appendix C. Configuration of Technology

C.1 Version

The CECA Cohort 2 testing used self-hosted Asimily servers and edge controllers version 5.5.2.

C.2 Installation

CECA followed the installation documentation provided by Asimily and created individual images for the servers and edge collectors. Installation scripts for the two servers, asimily-main and asimily-ad, provided by Asimily, were run on these VMs and configured.

C.3 API

The Asimily API was heavily leveraged to enable CECA's methodology to allow repeatable runs of each test. For details about all API calls, see Appendix D - Procedures.

Appendix D. Evaluation Procedures

The specific steps for each evaluation are described in this section. Using the phenix Scorch app and custom bash scripts, CECA translated the testing plan into a suite of tests that are scientific and repeatable. Each scenario can be thought of as a "program" that comprises several function calls, which are called "components" in Scorch. These components are listed in a separate section because many of them are repeated across several scenarios.

D.1 Scenarios

D.1.1 Scenario 1: Initial Discovery

Steps

1. `soh`: Run the phenix SoH app to ensure that all assets in the environment are powered and working as expected.
2. `get-asimily-info`: Perform a series of command line interface (CLI) commands to record basic information about the Asimily solution before starting the test.
3. start `tcpdump-solution-Sc123`: Start tcpdump on each Asimily edge processor, main server, and Asimily anomaly detection server (AD) Asimily database server (DB) server.
4. start `tcpdump-mirror-Sc123`: Start tcpdump on the mirror interface on each Asimily edge processor.
5. start `tcpdump-firewalls`: Start tcpdump on each firewall.
6. `pause30m`: Wait for 30 minutes.
7. stop `tcpdump-solution-Sc123`: Stop the tcpdumps started in Step 3.
8. stop `tcpdump-mirror-Sc123`: Stop the tcpdumps started in Step 4.
9. stop `tcpdump-firewalls`: Stop the tcpdumps started in Step 5.
10. `download-database`: Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.
11. `recv-files`: Download the artifacts created by this run for offline analysis.

D.1.2 Scenario 2: Change Discovery

Scenario 2 is exactly the same as Scenario 1. The only differences are:

- VM images are used to boot the Asimily main, and AD servers are "onboarded" in the state that they were at the end of Scenario 1.
- SoH checks in Step 1 are customized to the modified environment.

Steps

1. `soh`: Run the phenix SoH app to ensure that all assets in the environment are powered and working as expected.
2. `get-asimily-info`: Perform a series of CLI commands to record basic information about the Asimily solution before starting the test.
3. start `tcpdump-solution-Sc123`: Start tcpdump on each Asimily edge processor, main server, and ad-db2 server.
4. start `tcpdump-mirror-Sc123`: Start tcpdump on the mirror interface on each Asimily edge processor.
5. start `tcpdump-firewalls`: Start tcpdump on each firewall.

6. `pause30m`: Wait for 30 minutes.
7. stop `tcpdump-solution-Sc123`: Stop the tcpdumps started in Step 3.
8. stop `tcpdump-mirror-Sc123`: Stop the tcpdumps started in Step 4.
9. stop `tcpdump-firewalls`: Stop the tcpdumps started in Step 5.
10. `download-database`: Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.
11. `recv-files`: Download the artifacts created by this run for offline analysis.

D.1.3 Scenario 3: Alternate Discovery

Scenario 3 is similar to Scenario 1, except:

- Steps 3 and 11 are added to detect any unresponsive hosts that are affected by active scanning.
- Step 14 is added to download additional artifacts created by active scanning.
- Asimily edge configuration files are modified to set `ENABLE_ACTIVE_SCANNER=1`.
- Asimily main server's cronjob settings are updated to enable periodic active scanning.

Steps

1. `soh`: Run the phenix SoH app to ensure that all assets in the environment are powered and working as expected.
2. `get-asimily-info`: Perform a series of CLI commands to record basic information about the Asimily solution before starting the test.
3. start `measure-disruption`: Start the ICMP polling apparatus to detect any unresponsive hosts.
4. start `tcpdump-solution-Sc123`: Start tcpdump on each Asimily edge processor, main server, and addb2 server.
5. start `tcpdump-mirror-Sc123`: Start tcpdump on the mirror interface on each Asimily edge processor.
6. start `tcpdump-firewalls`: Start tcpdump on each firewall.
7. `pause30m`: Wait for 30 minutes.
8. stop `tcpdump-solution-Sc123`: Stop the tcpdumps started in Step 4.
9. stop `tcpdump-mirror-Sc123`: Stop the tcpdumps started in Step 5.
10. stop `tcpdump-firewalls`: Stop the tcpdumps started in Step 6.
11. stop `measure-disruption`: Stop the polling apparatus started in Step 3.
12. `download-database`: Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.
13. `recv-files`: Download the artifacts created by this run for offline analysis.
14. `recv-files-active`: Download the artifacts created by active scanning during this run for offline analysis.

D.1.4 Scenario 4: Scale Discovery

Scenario 4 follows a similar template as the previous scenarios, but it is tailored to a different environment, and the steps takes place in two phases.

Scenario 4A: Scale Discovery—Initial

1. `delete-previous`: Manually remove old entries in the Asimily asset database to ensure that the runs start with 0 devices.
2. `soh`: Run the phenix SoH app to ensure that all assets in the environment are powered and working as expected.
3. `get-asimily-info`: Perform a series of CLI commands to record basic information about the Asimily solution before starting the test.
4. start `tcpdump-firewall-Sc4`: Start tcpdump on the firewall.
5. start `tcpdump-solution-Sc4`: Start tcpdump on the Asimily edge processor, main server, and ad-db2 server.
6. start `tcpdump-mirror-Sc4`: Start tcpdump on the mirror interface on the Asimily edge processor.
7. `create-traffic`: Begin background processes with network traffic between AMI devices.
8. `pause30m`: Wait for 30 minutes.
9. `watch-asimily`: Manually trigger proceeding to the next step.[8]
10. `download-database-Sc3`: Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.
11. `recv-files-Sc3`: Download the artifacts created by this run for offline analysis.

Scenario 4b: Scale Discovery—Second

12. `create-alert`: Manually create a rule in the Asimily GUI to alert based on the newly identified devices.
13. `turn-on-attacker`: Start the attacker VM.
14. `pause5m`: Wait for 5 minutes.
15. stop `tcpdump-firewall-Sc4`: Stop the tcpdumps started in Step 4.
16. stop `tcpdump-solution-Sc4`: Stop the tcpdumps started in Step 5.
17. stop `tcpdump-mirror-Sc4`: Stop the tcpdumps started in Step 6.
18. `check-alert`: Manually check for an alert for the rule created in Step 12.
19. `download-database-Sc3`: Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.
20. `recv-files-Sc3`: Download the artifacts created by this run for offline analysis.

D.2 Components

D.2.1 soh

Run the phenix SoH app to ensure that all assets in the environment are alive and working as expected.

[8] This manual check ensured that the Asimily solution completed processing before proceeding to Scenario 4B. The 30-minute pause period was occasionally insufficient for the solution to complete parsing and displaying all identified assets. Identification of all assets required as much as 30 additional minutes, but all assets were identified within 60 minutes across all runs.

D.2.2 get-asimily-info

Perform a series of CLI commands to record information about the Asimily solution before a test was started.

Code

(On the Asimily main server)

```
PGPASSWORD=REDACTED_PASSWORD psql -U redacted_user redacted_db -c
 ↪  "select * from redacted_table;" |
tee /root/edge_facility_map.txt
```

Which is validated to ensure that the the appropriate number of rows is returned before the pipeline continues.

D.2.3 tcpdump-solution-Sc123

Start and stop tcpdump on each Asimily edge processor and server.

D.2.4 tcpdump-mirror-Sc123

Start and stop tcpdump on each Asimily edge processor's mirror interface.

D.2.5 tcpdump-firewalls

Start and stop tcpdump on each firewall interface in the environment.

D.2.6 pause30m

Wait for 30 minutes.

D.2.7 download-database

Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.

(On the helper VM that interacts with the Asimily portal)

```
curl -k -s -D /root/get-assets-header.txt -u
 ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "Content-Type:
 ↪  application/json" https://10.1.2.10:443/api/redacted/api/url |
  jq |
  tee /root/inventory.json
q_did="deviceIds="
for id in $(cat /root/inventory.json | jq '.content | .[].deviceID'); do
  q_did+=$id
  q_did+=","
done
q_did=${q_did::-1}
curl -k -s -D /root/get-app-ports-header.txt -u
 ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "Content-Type:
 ↪  application/json"
 ↪  https://10.1.2.10:443/api/extapi/assets/device-apps-ports?$q_did |
  jq |
  tee /root/app-ports.json
mkdir -p /root/api-outputs/anomaly
mkdir -p /root/api-outputs/port
mkdir -p /root/api-outputs/application
mkdir -p /root/api-outputs/cve
```

```
for mac in $(cat /root/inventory.json | jq '.content | .[].macAddr'); do
  mac=${mac:1:-1}
  out_file=${mac//:/-}
  curl -k -s -D /root/api-outputs/anomaly/header-$out_file.txt -u
   ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "accept: */*"
   ↪  https://10.1.2.10:443/api/redacted/api/url?macAddr=$mac |
    jq |
    tee /root/api-outputs/anomaly/$out_file.json
  curl -k -s -D /root/api-outputs/port/header-$out_file.txt -u
   ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "accept: */*"
   ↪  https://10.1.2.10:443/api/redacted/api/url?macAddr=$mac |
    jq |
    tee /root/api-outputs/port/$out_file.json
  curl -k -s -D /root/api-outputs/application/header-$mac.txt -u
   ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "accept: */*"
   ↪  https://10.1.2.10:443/api/redacted/api/url?macAddr=$mac |
    jq |
    tee /root/api-outputs/application/$out_file.json
  curl -k -s -D /root/api-outputs/cve/header-$out_file.txt -u
   ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "accept: */*"
   ↪  https://10.1.2.10:443/api/redacted/api/url?macAddr=$mac |
    jq |
    tee /root/api-outputs/cve/$out_file.json
done
tar czvf /root/api-outputs.tgz /root/api-outputs/
```

D.2.8 recv-files

The following CLI commands were used to save the Asimily databases at the end of each run.

Code

(In the Asimily main server)

```
PGPASSWORD=REDACTED_PASSWORD /usr/bin/pg_dump -U redacted_user -Fc
 ↪  redacted_db --exclude-table 'redacted_table' -f /tmp/test_main.backup
```

(In the Asimily adb-db2 server)

```
PGPASSWORD=REDACTED_PASSWORD /usr/bin/pg_dump -U redacted_user -Fc
 ↪  redacted_db -f /tmp/test_ad.backup
```

Then extract the following files from the following hosts for offline analysis:

- Helper VM:
  - `/root/inventory.json`
  - `/root/get-assets-header.txt`
  - `/root/app-ports.json`
  - `/root/get-app-ports-header.txt`
  - `/root/api-outputs.tgz`

- Asimily main server:
  - `/tmp/test_main.backup`
  - `/root/edge_facility_map.txt`
- Asimily ad-db2 server:
  - `/tmp/test_ad.backup`

D.2.9 measure-disruption

Starts and stops the ICMP polling apparatus to detect any unresponsive hosts.

Start

Code

(In each monitoring VM)

```
tmux new-session -s md -d \
  "fping -l -p 1000 -t 200 -r 3 -D -e -f /root/cc-monitor-ips.txt >
   ↪  /root/cc-disruption-results.txt 2>&1" &&
  true
```

Stop

Code

(In each monitoring VM)

```
tmux send-keys -t md "C-c"
```

(In the SCADA platform)

```
journalctl -u ot-sim -g ERROR > /root/ot-sim-errors.log
```

After which the file `/root/cc-disruption-results.txt` is extracted from each monitoring VM, and `/root/ot-sim-errors.log` is extracted from the SCADA platform for analysis.

D.2.10 recv-files-active

Extract the following files from the following hosts for offline analysis:

- Asimily main server: `/var/log/asimily/active_scanner.log`
- Asimily edge processor 1: `/var/log/asimily/active_scanner.log`
- Asimily edge processor 2: `/var/log/asimily/active_scanner.log`
- Asimily edge processor 3: `/var/log/asimily/active_scanner.log`
- Asimily edge processor 4: `/var/log/asimily/active_scanner.log`
- Asimily edge processor 5: `/var/log/asimily/active_scanner.log`

D.2.11 delete-previous

This is a "break" component that pauses execution until it is manually exited. During this time, CECA used the Asimily portal to delete devices in the database that were seen during the setup.

D.2.12 tcpdump-firewall-Sc4

Start and stop tcpdump on the firewall in the environment.

D.2.13 tcpdump-solution-Sc4

Start and stop tcpdump on the Asimily edge processor and both Asimily servers.

D.2.14 tcpdump-mirror-Sc4

Start and stop tcpdump on the Asimily edge processor's mirror interface.

D.2.15 create-traffic

Begin generating traffic to all AMI devices in the environment.

Code

(In the AMI admin VM)

```
date > /root/create-traffic.log
while true
do
  for ii in $(seq 0 7);
  do
    for jj in $(seq 0 255);
    do
      if [ $ii -eq 7 ] && [ $ii -ge 222 ];
      then
        echo "pass on $10.200.$ii.$jj" >> /root/create-traffic.log
      else
        curl "http://10.200.$ii.$jj:9101/api/v1/query" >>
         ↪  /root/create-traffic.log
      fi
    done
    sleep 10
  done
done
```

D.2.16 download-database-Sc3

Perform a series of API requests and CLI commands to record information from the Asimily servers at the conclusion of the test.

(On the helper VM that interacts with the Asimily portal)

```
for ii in $(seq 0 4);
do
    curl -k -s -D "/root/get-assets-header$ii.txt" -u
     ↪  "redacted_user:REDACTED_PASSWORD" -X GET -H "Content-Type:
     ↪  application/json"
     ↪  "https://10.1.2.10:443/api/redacted/api/url?page=$ii" |
    jq |
    tee "/root/inventory$ii.json"
done
```

D.2.17 recv-files-Sc3

The following CLI commands were used to save the Asimily databases at the end of each run.

Code

(In the Asimily main server)

```
PGPASSWORD=REDACTED_PASSWORD /usr/bin/pg_dump -U redacted_user -Fc
 ↪  redacted_db --exclude-table 'redacted_table' -f /tmp/test_main.backup
```

(In the Asimily adb-db2 server)

```
 PGPASSWORD=REDACTED_PASSWORD /usr/bin/pg_dump -U redacted_user -Fc
  ↪  redacted_db -f /tmp/test_ad.backup
```

Then extract the following files from the following hosts for offline analysis:

- Helper VM:
  - `/root/inventory0.json`
  - `/root/get-assets-header0.txt`
  - `/root/inventory1.json`
  - `/root/get-assets-header1.txt`
  - `/root/inventory2.json`
  - `/root/get-assets-header2.txt`
  - `/root/inventory3.json`
  - `/root/get-assets-header3.txt`
  - `/root/inventory4.json`
  - `/root/get-assets-header4.txt`
- Asimily main server:
  - `/tmp/test_main.backup`
  - `/root/edge_facility_map.txt`
- Asimily ad-db2 server:
  - `/tmp/test_ad.backup`
- AMI admin VM:
  - `/root/create-traffic.log`

D.2.18 create-alert

This is a "break" component that pauses execution until it is manually exited. During this time, CECA created an internal alert using the Asimily anomaly detection rules for any new device identified.

D.2.19 turn-on-attacker

Use minimega to start the attacker VM, which adds it to the environment.

D.2.20 pause5m

Wait for 5 minutes.

D.2.21 check-alert

This is a "break" component that pauses execution until it is manually exited. During this time, CECA checked the Asimily portal for anomalies and recorded a screenshot.

CESER
PUBLIC REPORTS

U.S. DEPARTMENT OF ENERGY
Office of Cybersecurity, Energy Security, and Emergency Response