Carnegie Mellon University
Software Engineering Institute

AN ASSESSMENT OF ML-POWERED SECURITY APPLIANCES FOR SITUATIONAL AWARENESS

Joshua Fallon
Timothy J. Shimeall

May 2021

Executive Summary

Network security teams may find the adoption and deployment of a security appliance to be expensive and time-consuming. This document presents a review process that examines how an appliance fits into and contributes to an organization's situational awareness and security posture, including identification of utility, adoption issues, and how to mitigate such issues. The review grows out of the Workflow Review of Analysis Products developed by the SEI Situational Awareness team.

Key insights produced by the review will include data dependencies and outputs, interaction with other tools and data flows, and enumeration of the principal functions of the appliance. Where machine learning (ML) is employed, the report will describe the model used, training data requirements, model management process, and required technical knowledge for staff who maintain the model.

The purpose of this assessment and report is not to solve the complex problem of ML and artificial intelligence (AI) test and evaluation. Instead, its goal is to guide an organization in developing situational awareness of the fitness of a security appliance to their needs, with particular emphasis on developing this awareness for appliances that incorporate ML and AI. This process follows guidance in NIST Special Publication 800-115 [6] for information security testing and assessment.

Introduction

Cybersecurity professionals view machine learning (ML) and artificial intelligence (AI) as helpful to support security analysis and alerting. They adopt these terms to describe security solutions that can adapt to the volatile network threat landscape. Machine learning refers to the development of models that can improve their performance upon exposure to new data without being programmed with additional explicit instructions. Artificial intelligence refers to the capacity of a system to make decisions that would traditionally require human intelligence to determine.

While ML-powered security appliances can increase versatility and decrease the time required to detect threats or respond to a security incident, it is important to consider how a tool fits organizational needs and what support is required to use the tool to its full potential. The SEI's Workflow Review of Analytical Products (WRAP) provides the basis on which this assessment process is built. WRAP is designed to provide a brief but thorough qualitative assessment of network security analysis tools.

Figure 1: Context for Appliance Evaluation

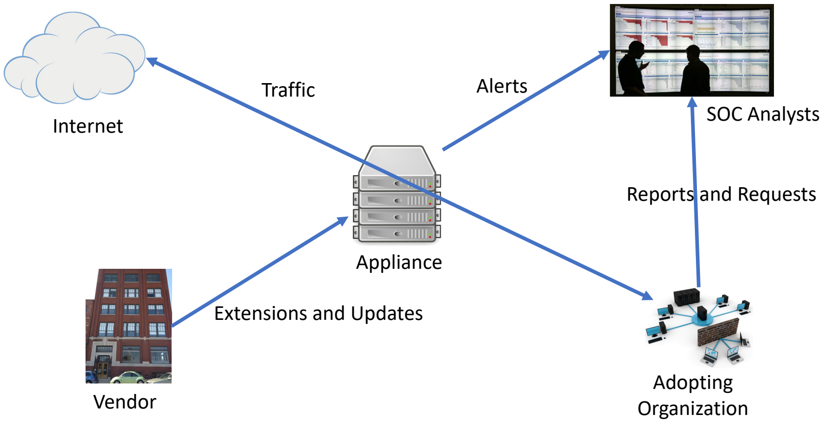

Figure 1 shows an example of a context diagram for a security appliance that monitors traffic at the adopting organization. Normally the appliance monitors traffic to and from the internet, although sometimes it monitors traffic within the organization. It uses functions and models provided by the vendor to identify specific conditions of interest in traffic for the adopting organization and generate alerts when these conditions are met. Security Operations Center (SOC) Analysts handle these reports and alerts to provide indicators, trends, and other information for the sponsoring organization.

ML models are not computational panaceas. A model provides a specific kind of information, such as classifying email as benign, spam, or phishing; clustering traffic with indicators of compromise; or identifying user or system authentication or data access as outliers from a baseline. A necessary part of the evaluation process for a tool is the specific functionality powered by machine learning. This functionality must be evaluated for its objective contribution to an organization's security posture as well the risk impact of incorporating machine learning into security as applied in the security tool.

Security appliances with built-in ML capabilities target a narrow set of use cases, such as spam detection or analyst ticket assignment. Understanding these use cases and how an ML-powered tool will improve an organization's security posture and process should be the first step in planning the acquisition of an ML-powered security appliance.

This report describes an assessment for security appliances as part of a defense-in-depth strategy. It reviews the organizational considerations and some of the impact of incorporating machine learning into a production security environment. Some prevalent use cases for machine learning built into security appliances are highlighted in an appendix.

Background

Test and evaluation for ML and AI present unique challenges, as presented in [2]. Three primary challenges are discussed in that work:

1. Systems leveraging AI struggle to enumerate requirements and testing all cases is impossible. Traditional human intuition about generalizing which cases fail does not simplify the test strategy.
2. AI systems are complex and exhibit emergent behavior. System decomposability assumptions may not be valid.
3. Systems are not “final” during the deployment. They change dynamically as algorithms learn and the environment changes.

The difficulty in evaluating security appliances that employ ML stems from several factors. The adaptability of artificially intelligent systems to provide flexibility in operation means that the capabilities of the appliance are not static. This flexibility should be captured and characterized. The evaluation team needs to assess the ML algorithms and their underlying assumptions in the context of the appliance functions, as the appliances embed the algorithms and assumptions at a sufficient depth to block that isolating them may be blocked. The variety of information flows or actions supported by the application increases the complexity of the evaluation. The role of the appliance in prevention or pre-emption of security weaknesses may override its role in alerting or identifying attempts to exploit those weaknesses, which may make evaluation of the alerting capabilities more conditional. All of these factors need to be considered when developing the evaluation methodology.

Where the original WRAP process focuses on analysis products (software and data), this assessment is designed for security appliances. This includes both software and physical devices such as firewalls, security information and event management systems, application security, email security, network segmentation, and web security—many of which currently employ machine learning.

Machine learning models typically require large quantities of data for training before they can be deployed. Planning to use a tool that provides value based on a model requires additional planning for infrastructure, cost, and data access to train and maintain the model. The focus of this assessment is how an appliance fits into an organization’s overall security posture, not a quantitative evaluation of robustness to a set of attacks or detection of a given event.

In particular, when a network defense device is based on artificial intelligence, there are not broadly accepted methods for evaluating the AI defense. Development of a methodology to evaluate the capabilities of an AI defense is the subject of an active research project at the SEI [3], [4].

This background informs the development of the assessment method. While none of the previous work is sufficient, that work does provide conceptual support for assessing the ML-supported functions of the appliances.

Assessment Overview

The remainder of this document outlines the components of an assessment of security appliances, with particular emphasis placed on the additional considerations demanded by use of machine learning. It describes the content of a report reviewing an appliance and the steps of assessing the appliance to obtain the insight presented in the report. A template for an assessment report is given in an appendix.

There are three points at which an assessment may be used:

1. When the system has been cleanly installed, but without any additional configuration or training. This is referred to as an *out of the box* assessment. The main purpose of this assessment is to gain an understanding of the major functions of the appliance, and what sort of outputs it can produce.
2. When the system has been cleanly installed and trained for the acquiring organization. This is referred to as a *trained* assessment. The main purpose of this assessment is to understand the range of applicability of the appliance in the installed environment, and to identify any gaps in its current configuration.
3. When the system has been cleanly installed, trained, and had its configuration updated to meet the needs of the acquiring organization. This is referred to as a *configured* assessment. The main purpose of this asseessment is to fully characterize how the appliance fits into the organization's work processes, and to identify the range of value added by the appliance.

At a high level, the assessment report will include the following components:

- A review of the major function of the appliance and its role in overall security posture
- A description of the data sources required for the tool to function as intended and the structure of the data
- A listing of the outputs of the appliance, whether reporting, alerts, or actions taken
- An overview of the machine learning model types used and guidance on training and testing the models
- A description of any barriers to use for this appliance, and how to mitigate those barriers

To obtain these insights, the assessment will progress through the following three stages, as permitted by access to and control over the appliance and related resources:

1. Review of appliance documentation
2. Installation and operation of appliance in a trial environment that emulates the operational environment as closely as possible—with effort varying for out-of-the-box, trained, or configured assessments
3. Systematic description, dynamic evaluation, and contextualization of the major aspects of the appliance and results of documentation review—with a degree of thoroughness depending on the purpose of the assessment

The sections that follow present an expanded view of the components of the assessment as listed above. A final section addresses the aspects specific to the capabilities suported by machine learning.

Previous Work

There have been a large number of efforts to evaluate security tools and appliances, assesing and assuring the quality of these products. These efforts have included code reviews [**?**], run-time assessments [1], and models.

Major Function

This section of the assessment will review the function of the major components of the appliance to identify the value it is intended to add to organizational security. This focuses on answering the question of what function is afforded by adding the appliance. The NIST Cyber Security Framework (CSF) lists five core functions in managing cybersecurity risk: identify, protect, detect, respond, and recover.

The *identify* function focuses on configuring pre-production processing resources to improve cybersecurity. Appliances associated with this function include static scanners and configuration validation tools. The assessment report should describe the range of potential security issues flagged by the appliance. For each flagged issue, the assessment results should determine how the appliance supports corrective workflows by the level of detail and clarity of identity to the flagged issues. If the appliance has the capability to validate addressing of the potential issues, the assessment results should characterize how this is done and the degree of accuracy obtained.

The *protect* function focuses on blocking cybersecurity threats during production use, but prior to damage from those threats. A wide range of appliances are associated with this function, including vulnerability scanners, email and web filters, data loss prevention systems, authentication/authorization servers, and firewalls. These appliances should be assessed for the range of threats they prevent. For those threats, the assessment report should characterize the appliance's reporting of the conditions in which it blocked the threat. The evaluation should also identify features of these appliances, if present, that work with detection appliances to evaluate how completely these threats were blocked.

The *detect* function focuses on signaling the presence of cybersecurity threats during production use, specifically those not blocked, or not fully blocked by the protect function. A wide range of appliances are associated with this function, including intrusion detection systems, host-based sytem call analysis systems, honeynets, network-based packet analysis systems, network-based flow analysis systems, and user behavior analysis systems. The evaluation should characterize the velocity of data supported, the degree of federation or coordination of results, the diversity of threats detected, and capabilitities to support responses to the threats.

The *respond* function focuses on diagnosing and limiting any damage caused by cybersecurity threats. Appliances that are associated with this function include intrusion prevention systems, active defense systems, and systems that proactively isolate suspect hosts or networks. The evaluation should characterize the range of networks or hosts protected, the response capabilities offered, the type of alerting or data inspection required to support response, and the notification of response generated by the appliance.

The *recover* function focuses on continuity and resumption of desired service on assets affected by cybersecurity threats or by the respond function. While some of the applications associated with this function are also associated with previous functions, particularly protect and identify, others include re-ghosting systems, dynamic host configuration monitoring, passive application usage monitoring, and service verification systems. The evaluation should characterize the functions as components of the organization's overall recovery process, including tracking the reestablishment of services and servers, validation of improved security, and identification of attempts to exploit backdoors or perform follow-on attacks.

In the context of an organization's current security posture, this assessment includes enumerating the set of previously-installed tools, separate from the appliance, that are actively being used to support or execute the functions identified in this section. Assumptions implicit in documentation about the team applying the tool, including their expertise (using definitions such as the NIST NICE[5]) and available additional resources, will be described here.

Review of the major functions will include discussion of the data processing done by the appliance, including examination of what key functions are done on-premises and in the cloud. Discussion of the architecture level being supported by the appliance, from site or enterprise network to nation-scale or internet-wide, provides visibility into the processing support needed and informs consideration of scale for relevant data sources.

To fully inform the remaining sections of the assessment, a key determination in the first pass is which of the core functions of the appliance are supported by machine learning, and to what extent. The data and support needs of an ML model are sufficiently different from static tools that each of the remaining parts of the assessment requires a different approach for ML-supported functionality. Many machine learning models require a large corpus of carefully pre-processed and labeled data for training and testing, which creates additional requirements for data sources. Model management also requires consistent access to subject-matter experts in machine learning to monitor for needed training cycles. As input data evolves or changes format, this will maintain the relevance of the processing done by the appliance.

Data Sources

This section of the assessment will review the data sources required to execute the functions of the appliance. It will address the implications of the location of an appliance with respect to boundary controls, access permissions, and other defenses.

Each data source that provides input to the appliance should be documented, including information available about the frequency with which the source is updated and the duration of data stored for persistent access. Where the appliance has a variety of applications, any that rely on data sources not available in the deployment environment should also be documented to support appropriate scoping of the use of the appliance.

Key features of the required data sources will be reflected in this section to provide insight into the impact the appliance has on data storage needs. The combination of architecture level, data detail, format, velocity, and persistence presented here serves to inform the data storage needs associated with operation of the appliance. The relative velocities of the data sources, access, and processing by the appliance should be considered to inform decisions about scaling and parallel implementation of multiple appliances, if possible, to handle larger velocities.

Data source enumeration includes a characterization of the collection methods of the appliance. Depending on the configuration and intended use, security appliances may use passive collection, host-based agents, network-based agents, or a combination thereof.

Any vendor-provided data (such as rule sets for intrusion detection or spam identification), needs to be documented in this section, along with any subscription requirements and update frequency. Where possible, the report should identify the source of this data.

The relative locations of the data source and appliance, when it is in operation, will also be documented here. This information helps to inform deployment decisions. For example, a tool that operates in the cloud as a service may require data that an organization restricts to a particular network location by security policy. Thus, the presence of necessary data sources is not sufficient to ensure an organization can make use of a tool that relies on that data.

Spam detection systems, for example, are widely employed with machine learning models informing classification. Many such models are trained on email data from many customers to generate broadly applicable rules and base models. Spam classifiers must process the deploying organization's email, so internal email servers will represent one data source for such an appliance, along with vendor rules and models.

Since the appliance under review will function as a consumer of data, flow of that data to the appliance and its internal processing must be validated against compliance restrictions on the data sources. This ensures that it inherits the most restrictive controls required of any source it uses as an input. A review of access to the appliance, its processing, and the data stored or consumed by the appliance helps confirm intended deployment is compatible with inheritance of the least privilege permitted by the data source.

When a capability is based on a machine learning model, large volumes of data and significant computing power are typically required to train and refine the model. ML-powered appliances therefore typically start with a vendor-trained model, which must then trained for additional cycles on organizational data to refine the model for the environment in which it is to be deployed. If a model is housed in the vendor's cloud environment, any data stored or processed on-premises that is processed by the model or used for training will cross network boundaries.

Intrusion prevention systems that use machine learning to detect and block files containing malware rely on frequent vendor updates to models and rules.

Appliance Output

This section of the assessment will capture the output produced by the appliance. Of particular interest is whether the appliance generates reports for analyst consumption or ingest by another tool, generates alerts on events of specific concern, or takes action on its own. Where data or reports are generated, this includes a description of the granularity of output. The processing and storage needs for effective use of the appliance output can be better evaluated and planned for with this information. Auditing the output of the appliance for responsiveness, consistency, and comparison to other tools or reports produces an essential additional output for continuous maintenance of the appliance. In addition to organization-specific auditing practice, this section will include a description of the auditing capability built into the appliance and the metrics reported in those audits.

A malware detection engine, for example, may produce alerts when files included as attachments to emails or downloaded from the Internet have a high similarity to known malicious software, or originate from a domain with a name that correlates to suspicious activity. In addition to alerts, such an engine may prevent access to a file labeled as malicious or remove it immediately upon detection.

The content of reports or alerts and the intended recipient of the alerts, whether other tools or analysts, will be recorded in this section. This can be used to compare with the needs of analysis tools already in use, such as network management software and dashboards. While the content of appliance data output is evaluated for compliance with reporting requirements, the value added to more broadly applied organizational reporting should be considered, and that reflected in this section.

The presence in an email of files labeled as malicious by a malware detection engine may be passed to a spam filtering appliance to ingest the metadata associated with the message as identifying characteristics for spam, or to an intrusion prevention system to block domains and IP addresses associated with the message.

Similarly, the assessment should record the types of actions taken by the appliance, if it acts autonomously, to illustrate the impact on affected traffic and systems. Identifying the scope of autonomous behavior will facilitate monitoring of and reporting on affected systems, ensuring that appropriate controls are maintained for those systems.

As with data sources, this section will record the relative locations of appliance and consumers for which it is a data source. This reference helps ensure the appliance can be used as intended, with no sensitive data flowing across network or enclave boundaries without confirming compliance with security requirements.

If a spam detector uses a model trained on data from many customers, in addition to mail data possibly being processed in an environment separate from the mail server, the appliance may by default share information about identified rules or model updates with the vendor. Any such data sharing must be documented to inform risk acceptance.

The output of the appliance, in both content and level of detail, should be compared with other tools already in deployment. This comparison will provide insight into three key features of the value added by the appliance. First, it shows the extent to which reporting or aggregation produced by the appliance is redundant with other tools. Second, it shows the utility of the appliance in confirming reporting or alerts from other tools already in use. Third, it produces a gap analysis identifying aspects of the overall security footing into which the appliance provides new visibility and analytic value.

Assessing Protection

With the data input and output by the appliance established and the major functions enumerated, the effective protections can be assessed in multiple ways. This section of the report does not provide a numeric score or quantitative comparison to alternatives. Instead, it provides a method for identifying strengths and weaknesses in the capacity of the appliance to strengthen an organization's security posture.

A paper assessment will evaluate the flow of data into the appliance, the processing done by the appliance, and the results produced. It represents the transformation of raw data to security information or action in a way that can be compared to deployed or competing solutions. The agility of the protection afforded by the appliance will be made clear here as the collection methods, whether passive or active, data source refresh cycle, and target of data output are presented in one location.

The paper assessment should include open dialogue with the vendor of the product to gain clarification and expand on information available in marketing materials and documentation. This may include case studies the vendor has developed in partnership with other organizations that have deployed the appliance, deep dives into architecture planning in the proposed deployment context, or connections with the vendor's technical staff to understand best practices.

A dynamic assessment will align the data flow of the paper assessment to organizational data. It incorporates a set of test inputs that are representative of live data flows to generate sample outputs that can be compared to competing or deployed solutions. This test can be more difficult to achieve without access to the appliance on the organization's own network. Where ML models are used, dynamic assessment may only be feasible on the vendor stock models, as training new models on live data takes time and a large quantity of real inputs.

The results of the dynamic assessment will provide detailed information for planning organizational infrastruture around the appliance. This assessment should generate volumetric and resource consumption data, which can be scaled up to a size representative of full deployment to predict load and align to the supporting resources available. If the appliance uses ML, the vendor stock model can be compared to any already deployed tools and processes that serve the same function as the appliance; if time and resources permit, this comparison should be iterated over multiple training cycles of the model.

Machine Learning (ML)

The previous sections of this report included some distinctions for appliances powered by ML. This section of the report discusses general considerations for the use of ML in a security appliance that apply to all aspects of an evaluation.

Machine Learning Models

While the major functions of a security appliance may be supported by ML, there are many different things this can mean. For completeness, any assessment of an ML solution must include information on the models in use.

This section of the assessment describes whether the tool clusters input data to learn rules, such as markers of spam email, or classifies input according to organizational need, such as categorizing network traffic as VoIP, streaming, email, or other relevant classes. Many ML techniques learn rules or categorize data; ideally, the specific type of model used is disclosed by the vendor, even if the finer details of the model design are proprietary. Some models, including some classifiers, output a probability distribution across class options. Determining whether this distribution is available, or just the output class label, is useful for understanding what methods are available for continuous evaluation of the appliance.

Learning Types

To identify the kinds of data needed to effectively train an ML model, an organization needs to know what type of learning is employed. Supervised learning requires a large quantity of labeled data, typically in a consistent format. Therefore, if such labeled data is not available in sufficient quantity to achieve the desired scores on ML evaluation metrics initially, more data will have to be captured, obtained, or labeled. This represents additional cost and time that must be considered.

Semi-supervised learning and unsupervised learning, typically used for different kinds of models than supervised learning, present similar requirements for quantity and type of data. A vendor should be able to supply guidance on how much data, and of what type, is needed to train an initial model or adapt the vendor-trained model to the specific context of the acquiring organization.

Whatever the supervision level, machine learning can also be either shallow or deep. Shallow machine learning requires expert intervention to select the specific features—combinations of data fields—on which to train the model. Deep machine learning does not have this requirement, but does require further data on which to automatically select features, and the models may train more slowly.

For shallow ML models, the issues are how much flexibility the appliance supports in selection of features to learn, and how easily those features may be selected via the appliance interface. The adopting organization would need to either ensure sufficient data science knowledge on the part of its staff to provide expertise in feature selection, or ensure that the vendor can provide after-market support to support this selection. Over time, features will change in criticality and predictive power. Expert intervention will be required periodically over the usage of the appliance, whether from the vendor or from within the adopting organization.

For deep ML models, the selection of training data is more critical than for shallow, as the appliance is learning both which features to track and what parameters or rules to apply to those features. This involves both careful selection of data sources and sufficient cleaning of the data to assure that distracting cases are excluded. The data processing to avoid distracting cases must be balanced with the need to maintain an accurate representation of the usage environment. As needs change over time, this data cleaning process will need to be applied to build successive retraining sets, and the appliance documentation needs to make clear both the level of data needed and any format or content constraints.

For models that learn parameter values to match data (e.g., clustering models or logistic regression models), the appliance assessment needs to ensure that the range of possible parameter values meets the needs of the adopting organization. An appliance designed to operate on local area networks may have host-count limitations that make it unsuitable for wide-area or ISP-level networks. The assessment of the appliance needs to ensure these scoping limitations are identified and consistent with the adopting organization.

For models that learn rules to apply against the data (e.g., association models or random forest models), the appliance assessment needs to ensure the rule learning minimizes or excludes coincidental connections rather than indicative ones (e.g., rather than learning “email is frequently proceeded by a DNS query”, learn “An email without a preceeding DNS query is likely spam, unless from a source that frequently contacts the organization”). The range in rules supported by the appliance, and the ability to extend this set of rules, should also be assessed to determine if the appliance will suit the adopting organization.

Training and Retraining

If an appliance ships with a model that is pre-trained by the vendor, this model can be evaluated by a variety of metrics on organizational data. For applications such as malware identification, a vendor model can be sufficient for a majority of the major functions of an appliance. In other cases, the utility of a tool relies heavily on the extent to which the underlying model is trained on data specific to the organization.

Clear expectations should be set regarding the expected or desired training interval, whether on vendor or organizational data. For some machine learning models, the accuracy of the model degrades over time since the last training. The retraining interval is a balance between effort dealing with the decrease in accuracy by retraining or by compensating for inaccurate results. Regular model updates maintain the reliability of an appliance in the face of system, network, architecture, and threat volatility. Providing information on the retraining interval will allow the organization to scope the effort required for use of the appliance.

Interpretability

Explainability in AI is an active field of research. The inner workings of an ML model are often opaque and the set of features used to produce an output (and their relative importance) cannot be easily determined. Some models, like decision tree classifiers, may record a set of thresholds for particular values as the basis on which the classifier assigns a class label.

The environment and function of an individual appliance may have transparency requirements for all systems. The extent to which the set of calculations that lead to a model output is explainable must be taken into consideration when assessing the fitness of an appliance to a deployment location and function.

Machine Learning Robustness

Incorporating machine learning into a security tool introduces new opportunities for compromise. The risks to which ML exposes a security function must be understood as well as possible and accepted before an appliance can be used. Adoption of machine learning introduces a new attack vector, as the model itself may be exploited along with the data flow and output content. Another risk is algorithm bias, by which an appliance that operates using a technique perceived as more sophisticated than its competitors may be favored, even in the presence of strong evidence of contradictory results.

Although an industry standard methodology for assessing the weaknesses of ML security tools does not yet exist, research is actively being conducted in this area. Some areas of concern that should be examined as thoroughly as possible, espcially with vendor input, are described below.

Training data may be compromised to include malicious content labeled as benign so a trained model will not identify certain threats. To some extent this attack can be mitigated against by tight controls over the data used to train the model. However, this may not always be possible given the large quantity of data that is often needed and the nature of unsupervised training.

An attacker may use a gradual increase of malicious activity to corrupt an unsupervised model that learns a baseline of behavior to incorporate the actions of the attacker into the baseline that is evaluated as benign. Even after discovery, this can have a profound impact on the use of the

ML-powered appliance, as the model will require a more robust retraining than is typical during a refresh cycle.

Conclusion

Before planning robust testing and evaluation for a security appliance, an organization must develop situational awareness of the impact of acquiring and deploying the appliance. A structured review of the deployment context of the appliance and its data flows, actions taken and outputs generated, and machine learning support will produce a report with valuable insights that can be used in the planning process.

The review itself has two principal phases. First, a thorough documentation review will be performed to identify the functionality and support needs of the appliance as described by the vendor. This review will inform production of detailed context and data flow diagrams. Second, a dynamic assessment will be performed, operating the appliance in an environment structured as described in the documentation and using data modeled as closely as possible on the planned deployment context.

The situational awareness derived from the documentation, steps carried out for testing, and insights gained from dynamic assessment will be collected in a report. Its structure should follow the template provided in the appendix to this document.

Appendix: Assessment Report Template

Title and Author

The title of the report should clearly identify the appliance being evaluated and the role for which the appliance was considered. The authors should be specified in the order of contributions to the assessment, or alphabetically by surname if contributions are equal.

Descriptive Summary

This section will provide a brief summary of the major functions of the appliance, with the associated NIST CSF functions identified. A context diagram for the appliance, noting major information flows, should be included, and sufficient background explanation to support the remainder of the assessment report. If possible, also include an architecture diagram for the appliance.

Assessment Process

This section will provide a summary of the steps followed in performing the assessment, including any assumptions made, what documentation was reviewed, how the assessment installation was done, any customization in that installation, support provided by the vendor, how the ML/AI functions were identified, what data was used and in what format during the assessment, and any other relevant details that will help the reader by giving context to the assessment results.

Proposed Use Cases

This section will identify use cases/user stories for the appliance in the context of the adopting organization. Included in these cases are a brief rationale for the utility of the appliance in each case, advantages and disadvantages in each case, and some assessment of the effort requirements of each case. For the AI/ML functions, the use cases need to explicitly identify training required, possible sources or steps for that training, explanation of results, and actions possible using those results.

Strengths and Weaknesses

This section will identify the strengths of the appliance in providing its major functions, coupled with any weaknesses in performance that were identified in the assessment. The strengths should be stated directly, without advocacy for the appliance, and the limitations should be stated clearly, without apology or denigration of the appliance.

Weakness Mitigation Recommendations

This section will identify possible workarounds, supplemental tools that may help, and configuraton mitigations that may address identified weaknesses. Where mitigations appear to be difficult, this should be stated.

Conclusion

This will summarize the assessment and explain recommendations regarding the utility of the appliance.

Appendix: Selected ML Use Cases for Security Appliances

Documentation from a variety of vendors reflects a set of common applications of machine learning in security appliances. These examples are not meant to be comprehensive, but rather to illustrate some contexts in which the above considerations are to be examined.

Ticket Ownership Assignment

When an indicator of compromise is detected and an incident ticket opened, ownership of the response is assigned to an individual or team. Some naïve approaches include simple rotation of analysts, priority based on current ticket load, and static indicator severity alignment to analyst time-to-close history.

A machine learning model used to assign tickets to owners may train on a larger set of features, including incident features like severity, relation or similarity to known threat, geolocation, DNS records, and incident ownership features like historic time to resolve, status, phase, and incident load. Such a model is designed to empower a security tool to more accurately assign incident ownership to the analyst who can most effectively resolve an incident, including clustering similar incidents under the same owner.

Analyst Support Recommendation

Similar to incident ownership assignment, some tools employ a machine learning model to identify analysts working on related tickets. In addition to the features used for ticket ownership assignment, such models can include clustering of incidents to identify common features to encourage sharing of query results to reduce duplication of effort.

In the same vein, incident response playbooks can be used as inputs to the model, aligning playbooks based on shared features of different incidents to alert analysts to results they obtain in investigations that should be shared with other incident responses.

Playbook Selection

Some security orchestration platforms use clustering to build playbooks based on actions taken in response to prior incidents. This may involve multiple models, clustering response actions based on features of an indicator of compromise in one model and classifying which playbook to execute in another model, based on features of playbooks and of the incident ticket.

A model that learns collections of responses can reflect learning of analyst expertise, supporting the training of junior analysts based on recommendations that incorporate historical actions of more experienced responders. Automated playbook development supports consistent application of best practices and, in some cases, is paired with autonomous execution of the playbook in part or in whole to reduce analyst active time, especially in the early stages of incident response.

References

[1] A. R. Chandan and V. D. Khairnar. Security testing methodology of iot,. pages 1431–1435, 2018.

[2] Laura Freeman. Test and evaluation for artificial intelligence. *INSIGHT*, 23(1):27–30, 2020.

[3] Shing-Hon Lau. AIDE: Artificial intelligence defense evaluation. 2020.

[4] Shing-Hon Lau and Grant Deffenbaugh. Artificial intelligence defense evaluation. 2020.

[5] Rodney Petersen, Danielle Santos, Matthew Smith, Karen Wetzel, and Greg Witte. Workforce framework for cybersecurity(nice framework). Technical Report Special Publication 800-181, National Institutes of Science and Technology, November 2020.

[6] Karen Scarfone (NIST), Murugiah Souppaya (NIST), Amanda Cody (BAH), and Angela Orebaugh (BAH). Technical guide to information security testing and assessment. Technical Report NIST Special Publication (SP) 800-115, National Institute of Standards and Technology, Gaithersburg, MD, 2008.

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612
Phone: 412.268.5800 | 888.201.4479
Web: www.sei.cmu.edu
Email: info@sei.cmu.edu