ADSICS: Anomaly Detection System for Industrial Control Systems

Design Document

Team Number
38
Client
Manimaran Govindarasu
Advisers
Manimaran Govindarasu
Moataz A. Abdelkhalek
Xinyao Li
Team Members/Roles
Planning: Pallavi Santhosh

Communication: Jungho Suh

Organizing: Alex Nicolellis

Controlling: Muhamed Stilic

Team Email
sdmay22-38@iastate.edu
Team Website
https://sdmay22-38.sd.ece.iastate.edu/

Executive Summary

Development Standards & Practices Used

Procedures: Agile

Standards: IEEE 1711.2-2019, IEEE 2030.5, IEEE 692-2013, IEEE 2413-2019, ISO IEC 27039-2015, ISO/IEC 27017:2015 & IEEE 802

Summary of Requirements

Functional Requirements:

- Use machine learning to detect network anomalies
- Verify incoming alerts and discard false positives
- Display alerts for easy human understanding
- Present temporal and spatial details for each alert

Non-Functional Requirements:

- Alerts should be presented intuitively
- Alerts should be received within 10ms
- The system should be able to handle a large volume of alerts
- The system should be reliable and maintain uptime continuously

Applicable Courses from Iowa State University Curriculum

COMS 474, CPRE 431, CPRE 430, SE 422X

New Skills/Knowledge acquired that was not taught in courses

Elasticsearch, SecurityOnion, VMware, Machine Learning Algorithms, Hacking Scripts

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

1 Team SDMay22-38

1.1 Team Members

Alex Nicolellis, Jung Ho Suh, Muhamed Stilic, Pallavi Santhosh

1.2 Required Skill Sets for Your Project

Exposure to cloud programming, cybersecurity tools, machine learning algorithms, or python programming.

1.3 Skill Sets covered by the Team

Machine learning algorithms: Alex, Jungho, Pallavi

Python Programming: Alex, Jungho, Muhamed

Cloud Programming: None

Cybersecurity tools: None

1.4 Project Management Style Adopted by the team

Agile

1.5 Initial Project Management Roles

Planning: Pallavi

Communicating to the client: Jungho

Organizing: Alex

Controlling: Muhamed

2 Introduction

2.1 Problem Statement

Attacks on power distribution companies are now more common due to the increased use of IoT devices and the lack of security on power grid systems. ADSICS is a surveillance program that detects and prevents cyber attacks using anomaly detection.

2.2 Requirements & Constraints

Functional Requirements:

- Use machine learning to detect network anomalies
- Verify incoming alerts and discard false positives
- Display alerts for easy human understanding
- Present temporal and spatial details for each alert

Non-Functional Requirements:

- Alerts should be presented intuitively
- Alerts should be received within 10ms
- The system should be able to handle a large volume of alerts
- The system should be reliable and maintain uptime continuously

Constraints:

- The anomaly detection must use SecurityOnion tools (specifically Elasticsearch)

2.3 Engineering Standards

Standard	Application	Justification
IEEE 692-2013	IADS SENSOR IADS MASTER	Addresses cybersecurity and control related equipment requirements for threat assessment.
ISO IEC 27039-2015	IADS MASTER	Provides guidelines for selection, deployment, and operations of intrusion detection system detection and prevention systems.

ISO/IEC 27017:2015	CLOUD SERVER	Provides guidance on the information security aspects of cloud computing, recommending the implementation of cloud-specific information security controls that supplement the guidance of the **ISO/IEC 27002** and **ISO/IEC 27001** standards.
IEEE 1711.2-2019	IADS SENSOR	Protects communication of intelligent devices in the power industry.
IEEE 802	IADS SENSOR	Describes recommended practices for communication over various types of networks, such as wireless networks.

[1]

2.4 Intended uses and users

As of now, the detection system in place is lacking a master program to analyze security breach alerts. Once implemented, a major benefit of our program will be that it takes away the need for a host to double check whether or not a detected anomaly is a false positive because it will be able to do that as part of its design.

Power distribution and utility companies (ex. City of Ames Iowa Electric Services) will benefit from the implementation of this project because they are the ones who will have to deal with intrusions on the grid. More areas include (but are not limited to) DER field devices, plant controllers, and grid edge hardware.

3 Project Plan

3.1 Project Management/Tracking Procedures

Agile - The reason why we chose agile is because it's easy to break up the project's specific steps in order to create our system.

Discord - The reason why we chose discord is because it's an easy way to communicate to one another and we can pin certain messages to exactly know what's going on.

[1] Engineering Standards

3.2 Task Decomposition

In order to solve the problem at hand, it helps to decompose it into multiple tasks and subtasks and to understand interdependence among tasks. This step might be useful even if you adopt agile methodology. If you are agile, you can also provide a linear progression of completed requirements aligned with your sprints for the entire project.

(Start 9/13/21):

Sprint1: Research about cybersecurity basics

Sprint2: Research and work with the power grid test-bed

Sprint3: Create 3 VM's(Attacker,Target,SNORT(primary focus)) and Git LAB

Sprint4: Data augmentation using existing open-source dataset and actual data

Sprint5: Create Security Onion VM to correlate alerts

(Start 1/17/22):

Sprint1: Set Up IADS Sensor VM to provide filtered information

Sprint2: Test connection between Master and Sensor

Sprint3: Use machine learning algorithms to detect anomalies

Sprint4: Compare between algorithms to select ideal one

Sprint5: Put IADS Master in cloud and test correlation algorithms here

Sprint6: Build the user interface in the cloud including data visualization

Sprint7: Review and test the final system

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

Key Milestones:

1. Filter and track alerts through spatial and temporal data
2. Accept continuous alerts from the sensor VMs
3. Build a UI to display tracked anomaly data intuitively
4. Process every alerts in 10 ms delay for real time operation
5. Eliminate false positives with 90% accuracy

3.4 Project Timeline/Schedule

Gantt Chart:

Fall 2021:

PROJECT TITLE	SDMAY22-38	COMPANY NAME	SDMAY22-38
PROJECT MANAGER	Manimaran Govindarasu	DATE	10/10/21

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	Sprint 1					
1.1	Research Security Basics	All	9/13/21	9/26/21	9	100%
1.1.1	Read IDS paper	All	9/13/21	9/26/21	7	100%
1.2	Explore cybersecurity tools	All	9/13/21	9/26/21	7	100%
1.3	Project requirements	All	9/13/21	9/26/21	8	100%
2	Sprint 2					
2.1	Work with test bed attacks	All	9/27/21	10/10/21	5	100%
2.2	Firewall rule edit	All	9/27/21	10/10/21	6	100%
2.3	Sniffing packets using Wireshark	All	9/27/21	10/10/21	6	100%
2.3	Vulnerability scanning using Nmap	All	9/27/21	10/10/21	6	100%
2.4	Perform attack and observe the effect	All	9/27/21	10/10/21	6	100%
3	Sprint 3					
3.1	Create GitLab	Alex N.	10/11/21	10/22/21	5	100%
3.2	Create VM Attacker	Jung S.	10/11/21	10/22/21	5	100%
3.2.1	Create VM Target	Muhamed S.	10/11/21	10/22/21	3	100%
3.2.2	Create VM SNORT	Pallavi S.	10/11/21	10/22/21	5	100%
3.3	Work with SecurityOnion	All	10/11/21	10/22/21	6	100%
4	Sprint 4					
4.1	Explore Kibana	All	10/22/21	11/1/21	6	100%
4.2	Visualize graphs using Kibana sample dataset	All	10/22/21	11/1/21	6	100%
4.3	Try Kibana's built in anomaly detection system	All	10/22/21	11/1/21	5	100%
5	Sprint 5					
5.1	Explore Elasticsearch	Alex N.	11/5/21	11/19/21	6	100%
5.2	Explore IRIS dataset in local machine	Jung S.	11/5/21	11/19/21	6	100%
5.3	Run machine learning in IRIS dataset	Muhamed S.	11/5/21	11/19/21	6	100%
5.4	Run IRIS dataset in Elasticsearch	Pallavi S.	11/5/21	11/19/21	7	100%
5.5	Run NSLKDD dataset in Elasticsearch	All	11/5/21	11/19/21	5	100%

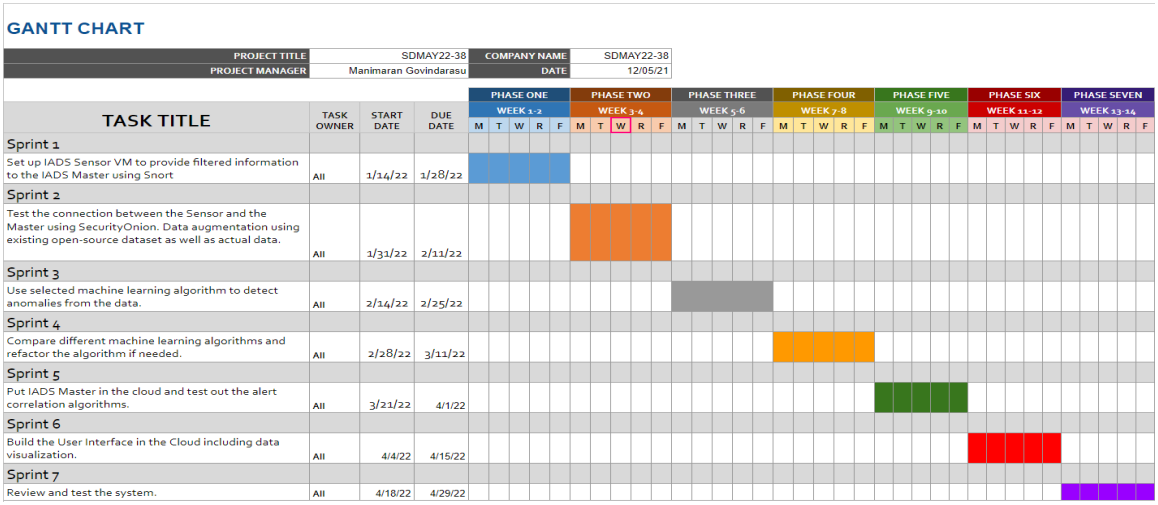
PHASE ONE: WEEK 1, WEEK 2 | PHASE TWO: WEEK 3, WEEK 4 | PHASE THREE: WEEK 7, WEEK 8 | PHASE FOUR: WEEK 10, WEEK 11 | PHASE FIVE: WEEK 10, WEEK 11

[2]

https://docs.google.com/spreadsheets/d/1OoQfat1zegoAtTNIzD_3ElfAQm126rLRqIKcQKo-1Do/edit?usp=sharing

Spring 2022:

[2] Semester 1 GANTT chart



GANTT CHART

PROJECT TITLE	SDMAY22-38	COMPANY NAME	SDMAY22-38
PROJECT MANAGER	Manimaran Govindarasu	DATE	12/05/21

TASK TITLE	TASK OWNER	START DATE	DUE DATE
Sprint 1			
Set up IADS Sensor VM to provide filtered information to the IADS Master using Snort	All	1/14/22	1/28/22
Sprint 2			
Test the connection between the Sensor and the Master using SecurityOnion. Data augmentation using existing open-source dataset as well as actual data.	All	1/31/22	2/11/22
Sprint 3			
Use selected machine learning algorithm to detect anomalies from the data.	All	2/14/22	2/25/22
Sprint 4			
Compare different machine learning algorithms and refactor the algorithm if needed.	All	2/28/22	3/11/22
Sprint 5			
Put IADS Master in the cloud and test out the alert correlation algorithms.	All	3/21/22	4/1/22
Sprint 6			
Build the User Interface in the Cloud including data visualization.	All	4/4/22	4/15/22
Sprint 7			
Review and test the system.	All	4/18/22	4/29/22

PHASE ONE	PHASE TWO	PHASE THREE	PHASE FOUR	PHASE FIVE	PHASE SIX	PHASE SEVEN
WEEK 1-2	WEEK 3-4	WEEK 5-6	WEEK 7-8	WEEK 9-10	WEEK 11-12	WEEK 13-14
M T W R F	M T W R F	M T W R F	M T W R F	M T W R F	M T W R F	M T W R F

3

https://docs.google.com/spreadsheets/d/1h7w3d-d7OkNHX0BU59j-0NgOR_MuK7RG5NPXqGV5VJc/edit?usp=sharing

3.5 Risks And Risk Management/Mitigation

Agile projects can associate risks and risk mitigation with each sprint.

Risk Factor & Mitigation Table:

Risk Factor	Probability	Risk Mitigation
The false positive rate is higher than +10%	0.8	Adjusting parameters and applying different forms of analysis.
Attack types are identified incorrectly	0.3	Adjusting the algorithm so it can identify all types of attacks
System is too slow (>10ms response time)	0.7	Adjust input volume and identify inefficiencies.
Loss of data	0.7	Adjusting the backend system so no data gets leaked
System is down	0.8	Adjusting a secure back-up incase of an emergency

4

3.6 Personnel Effort Requirements

[3] Semester 1 GANTT chart
[4] Risk Factor and Mitigation Table

Semester 1:

Tasks	Person-Hours
S1 TASK 1 - Create Testbed with SO VM	5 HRS
S1 TASK 2 - Read project materials	5 HRS
S1 TASK 3 - Experiment with testbed tools (modules)	20 HRS
S2 TASK 1 - Research anomaly detection techniques	10 HRS
S2 TASK 2 - Experiment with Elastic Stack tools	20 HRS
S2 TASK 3 - Research relevant datasets	5 HRS
S2 TASK 4 - Test Security Onion VM with alerts*	10 HRS
S3 TASK 1 - Implement machine learning algorithms	20 HRS
S3 TASK 2 - Refine design by comparing and contrasting algorithms and datasets	5 HRS
S4 TASK 1 - Implement design using Elastic Stack	20 HRS
S5 TASK 1 - Implement design on the SO testbed	20 HRS

See Appendix below for Textual References on specific tasks* [5]

3.7 Other Resource Requirements

- Pre Existing IDS structure
- KDD, ICS, NSL+KDD, IoT 23 datasets
- **D-IDS for Cyber-Physical DER Modbus System - Architecture, Modeling, Testbed-based Evaluation (see section 8.3 for full reference)**
- Faculty Members: Moataz Abdelkhalek, Manimaran Govindarasu

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	Description	Examples

[5] Sprints Semester 1

Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	Hardening the security in the power grid increases the well-being of the users by securing constant power flow. Detecting the anomaly in the power grid to find out potential cyber threat in the early stage is critical to ensure the users private data such as electricity usage. Reducing the risk of cyber attack that could take down the entire power grid, unabling electricity.
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Proper security is a necessary value of any company, especially for those that heavily use computing resources. Our project provides useful security measures that help an energy company uphold this value. This project indirectly supports renewable energy systems by protecting a system of solar panels. Supporting alternative energy sources is a goal of many cultures in the modern world.
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	Our project serves to support a renewable energy source, which provides a comparatively better environmental impact than traditional energy sources. The computing resources of our project consume energy, but that cost should be offset by the protection it grants to the solar panel system.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	When malicious actors tamper with power grids systems the loss of service increases costs for consumers and businesses, so our project is here to mitigate those costs.

4.1.2 User Needs

Group: Power-Grid Employees

- Need a way to monitor the entire power grid.
- Need a way to track down the origin sensor of each anomaly.
- Need a way to communicate with each DER to get the detailed information.
- Need a way to update DER's policy and software remotely.

Group: IDS Algorithm Developers

- Need a way to fix bugs in code
- Need a way to check for logs
- Need a way to run tests

4.1.3 Prior Work/Solutions

Background work:

Mirheidari et al: Alert Correlation Algorithms: A Survey and Taxonomy

- This work lists the primary challenges of an anomaly detection algorithm and a categorization and comparison of various types of algorithms, notably including machine learning.

Zang et al: A Survey of Alert Fusion Techniques for Security Incident

- This work describes the process of alert correlation step-by-step.

Sadoddin, Ghorbani: Alert Correlation Survey: Framework and Techniques

- In addition to explaining the process of alert correlation, this paper goes into detail about different alert correlation algorithms and identification of false positives.

Previous work:

- Using RED (Reconstruction Error Distribution) and HPCs (Hardware Performance Counters) to detect and protect against cyber attacks and issues in the power grid.Focuses on zero-day attacks to prevent against previously undetected gaps in security an attacker may use to shut down the system (He et al., 2019)
- Analyzes the restrictions on detection methods for anomalies that electricity suppliers have to deal with. Focuses on cloud computing and prevention of internalized errors such as design flaws rather than external infiltrators. (Feng et al., 2020)
- Uses Micro-MPUs to better detect anomalies in power grid security as well as expand on the type of anomalies being detected. Additionally, they used this new algorithm to

[6] Broader Design Context

increase the speed and accuracy of detection with phasor measurement. (Chamie et al., 2018)

- This is last year's paper written by our clients about their work to design the system of sensors we will be creating a master control for. (Ravikumar et al., 2021)

4.1.4 Technical Complexity

1. The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles –AND–
2. The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.
3. The design consists of multiple layers of components. Intrusion Detection System, machine learning algorithm, establishing stable relationships between the IDS master and sensors(nodes), and utilizing cloud computing to calculate the data.
4. The problem scope exceeds the solutions because we will be using anomaly detection systems using machine learning algorithms, not only utilizing existing cyber-attack databases.
5. Additionally, we must match multiple requirements that must protect the users safety and be able to function on its own with no errors.

4.2 Design Exploration

4.2.1 Design Decisions

1. Must use AWS-Cloud for cloud computing.
2. Must use 4 VM's: ATTACKER,TARGET,SENSOR,MASTER(SECURITY ONION).
3. Must use a Machine Learning Algorithm.
4. Must use an appropriate visual aid for the frontend.

4.2.2 Ideation

The design we chose:

Must use a machine learning algorithm

Identification method:

Lotus blossom technique

The options we considered:

Alert correlation classifications: Similarity-based (sim), Knowledge-based (K), Statistical-based (stat)

Sub-classifications:

1. Simple rules (sim)
2. Hierarchical rules (sim)
3. Scenario (K)

4. Casual Relationship Estimation (stat)
5. Statistical Traffic Estimation (stat)

4.2.3 Decision-Making and Trade-Off

Weighted Decision Matrix:

Key: H- High A- Average L - Low

	Accuracy	Flexibility	Extendability	Required Memory	Computation Power	Parallelizing
Machine Learning	A	A	A	A	A	A
Simple Rules	A	H	H	A	A	H
Hierarchical Rules	A	H	H	A	A	H
Scenario	H	H	H	A	A	L
Casual Relationship Estimation	A	L	L	A	A	H
Statistical Traffic Estimation	A	H	A	L	H	H

*Figure was derived from Mirheidari et al. See section 8.3 for full reference. [7]

We chose Machine Learning (decision-tree) over the other ideas due to the fact it matches our requirements while also being easy to understand and implement. Machine Learning uses multi-step clustering to detect alerts and attack sequences. We can use a certain training set for the system to comprehend the types of alerts that will occur. That way the algorithm can find a relationship between the attacks and not create false positives. The other ideas may be more rigorous than the decision tree algorithm, but they don't match our requirements.

4.3 Proposed Design

So far we have worked with and become experts in 4 virtual machines, labeled attacker, IDS master, victim, and IDS Sensor. For operating systems, the attacker uses Kali VM, the victim uses Windows, and the IDS master and sensor use Ubuntu for Security Onion.

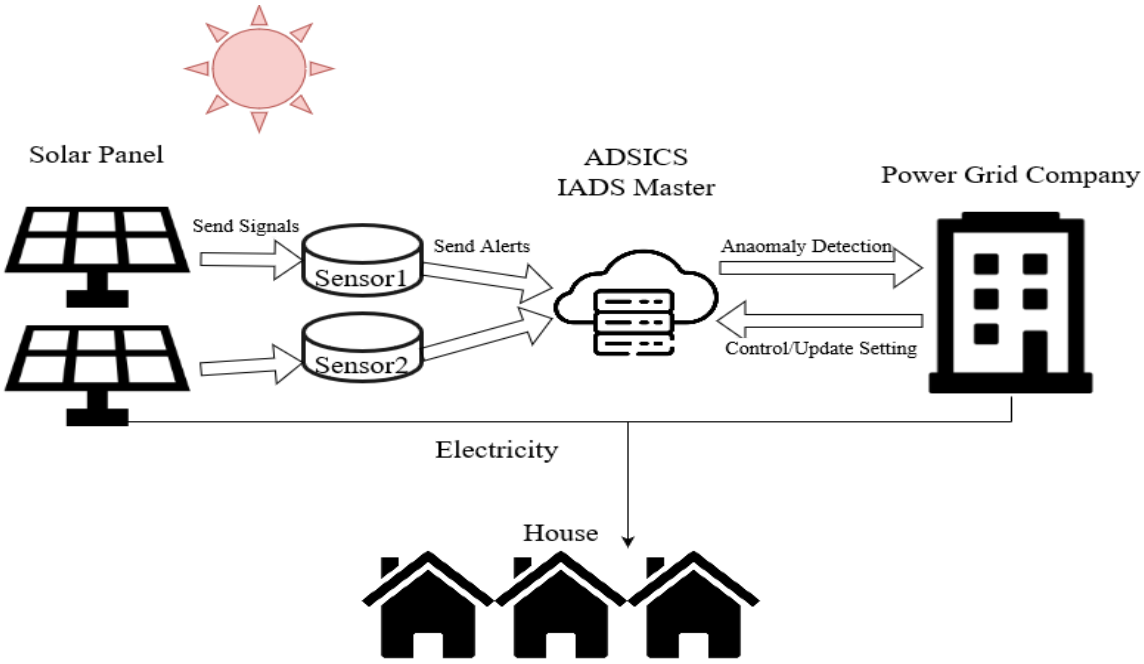
We have experimented with various machine learning algorithms including Support Vector Machine, K nearest neighbors, and Decision Tree. These algorithms were selected based on our research into the anomaly detection field (see section 8.3 for full references). They were implemented using Scikit Learn. After Decision Tree was selected as our preferred algorithm, we implemented it using ElasticSearch on the SecurityOnion testbed and tested it with KDD, a specialized cybersecurity dataset.

---

[7] Weighted Decision Matrix:

4.3.1 Design Visual and Description

CONCEPTUAL/VISUAL SKETCH:



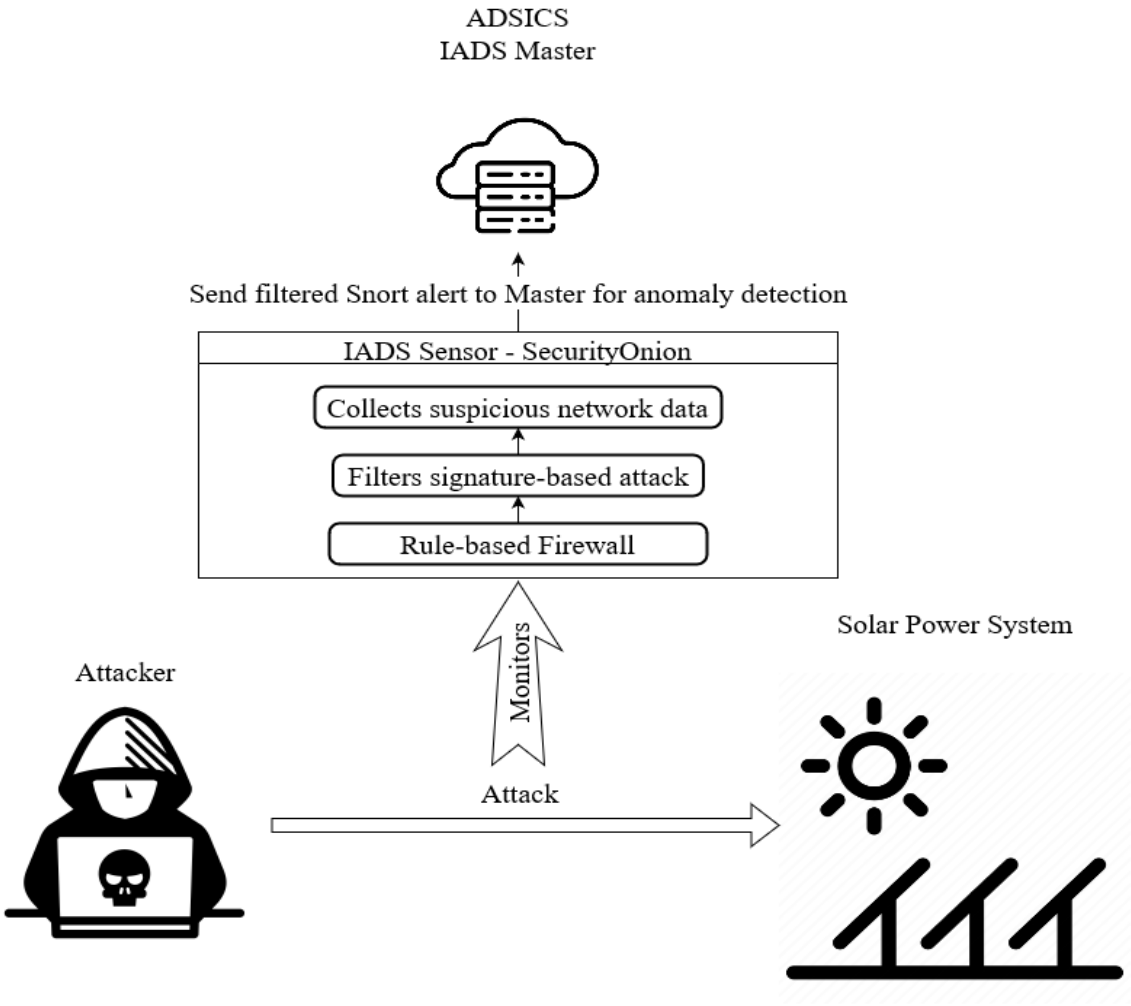
[8]

Description:

The conceptual diagram shows how a set of solar panels sends signals to sensors that will allow the IADS master to check for alerts. The power grid company will be able to check how their distributed electricity is by monitoring their anomaly detection system and controlling exactly what type of alerts they want to detect.

[8] Conceptual Diagram

COMPONENT DIAGRAM (IADS Sensor):



[9]

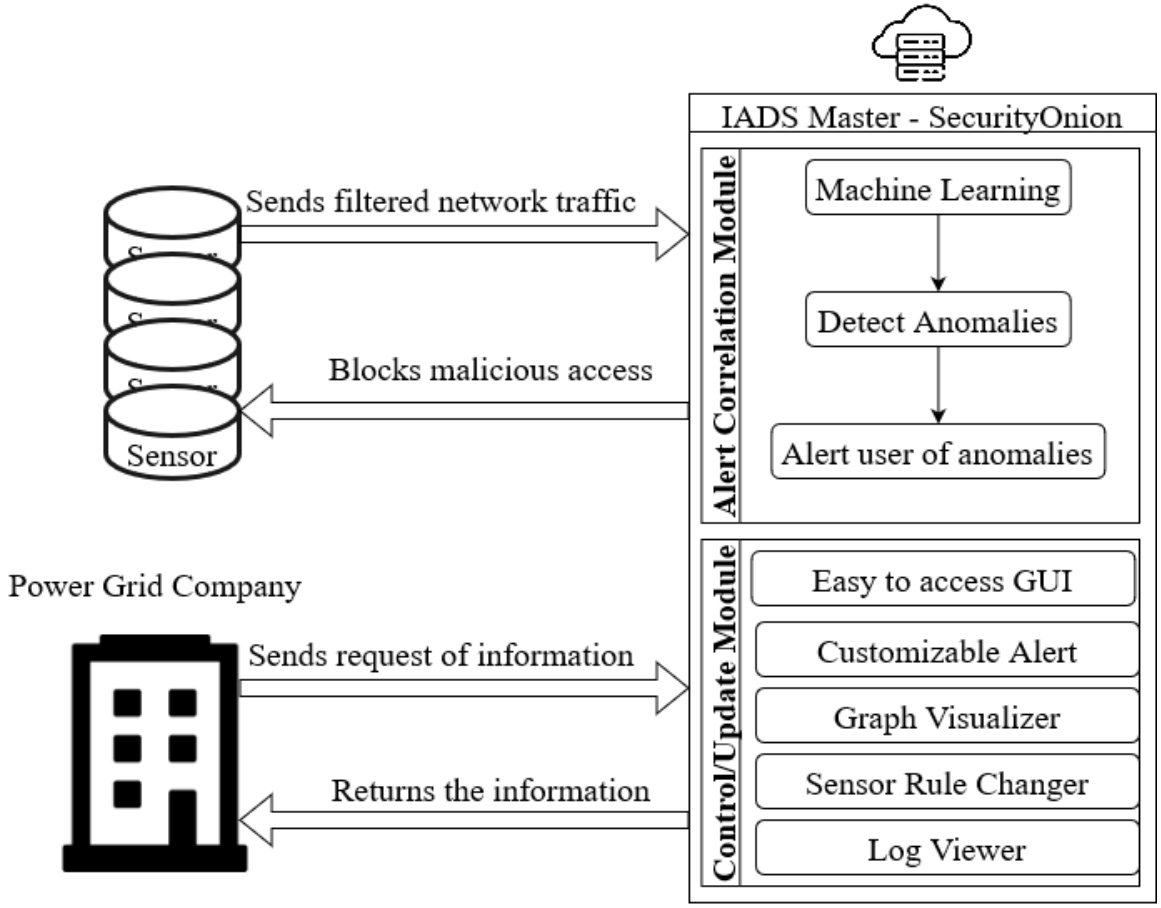
Description:

The diagram above shows how the IADS Sensor monitors cybersecurity attacks on power companies (in this case the "Solar Power System." It does so by first blocking the attacks with a firewall, then filters signature-based attacks, then collects any data from this attack if it is

[9] Component Diagram (IADS Sensor)

suspicious (a possible anomaly). Finally, the sensor will send the collected information to the IADS Master which can be seen below.

COMPONENT DIAGRAM (IADS Master):



[10]

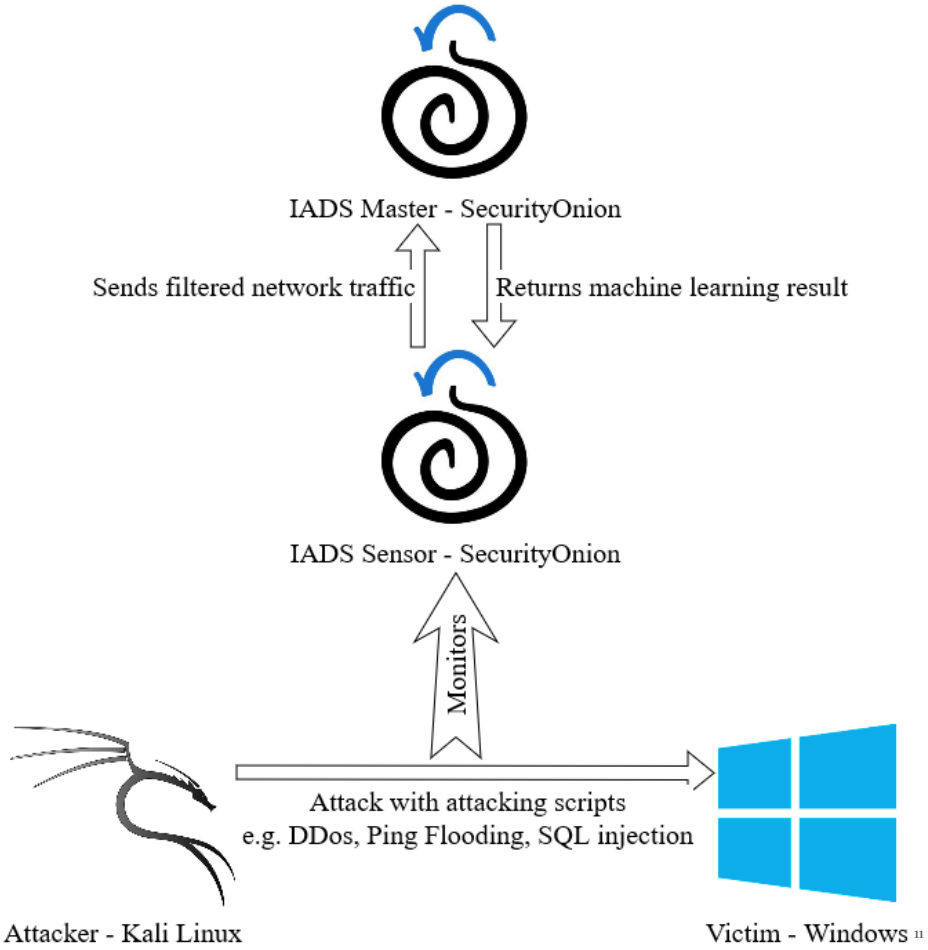
Description:

The component diagram shows how all modules interact with one another. The IADS Master is the main component that has the alert correlation module to detect anomalies and the control module to change their visual aid for anomalies. The AIDS blocks anomalies that interfere with the sensors

[10] Component Diagram (IADS Master)

and the sensors send back the filtered network traffic back to the master. The IADS Master visualizes anomalies to power grid companies when they want to monitor their network traffic.

DETAILED DIAGRAM:



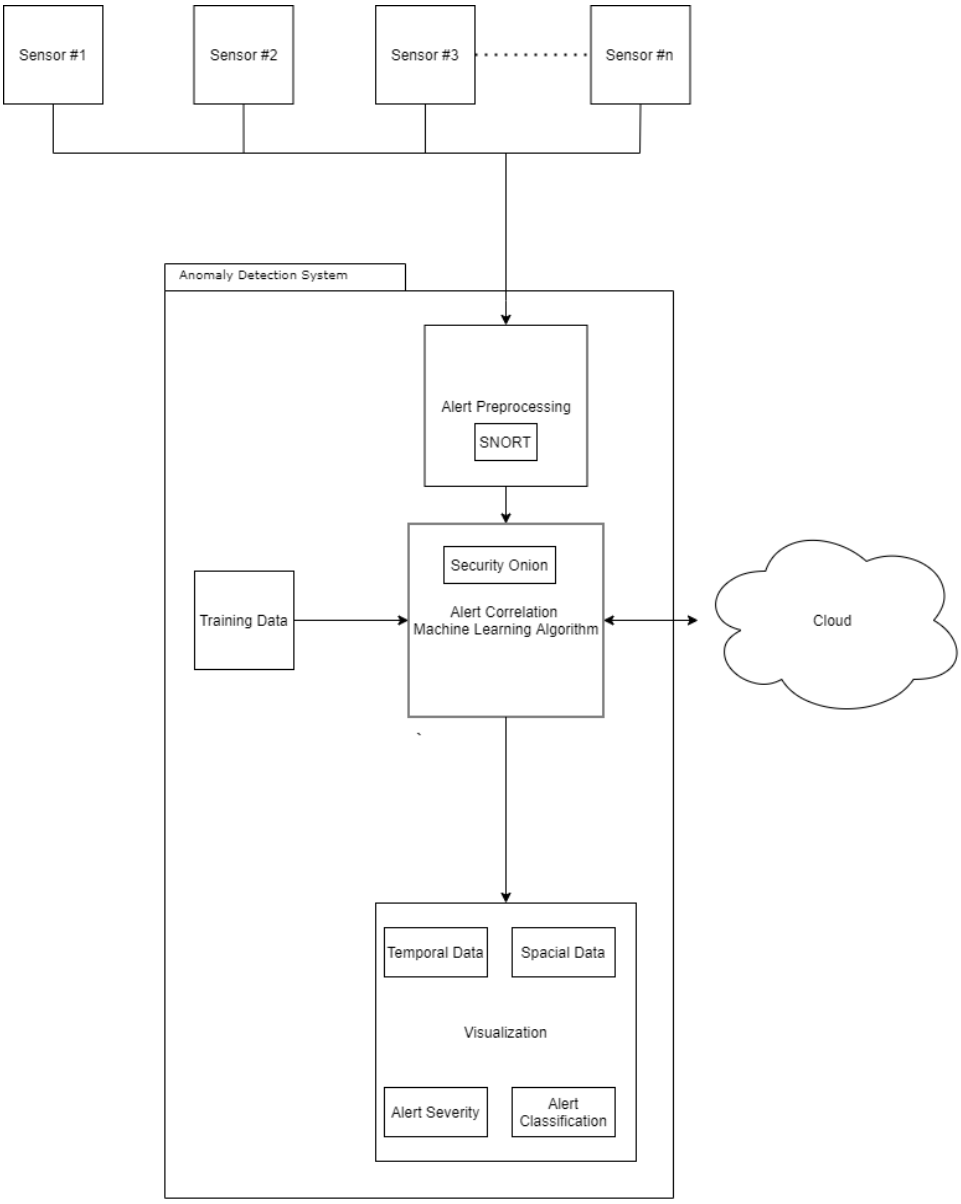
[11]

Description:

Diagram of the prototype implementation of the system. Attacker uses Kali Linux, Victim uses Windows XP, IADS Sensor and Master uses SecurityOnion

[11] Detailed Diagram

BLOCK DIAGRAM:



[12]

Description:

Our design accepts many alerts as input, originating from a variety of sensors. These alerts will be processed, ensuring that they enter the alert correlation algorithm with a uniform format and contain all the necessary information (time, location, type) for future analysis. Then, our algorithm will perform correlation on the alerts via the Cloud. Our machine learning model will be trained by

[12] Block Diagram

a dataset to draw relationships between alerts. The conclusions will then be presented to users through a variety of visualizations, allowing alerts to be filtered by time, space, type, and severity.

4.3.2 Functionality

Our design is intended to work by controlling and monitoring the sensor network built into the provided testbed. It will analyze information collected by the sensors and assign alert labels to events completely autonomously.

Our current design fulfills each requirement well by describing a detailed visual output including temporal and spatial correlation, tracking alert information, and performing machine learning analysis quickly and efficiently using the desired platforms.

Visual: Our design is intended to monitor & control a set of sensors. This design satisfies the functional requirements by having good visualization of log results for easy analysis. This design satisfies the non-functional requirements by being accessible and having a strong user interface.

4.3.3 Areas of Concern and Development

Our primary concern for the system is that it needs many functions to develop. It has to gather all alerts, filter the duplicate, flag the important ones, create an automated analysis of attacks using machine learning, give the visualized results to the user such as graphs to assist analysis, and communicate with the nodes to fetch detailed information and update firmware/software. In addition, this process must be made continuous and as uninterrupted as possible. This type of system is one that is unusual for our group, which may cause difficulties during the implementation process.

For the solution, we are using an existing intrusion detection system, Snort, to help implement the Anomaly Detection System. Snort already provides us with the raw event data in real time, so we can use it and add the machine learning algorithm to fit our needs.

Our current questions relate to how to continuously accept input to our machine learning algorithm from the sensors in a timely manner.

4.4 Technology Considerations

The choice to use the Decision Tree algorithm has a weakness, which is that it is not the most powerful algorithm available. However, it is easier to use than some other algorithms, and is easy to use with ElasticSearch. Alternatives include Support Vector Machine, Local Outlier Factor, and K Nearest Neighbors. Support Vector Machine provides additional depth, especially the one-class variant. It is also possible to use a series of decision trees to improve the algorithm's accuracy. Ultimately, the easy integration with ElasticSearch and intuitive nature makes Decision Tree a logical choice of algorithm.

4.5 Design Analysis

Our current implementation successfully demonstrates the ability to conduct machine learning on the desired platform. The necessary modifications are accepting continuous input data from the sensors and outputting temporal and spatial data to a UI.

4.6 Design Plan

Our plan is to add additional modules one by one through our Agile design process. This schedule is described in our GANTT chart. Eventually, the new modules will include the UI and the continuous sensor input. This will fulfill our use-cases by being able to connect to sensors and detect any anomaly that occurs during their operation. The UI would then serve the purpose of any worker who needs to oversee those sensors by providing the temporal and spatial data for any anomalies that occur.

5 Testing

5.1 Unit Testing

The unit we will test is the VM environment. The units within the VMs are the Attacker, Main IADS, IADS Sensor, and Victim. We will test the Victim and Attacker using snort to check what alerts are being sent through. We will test the IADS sensor and Master using security onion and elastic search to determine that our alert correlation system is correctly labeling alerts with severity, location, and time.

5.2 Interface Testing

The interfaces in our design are intrusion detection using SNORT, alert correlation using ElasticSearch, the interface using Kibana. They must work together to find, label, and display anomalies so when we test the program we are essentially testing their composition. The way these interfaces are tested is by using tools to send out simulated anomaly activity that the interfaces can show. Information must properly pass through the interfaces.

5.3 Integration Testing

Critical integration paths for our design are setting up our sensors, alert correlation master unit, and the machine learning algorithm. The reason they are so critical is because the sensors need to be created first before we can test our main control unit which uses Security Onion tools like ElasticSearch and Kibana. That way when an anomaly is sent through the sensors, the master unit can successfully detect it. The algorithm will take the longest and will go into the next semester to be created. We will update our prototype with these features and repeatedly integrate the new features to ensure that we always have a working product.

5.4 System Testing

Our system level testing strategy is black box testing. This fulfills the requirements needed because the whole system is acting like a black box since we don't know what attacks will actually be processed in the future when the software is deployed. Black box testing will help us use all of our components with similar random attacks to generate useful information.

5.5 Regression Testing

We are ensuring that any new additions do not break the old functionality by saving all prior work and testing at each new addition of the project. This way we will be able to pinpoint what additional work causes the issue and will easily be able to revert it by just that portion. If needed, we can revert back to a previous image with the snapshot feature on our VMs.

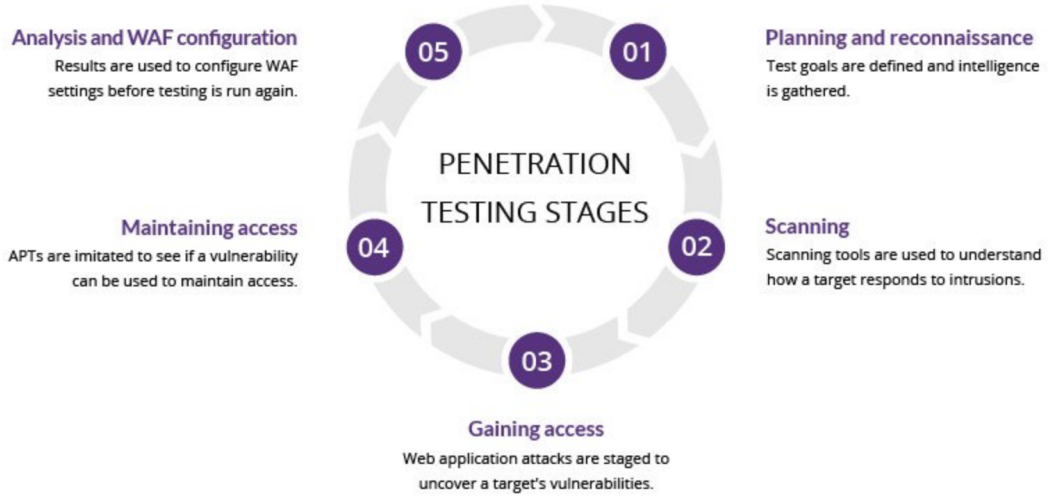
5.6 Acceptance Testing

To demonstrate both non-functional and functional requirements are being met through testing, we will go through and check off all the requirements once we get our results. After we feel confident in our results, we will give our client a demonstration and allow them to operate the system themselves. We plan to do this multiple times until the client is absolutely satisfied with the product in all aspects.

5.7 Security Testing

A majority of our testing in the second semester will be focused on security testing as this is a cyber security project. In the second semester we plan to focus on developing the master control for all the sensors so in a way the performance testing will be security testing. We plan to use the red team method as the final step of this to ensure the security of our design.

Penetration testing: Scanning tools used to understand the target's resprusion. Then, access is attempted (see image below).



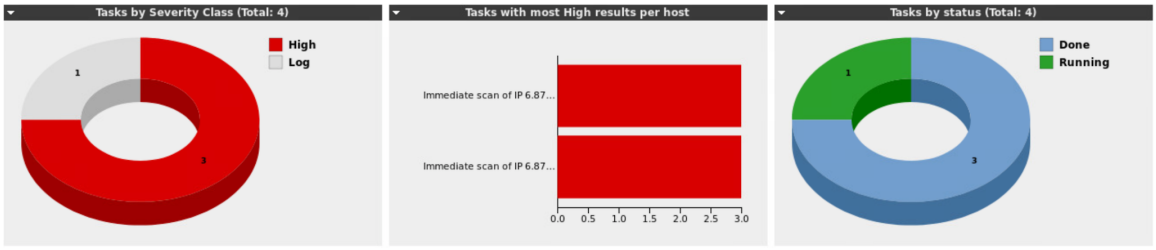
[13]

image from: imperva.com

[13] Penetration Testing

5.8 Results

We have done tests on the software we are using including one to test for vulnerabilities in our network using OpenVas which we have included below:



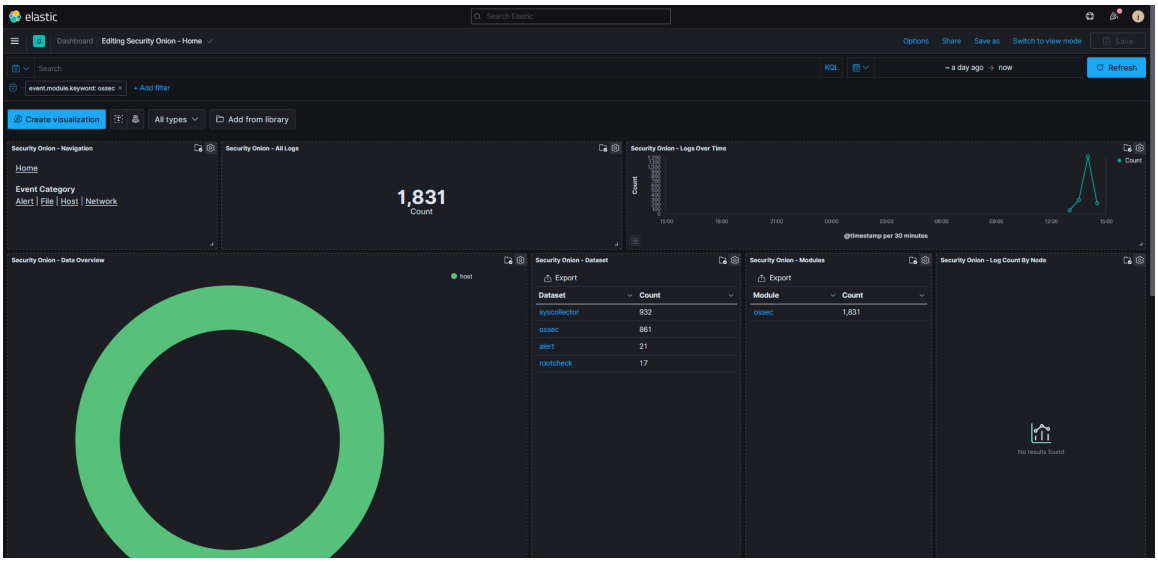
1 - 4 of 4

Name	Status	Reports		Severity	Trend	Actions
		Total	Last			
Immediate scan of IP 6.87.151.210	Done	1 (1)	Nov 18 2019	10.0 (High)		
Immediate scan of IP 6.87.151.210	Done	1 (1)	Aug 2 2021	10.0 (High)		
Immediate scan of IP 6.87.151.210	76 %	0 (1)				
Immediate scan of IP 6.87.153.210	Done	1 (1)	Aug 2 2021	10.0 (High)		

[14]

As can be seen above, we were successful in getting the software to work and run the programs we needed to.

Here is the screenshot of the Security Onion scanning the traffic of the sample network. It is using Kibana to visualize the traffic such as graphs.



[15]

[14] Elastic Search #1
[15] Elastic Search #2

This image shows the Decision Tree machine learning algorithm running anomaly detection on the KDD dataset with 99% accuracy. We will eventually feed our own data to this model during our system's runtime.

6 Implementation

Our goal for the end of this semester was to have completed the analysis of a power systems data set using security onion within our VM which was provided by the client. Next semester, our overall plan will follow Sprints 5 & 6; creating Security Onion VM to correlate alerts and building the User Interface in the Cloud to provide insights on each alerts respectfully. The preliminary implementation plan will focus on Sprint 5 and we will work on setting up the algorithm we choose within Security Onion to check our sensor data and relay the information to the master as shown in the block diagram. More information on this can be found in the Design section.

7 Professionalism

7.1 Areas of Responsibility

Area	IEEE code of Ethics
Work competence	1, 5, 6, 10
Financial responsibility	4, 9
Communication honesty	3, 7, 8
Health, safety, well-being	1, 9
Property Ownership	4, 9

Sustainability	5, 6
Social responsbility	1, 2, 5, 8

[16]

IEEE code of ethics

1) to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;
2) to avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;
3) to be honest and realistic in stating claims or estimates based on available data;
4) to reject bribery in all its forms;
5) to improve the understanding of technology; its appropriate application, and potential consequences;
6) to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;
7) to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;
8) to treat fairly all persons and to not engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression;
9) to avoid injuring others, their property, reputation, or employment by false or malicious action;
10) to assist colleagues and co-workers in their professional development and to support them in following this code of ethics.

7.2 Project Specific Professional Responsibility Areas

Area	Application to the project	How we are doing
Work competence	Yes, incompetent work would make our security system	High, we are thoroughly examining our work and

[16] Areas of Responsibility

	worthless.	requesting the advice of experts in the field to ensure our project is high-quality.
Financial responsibility	Yes, optimizing our project will reduce the amount of computing resources required, which will lower costs.	High, we are weighing all of our options and seeing which way is the best in order to optimize our project and reduce our costs.
Communication honesty	Yes, if our project misleads our client about its effectiveness, it could jeopardize the integrity of the power grid and negatively affect many lives.	High, we are in constant communication with the client to understand if we are meeting requirements.
Health, safety, well-being	No, our project's quality does affect well-being but this aspect is already covered by the Work Competence category.	N/A
Property Ownership	Yes, we are working with an existing system developed by others. We must respect their contributions.	High, we are in communications with the original owners of the system on any additions we create
Sustainability	No, optimizing our system will reduce power usage, but this aspect is already covered by Financial Responsibility	N/A
Social responsibility	Yes, protecting the solar power grid from attack benefits society by preventing power outages.	High, our project prioritizes a useful interface that presents information in a way suited for preventing threats to the power grid, which will maximize the project's benefits to society.

7.3 Most Applicable Professional Responsibility Area

Work competence	split into 1,5,6,10 IEEE Competencies Our goal is to detect and analyze various alerts, which are triggered by cyber attacks against a power grid. Therefore, work competence is very important to the project since our ability to detect attacks depends on the quality of the implementation of the detection and analysis systems. (5,6) So far, we have ensured quality by conducting a thorough review of the anomaly detection

	field and continuously reviewing and refining our work with the help of our client, who is an expert in the field of anomaly detection. (1,10)By taking this approach, we have been able to gain a complete understanding of relevant concepts for our project, and that knowledge has caused our decisions to be made with confidence. (5,6,10)We have not reached the point of obtaining data to represent our project's quality, but we are taking steps to proactively increase its effectiveness at all stages of development.

[17]

8 Closing Material

8.1 Discussion

We are unable to discuss the results of this project at this time as we have not yet completed it. However, our prototype has demonstrated the effectiveness of our current anomaly detection system design, as well as its integration with the required frameworks like ElasticSearch and the group's testbed.

8.2 Conclusion

We have researched about cybersecurity basics, researched and worked with the power grid test-bed, did data augmentation using existing open-source dataset and actual data, and experimented with machine learning algorithms on the testbed environment. We aim to use a machine learning algorithm within our VM to process alerts, ensuring that they enter the alert correlation algorithm with a uniform format and contain all the necessary information (time, location, type) to be analyzed by a user. To achieve these goals, we plan to follow the GANTT chart above.

Issues we had that constrained us were usually caused by problems within the virtual machine. Because we do not have full control over the testbed we had to wait for the TA to resolve the issue before we could continue our work. In the future, we should continue to work on other aspects of the project or work on the local machine so we can still make progress while waiting for the client to resolve the issue.

8.3 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

- KDD, ICS, NSL+KDD, IoT 23 datasets

[17] Project Specific Prof. Areas

- G. Ravikumar, A. Singh, J. R. Babu, A. Moataz A and M. Govindarasu, "D-IDS for Cyber-Physical DER Modbus System - Architecture, Modeling, Testbed-based Evaluation," 2020 Resilience Week (RWS), 2020, pp. 153-159, doi: 10.1109/RWS50334.2020.9241259.
- Mirheidari, S. A., Arshad, S., & Jalili, R. (2013, November). Alert correlation algorithms: A survey and taxonomy. In *International Symposium on Cyberspace Safety and Security* (pp. 183-197). Springer, Cham.
- Sadoddin, R., & Ghorbani, A. (2006, October). Alert correlation survey: framework and techniques. In *Proceedings of the 2006 international conference on privacy, security and trust: bridge the gap between PST technologies and business services* (pp. 1-10).
- T. Zang, X. Yun and Y. Zhang, "A Survey of Alert Fusion Techniques for Security Incident," 2008 The Ninth International Conference on Web-Age Information Management, 2008, pp. 475-481, doi: 10.1109/WAIM.2008.104.

8.4 Appendices

*__S1 TASK 1 -__ A virtual environment where security onion will be hosted and used to check out alerts coming from the victim virtual machine
*__S2 TASK 3 -__ The KDD data set is a well known benchmark in the research of Intrusion Detection techniques.
*__S2 TASK 3 -__ NSL-KDD: Benchmark for modern-day internet traffic, advanced version of KDD dataset
*__S2 TASK 3 -__ IoT-23 is a new dataset of network traffic from Internet of Things (IoT) devices. It has 20 malware captures executed in IoT devices, and 3 captures for benign IoT devices traffic.
*__S2 TASK 4 -__ Set up VM using SecurityOnion and test the basic rule-based alert correlation system using Security Onion
*__S5 TASK 2 -__ Correlate alerts using our Machine Learning Algorithm to pick up on new and old attacks
*__S6 TASK 1 -__ Create User Interface in the cloud so the user can interact with the anomaly detection system
*__S6 TASK 2,3 -__ Actual environment testing to test all of the VM's we have together to simulate attacks happening in real-time

8.4.1 Team Contract

THE TEAM - sdmay22-38

Team Members: Alex Nicolellis, Jung Ho Suh, Muhamed Stilic, Pallavi Santhosh

Required Skill Sets for Your Project: (if feasible – tie them to the requirements)

Exposure to cloud programming, cybersecurity tools, machine learning algorithms, or python programming.

Skill Sets Covered by the Team: (for each skill, state which team member(s) cover it)

Machine learning algorithms: Alex, Jungho, Pallavi

Python Programming: Alex, Jungho, Muhamed

Cloud Programming: None

Cybersecurity tools: None

Project Management Style Adopted by the Team:

Waterfall Management Style

Initial Project Management Roles: (enumerate which team member plays what role)

Planning: Pallavi

Communicating to the client: Jungho

Organizing: Alex

Controlling: Muhamed

Team Name sdmay22-38

Team Members:

1) Alex Nicolellis 2) Jung Ho Suh

3) Muhamed Stilic 4) Pallavi Santhosh

Team Procedures

1. **Day, time, and location (face-to-face or virtual) for regular team meetings:** Face-to-face, Mondays at 1:00pm at Parks

2. **Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):**
Discord

3. **Decision-making policy (e.g., consensus, majority vote):**
Consensus

4. **Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):**
Google Doc maintained by Alex

Participation Expectations

1. **Expected individual attendance, punctuality, and participation at all team meetings:**

Full attendance and participation is expected. If a member has a valid excuse they must communicate with the rest of the group in advance.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
Established deadlines must be met by all group members. Assignments can be modified if a problem arises ahead of the due date.

3. Expected level of communication with other team members:
Members must check the discord at least once a day to maintain communication. All issues should be raised there or during a face-to-face meeting.

4. Expected level of commitment to team decisions and tasks:
The team is responsible for working together to arrive at decisions that are approved by all. Once a compromise is reached, the team must commit to it or continue discussions with the group.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
Team organization: Alex
Client Interaction: Jungho
Planning: PJ
Testing: Muhamed

2. Strategies for supporting and guiding the work of all team members:
Strong and respectful communication as well as group interdependence to ensure all team members work well together.

3. Strategies for recognizing the contributions of all team members:
Planning out responsibilities and deadlines well to properly recognize each member for their individual accomplishments in the project.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
Alex: Some prior experience with researching machine learning anomaly detection algorithms.
Jungho: Connect Electrical Engineering and Software Engineering by Computer Engineering knowledge. Maintaining cybersecurity experience in a military bunker.
Muhamed: Formal methods security research.
Pallavi: Background knowledge in control systems and signals (hardware aspect).

2. Strategies for encouraging and support contributions and ideas from all team members:
Strong communication as well as group interdependence so that every member feels valued and respected.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

Open communication with team members. Positive feedback to foster a productive environment.

Goal-Setting, Planning, and Execution

1. Team goals for this semester:

Study Industrial Control Systems(ICS) and Intrusion Detection System (IDS) to find out what to develop later and possibly develop Anomaly Detection System using machine learning algorithms.

Implement client-server connection as described in the project.

2. Strategies for planning and assigning individual and team work:

Work together to fully describe the problem and break it down into multiple tasks.

Collectively agree on a fair distribution of tasks.

3. Strategies for keeping on task:

Be in regular contact with each other.

<u>Consequences for Not Adhering to Team Contract</u>

1. How will you handle infractions of any of the obligations of this team contract?

First, infractions will be handled between teammates, and a plan will be established for the offending teammate to improve.

2. What will your team do if the infractions continue?

We will contact the TA to try and intervene if the infractions continue even after a group discussion.

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to abide by these terms and conditions.*

c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Pallavi Santhosh DATE Sept. 14, 2021

2) Jung Ho Suh DATE Sept. 14, 2021

3) Alexander Nicolellis DATE Sept. 14 2021

4) Muhamed Stilic DATE Sept. 14 2021